

Package: ruODK (via r-universe)

October 9, 2026

Type Package

Title An R Client for the ODK Central API

Version 1.6.4

Description Access and tidy up data from the 'ODK Central' API. 'ODK Central' is a clearinghouse for digitally captured data using ODK <<https://docs.getodk.org/central-intro/>>. It manages user accounts and permissions, stores form definitions, and allows data collection clients like 'ODK Collect' to connect to it for form download and submission upload. The 'ODK Central' API is documented at <<https://docs.getodk.org/central-api/>>.

License GPL-3

URL <https://docs.ropensci.org/ruODK>, <https://github.com/ropensci/ruODK>

BugReports <https://github.com/ropensci/ruODK/issues>

Depends R (>= 4.1)

Imports dplyr (>= 1.1.4), fs (>= 1.6.5), glue (>= 1.8.0), httr2 (>= 1.0.0), janitor (>= 2.2.0), lifecycle (>= 1.0.4), lubridate (>= 1.9.3), magrittr (>= 2.0.3), purrr (>= 1.0.2), readr (>= 2.1.5), rlang (>= 1.1.4), semver (>= 0.2.1), stringr (>= 1.5.1), tibble (>= 3.2.1), tidyr (>= 1.3.1), usethis (>= 3.0.0), xml2 (>= 1.3.6)

Suggests covr (>= 3.6.4), DT (>= 0.33), ggplot2 (>= 3.5.1), here (>= 1.0.1), httr (>= 1.4.7), jsonlite (>= 1.8.8), knitr (>= 1.48), lattice (>= 0.22-6), leafem (>= 0.2.3.9007), leaflet (>= 2.2.2), leafpop (>= 0.1.0), listviewer (>= 4.0.0), mapview (>= 2.11.2.9000), rmarkdown (>= 2.29), roxygen2 (>= 7.3.2), sf (>= 1.0-19), skimr (>= 2.1.5), terra (>= 1.7-83), testthat (>= 3.2.1.1), tmap (>= 3.3-4), withr (>= 3.0.2)

VignetteBuilder knitr

RdMacros lifecycle

Remotes r-spatial/leafem, r-spatial/mapview

Encoding UTF-8

Language en-AU

LazyData true

Roxygen list(markdown = TRUE)

X-schema.org-applicationCategory Data Access

X-schema.org-keywords database, open-data, odk, api, data, dataset

Config/roxygen2/version 8.1.0

Config/pak/sysreqs

cmake git make libgit2-dev libicu-dev libuv1-dev libxml2-dev libssl-dev libx11-dev

Repository <https://humanitarian-user-group.r-universe.dev>

Date/Publication 2026-10-09 13:24:10 UTC

RemoteUrl <https://github.com/ropensci/ruODK>

RemoteRef HEAD

RemoteSha ded8a7ecf0267a5d6693d980f769ab97dc504ade

Contents

app_user_create	6
app_user_delete	8
app_user_list	9
assignment_actors	11
assignment_grant	13
assignment_list	14
assignment_revoke	16
attachment_delete	17
attachment_get	19
attachment_link	21
attachment_list	23
attachment_upload	25
audit_get	27
config_public	29
drop_null_coords	30
encryption_key_list	31
entity_audits	33
entity_bulk_delete	36
entity_bulk_restore	38
entity_changes	40
entity_create	42
entity_creators	46
entity_delete	48
entity_detail	50
entity_geodata	53
entity_geojson	55
entity_list	57
entity_restore	59

entity_update	61
entity_versions	64
entitylist_create	66
entitylist_delete	69
entitylist_detail	71
entitylist_download	73
entitylist_list	76
entitylist_property_create	78
entitylist_property_delete	80
entitylist_trash_download	82
entitylist_update	84
form_assignment_actors	86
form_assignment_grant	89
form_assignment_list	90
form_assignment_revoke	92
form_assignment_role_list	94
form_attachment_download	95
form_attachment_list	97
form_create	99
form_dataset_diff	101
form_delete	103
form_detail	105
form_draft_attachment_delete	106
form_draft_attachment_download	108
form_draft_attachment_link	110
form_draft_attachment_list	112
form_draft_attachment_upload	114
form_draft_create	116
form_draft_dataset_diff	118
form_draft_delete	119
form_draft_detail	121
form_draft_publish	123
form_draft_xlsx	125
form_draft_xml	126
form_link	128
form_list	130
form_restore	132
form_schema	133
form_schema_ext	137
form_schema_parse	141
form_update	142
form_version_attachment_download	144
form_version_attachment_list	146
form_version_detail	148
form_version_list	150
form_version_xlsx	152
form_version_xml	154
form_xlsx	155

form_xml	157
fq_attachments	159
fq_data	160
fq_data_strata	161
fq_data_taxa	162
fq_form_detail	163
fq_form_list	163
fq_form_schema	164
fq_form_xml	165
fq_meta	165
fq_project_detail	166
fq_project_list	167
fq_raw	167
fq_raw_strata	168
fq_raw_taxa	169
fq_submission_list	170
fq_submissions	170
fq_svc	171
fq_zip_data	172
fq_zip_strata	172
fq_zip_taxa	173
fs_v7	173
fs_v7_raw	174
geo_fs	175
geo_gj	175
geo_gj_raw	176
geo_gj88	176
geo_wkt	177
geo_wkt_raw	178
geo_wkt88	178
get_one_attachment	179
get_one_submission	181
get_one_submission_att_list	183
get_one_submission_audit	185
handle_ru_attachments	187
handle_ru_datetimes	189
handle_ru_geopoints	190
handle_ru_geoshapes	192
handle_ru_geotraces	193
odata_entitylist_data_get	195
odata_entitylist_metadata_get	198
odata_entitylist_service_get	201
odata_metadata_get	203
odata_service_get	204
odata_submission_get	206
odata_submission_rectangle	210
odata_svc_parse	212
parse_odkc_version	212

project_assignment_actors	213
project_assignment_grant	215
project_assignment_list	217
project_assignment_revoke	218
project_create	220
project_delete	221
project_detail	223
project_enable_encryption	224
project_list	226
project_replace	228
project_update	229
public_link_create	231
public_link_delete	233
public_link_detail	235
public_link_list	237
public_link_update	238
role_detail	240
role_list	242
ru_msg_abort	243
ru_msg_info	244
ru_msg_noop	245
ru_msg_success	245
ru_msg_warn	246
ru_settings	247
ru_setup	248
semver_gt	251
semver_lt	252
split_geopoint	253
split_geoshape	254
split_geotrace	256
submission_audit_get	259
submission_changes	261
submission_comment_create	262
submission_comment_list	264
submission_create	266
submission_csv	268
submission_delete	270
submission_detail	272
submission_draft_attachment_delete	273
submission_draft_attachment_download	275
submission_draft_attachment_list	277
submission_draft_attachment_upload	279
submission_draft_create	281
submission_draft_export	283
submission_draft_get	285
submission_draft_keys	287
submission_draft_list	288
submission_edit	290

submission_export	292
submission_geodata	296
submission_geojson	298
submission_get	300
submission_list	302
submission_restore	304
submission_review	306
submission_submitters	308
submission_update	309
submission_version_attachment_delete	312
submission_version_attachment_download	314
submission_version_attachment_list	316
submission_version_detail	318
submission_version_geojson	320
submission_version_xml	322
submission_versions	324
user_create	326
user_delete	327
user_detail	329
user_list	330
user_preference_project_delete	332
user_preference_project_set	334
user_preference_site_delete	335
user_preference_site_set	337
user_reset_password	338
user_update	340
user_update_password	341

Index	344
--------------	------------

app_user_create	<i>Create a new App User.</i>
-----------------	-------------------------------

Description

[Experimental]

Usage

```
app_user_create(  
    pid = get_default_pid(),  
    display_name,  
    url = get_default_url(),  
    un = get_default_un(),  
    pw = get_default_pw(),  
    retries = get_retries()  
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code> . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
display_name	(character) The friendly nickname of the App User to create.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

The only information required to create a new App User is its display name (called "Nickname" in the administration panel). When an App User is created, it is assigned no rights. To grant rights, assign the App User a Form Role with `form_assignment_grant()`.

Value

A tibble with one row holding the new App User's metadata as columns, including the token to authenticate requests as this App User. The token is only returned at creation time.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#creating-a-new-app-user>

Other user-management: `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

au <- app_user_create(display_name = "My App User")
```

```

au |> knitr::kable()

## End(Not run)

```

app_user_delete

Delete an App User.

Description

[Experimental]

Usage

```

app_user_delete(
  pid = get_default_pid(),
  id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
id	(numeric) The integer ID of the App User.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Deleting an App User revokes its access.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#deleting-an-app-user>

Other user-management: `app_user_create()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

au <- app_user_list()

app_user_delete(id = au$id[[1]])
# > $success
# > [1] TRUE

## End(Not run)
```

app_user_list	<i>List all App Users of a Project.</i>
---------------	---

Description

[Experimental]

Usage

```
app_user_list(
  pid = get_default_pid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code>. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code>. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: <code>get_default_un</code>. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: <code>get_default_pw</code>. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

App Users may only be created, fetched, and manipulated within the nested Projects subresource, as App Users themselves are limited to the Project in which they are created.

Value

A tibble with one row for each App User of the Project as per the ODK Central API docs. Column names are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#listing-all-app-users>

Other user-management: `app_user_create()`, `app_user_delete()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

au <- app_user_list()

au |> knitr::kable()

## End(Not run)
```

assignment_actors	<i>List all Actors assigned some server-wide Role.</i>
-------------------	--

Description

[Experimental]

Usage

```
assignment_actors(
  role_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

role_id	(character or numeric) The Role ID, typically the integer ID of the Role. A Role system name can also be supplied if the Role has one.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

Given a Role, this endpoint lists all Actors that have been assigned that Role upon the entire system. This endpoint requires ODK Central v0.5 or later.

Value

A tibble with one row for each Actor assigned the Role as per the ODK Central API docs. Column names are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#listing-all-actors-assigned-some-role>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

aa <- assignment_actors(role_id = "admin")

aa |> knitr::kable()

## End(Not run)
```

assignment_grant	<i>Assign an Actor to a server-wide Role.</i>
------------------	---

Description

[Experimental]

Usage

```
assignment_grant(
  role_id,
  actor_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version()
)
```

Arguments

role_id	(character or numeric) The Role ID, typically the integer ID of the Role. A Role system name can also be supplied if the Role has one.
actor_id	(numeric) The integer ID of the Actor.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

Details

Given a Role and an Actor, this endpoint assigns that Role to that Actor upon all system objects. No request body data is required. This endpoint requires ODK Central v0.5 or later.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#assigning-an-actor-to-a-server-wide-role>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

assignment_grant(role_id = "admin", actor_id = 14)
# > $success
# > [1] TRUE

## End(Not run)
```

assignment_list

List all server-wide Assignments.

Description

[Experimental]

Usage

```
assignment_list(
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This lists every server-wide Assignment in the form of actor_id/role_id pairs. It does not list Project-specific Assignments; those need `project_assignment_list()`.

Value

A tibble with one row per Assignment and the columns actor_id and role_id.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#listing-all-assignments>

Other user-management: [app_user_create\(\)](#), [app_user_delete\(\)](#), [app_user_list\(\)](#), [assignment_actors\(\)](#), [assignment_grant\(\)](#), [assignment_revoke\(\)](#), [form_assignment_list\(\)](#), [project_assignment_list\(\)](#), [role_detail\(\)](#), [role_list\(\)](#), [user_create\(\)](#), [user_delete\(\)](#), [user_detail\(\)](#), [user_list\(\)](#), [user_preference_project_delete\(\)](#), [user_preference_project_set\(\)](#), [user_preference_site_delete\(\)](#), [user_preference_site_set\(\)](#), [user_reset_password\(\)](#), [user_update\(\)](#), [user_update_password\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "....svc", un = "me@email.com", pw = "...")

al <- assignment_list()

al |> knitr::kable()

## End(Not run)
```

assignment_revoke

*Strip a server-wide Role Assignment from an Actor.***Description****[Experimental]****Usage**

```
assignment_revoke(
  role_id,
  actor_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version()
)
```

Arguments

role_id	(character or numeric) The Role ID, typically the integer ID of the Role. A Role system name can also be supplied if the Role has one.
actor_id	(numeric) The integer ID of the Actor.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

Details

Given a Role and an Actor, this endpoint unassigns that Role from that Actor upon all system objects. This endpoint requires ODK Central v0.5 or later.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#stripping-an-role-assignment-from-an-actor>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

assignment_revoke(role_id = "admin", actor_id = 14)
# > $success
# > [1] TRUE

## End(Not run)
```

attachment_delete	<i>Clear a Submission attachment.</i>
-------------------	---------------------------------------

Description

[Experimental]

Usage

```
attachment_delete(
  iid,
  filename,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
filename	(character) The name of the file as given by the attachment listing resource, e.g. "file1.jpg".
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Because Submission Attachments are completely determined by the XML data of the Submission itself, there is no direct way to entirely remove a Submission Attachment entry from the list, only to clear its uploaded content. Thus, a DELETE to the attachment's endpoint clears the uploaded bytes while the expected file slot remains listed. This endpoint clears attachments on the current version of the Submission.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-submission-management/#clearing-a-submission-attachment>

Other submission-management: [attachment_list\(\)](#), [attachment_upload\(\)](#), [encryption_key_list\(\)](#), [submission_audit_get\(\)](#), [submission_changes\(\)](#), [submission_comment_create\(\)](#), [submission_comment_list\(\)](#), [submission_create\(\)](#), [submission_csv\(\)](#), [submission_delete\(\)](#), [submission_detail\(\)](#), [submission_draft_attachment_download\(\)](#), [submission_draft_attachment_list\(\)](#), [submission_draft_attachment_upload\(\)](#).

```

submission_draft_create(), submission_draft_export(), submission_draft_get(), submission_draft_keys(),
submission_draft_list(), submission_edit(), submission_export(), submission_geodata(),
submission_geojson(), submission_get(), submission_list(), submission_restore(), submission_review(),
submission_submitters(), submission_update(), submission_version_attachment_delete(),
submission_version_attachment_download(), submission_version_attachment_list(), submission_version_de
submission_version_geojson(), submission_version_xml(), submission_versions()

```

Examples

```

## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()
iid <- sl$instance_id[[1]]

attachment_delete(iid, "photo.jpg")
# > $success
# > [1] TRUE

## End(Not run)

```

attachment_get

Download attachments and return the local path.

Description

[Stable]

Usage

```

attachment_get(
  sid,
  fn,
  local_dir = "media",
  separate = FALSE,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  verbose = get_ru_verbose()
)

```

Arguments

sid	One or many ODK submission UUIDs, an MD5 hash.
fn	One or many ODK form attachment filenames, e.g. "1558330537199.jpg".
local_dir	The local folder to save the downloaded files to, default: "media".
separate	(logical) Whether to separate locally downloaded files into a subfolder named after the submission uuid within local_dir, default: FALSE. The defaults mirror the behaviour of submission_export , which keeps all attachment files together in a folder media. Enable this option if downloaded files collide on identical names. This can happen if two data collection devices by chance generate the same filename for two respective media files, e.g. DCIM0001.jpg.
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through ru_setup(pid="..."). See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through ru_setup(fid="..."). See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through ru_setup(url="..."). See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through ru_setup(un="..."). See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through ru_setup(pw="..."). See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.
verbose	Whether to display debug messages or not. Read vignette("setup", package = "ruODK") to learn how ruODK's verbosity can be set globally or per function.

Details

This function is the workhorse for [handle_ru_attachments](#). This function is vectorised and can handle either one or many records. Parameters submission_uuid and attachment_filename accept single or exactly the same number of multiple values. The other parameters are automatically repeated.

The media attachments are downloaded into a folder given by local_dir:

```
workdir/media/filename1.jpg
workdir/media/filename2.jpg
workdir/media/filename3.jpg
```

Value

The relative file path for the downloaded attachment(s)

See Also

<https://docs.getodk.org/central-api-form-management/#downloading-a-form-attachment>

<https://docs.getodk.org/central-api-submission-management/#downloading-an-attachment>

Other utilities: `attachment_link()`, `attachment_url()`, `drop_null_coords()`, `form_schema_parse()`, `get_one_attachment()`, `get_one_submission()`, `get_one_submission_att_list()`, `get_one_submission_audit()`, `handle_ru_attachments()`, `handle_ru_datetimes()`, `handle_ru_geopoints()`, `handle_ru_geoshapes()`, `handle_ru_geotraces()`, `isodt_to_local()`, `odata_submission_rectangle()`, `predict_ruodk_name()`, `prepend_uuid()`, `ru_http_request()`, `split_geopoint()`, `split_geoshape()`, `split_geotrace()`, `strip_uuid()`, `tidyeval`, `unnest_all()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

a_local_dir <- here::here()

# Step 2: Get unparsed submissions
fresh_raw <- odata_submission_get(parse = FALSE)

# Step 3: Get attachment field "my_photo"
fresh_parsed <- fresh_raw %>%
  odata_submission_rectangle() %>%
  dplyr::mutate(
    my_photo = attachment_get(id,
      my_photo,
      local_dir = a_local_dir,
      verbose = TRUE
    )
    # Repeat for all other attachment fields
  )

## End(Not run)
```

attachment_link	<i>Prefix attachment columns from CSV export with a local attachment file path.</i>
-----------------	---

Description

[Stable]

Usage

```
attachment_link(data_tbl, form_schema, att_path = "media")
```

Arguments

<code>data_tbl</code>	The downloaded submissions from submission_export read into a tibble by <code>readr::read_csv</code> .
<code>form_schema</code>	The form_schema for the submissions. E.g. the output of <code>ruODK::form_schema()</code> .
<code>att_path</code>	A local path, default: "media" (as per .csv.zip export). Selected columns of the dataframe (containing attachment filenames) are prefixed with <code>att_path</code> , thus turning them into relative paths.

Value

The dataframe with attachment columns modified to contain relative paths to the downloaded attachment files.

See Also

Other utilities: [attachment_get\(\)](#), [attachment_url\(\)](#), [drop_null_coords\(\)](#), [form_schema_parse\(\)](#), [get_one_attachment\(\)](#), [get_one_submission\(\)](#), [get_one_submission_att_list\(\)](#), [get_one_submission_audit\(\)](#), [handle_ru_attachments\(\)](#), [handle_ru_datetimes\(\)](#), [handle_ru_geopoints\(\)](#), [handle_ru_geoshapes\(\)](#), [handle_ru_geotraces\(\)](#), [isodt_to_local\(\)](#), [odata_submission_rectangle\(\)](#), [predict_ruodk_name\(\)](#), [prepend_uuid\(\)](#), [ru_http_request\(\)](#), [split_geopoint\(\)](#), [split_geoshape\(\)](#), [split_geotrace\(\)](#), [strip_uuid\(\)](#), [tidyeval](#), [unnest_all\(\)](#)

Examples

```
## Not run:
t <- tempdir()
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# Predict filenames (with knowledge of form)
fid <- get_default_fid()
fid_csv <- fs::path(t, glue::glue("{fid}.csv"))
fid_csv_tae <- fs::path(t, glue::glue("{fid}-taxon-encounter.csv"))
fs <- form_schema()

# Download the zip file
se <- ruODK::submission_export(
  local_dir = t,
  overwrite = FALSE,
  verbose = TRUE
)

# Unpack the zip file
f <- unzip(se, exdir = t)
fs::dir_ls(t)
```

```
# Prepend attachments with media/ to turn into relative file paths
data_quadrat <- fid_csv %>%
  readr::read_csv(na = c("", "NA", "na")) %>%
  janitor::clean_names() %>%
  handle_ru_datetimes(fs) %>%
  attachment_link(fs)

## End(Not run)
```

attachment_list	<i>List all attachments for a list of submission instances.</i>
-----------------	---

Description

List all attachments for a list of submission instances.

Usage

```
attachment_list(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	A list of submission instance IDs, e.g. from <code>submission_list\$instance_id</code> .
pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code> . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: <code>get_default_fid</code> . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").

pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Value

A tibble containing some high-level details of the submission attachments. One row per submission attachment, columns are submission attributes:

- * name: The attachment filename, e.g. 12345.jpg
- * exists: Whether the attachment for that submission exists on the server.

See Also

<https://docs.getodk.org/central-api-submission-management/#listing-expected-submission-attachments>

<https://docs.getodk.org/central-api-form-management/#listing-form-attachments>

Other submission-management: `attachment_delete()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_attachment_upload()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# Step 1: Setup ruODK with OData Service URL (has url, pid, fid)
ruODK::ru_setup(svc = "...")

# Step 2: List all submissions of form
sl <- submission_list()

# Step 3a: Get attachment list for first submission
al <- get_one_submission_att_list(sl$instance_id[[1]])

# Ste 3b: Get all attachments for all submissions
all <- attachment_list(sl$instance_id)

## End(Not run)
```

attachment_upload	<i>Upload a Submission attachment.</i>
-------------------	--

Description**[Experimental]****Usage**

```
attachment_upload(
  iid,
  filename,
  file,
  content_type = NULL,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
filename	(character) The name of the file as given by the attachment listing resource, e.g. "file1.jpg".
file	(character) Path to a local file whose bytes are uploaded to the expected file slot.
content_type	(character) The MIME type sent as Content-Type header, e.g. "image/jpeg". Default: NULL, which sends "application/octet-stream".
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through ru_setup(pid="..."). See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through ru_setup(fid="..."). See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through ru_setup(url="..."). See vignette("Setup", package = "ruODK").

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

To upload a binary to an expected file slot, POST the binary to its endpoint. The expected file slots are determined by the Submission XML; list them with `attachment_list()`. filename must match a slot name from the attachment listing. file is the local file. Its bytes fill the slot.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-submission-management/#uploading-an-attachment>

Other submission-management: `attachment_delete()`, `attachment_list()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_attachment_upload()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()
iid <- sl$instance_id[[1]]

al <- attachment_list(iid)
al$name
# > "photo.jpg"

attachment_upload(iid, "photo.jpg", "/path/to/photo.jpg")
# > $success
# > [1] TRUE
```

```
## End(Not run)
```

audit_get	<i>Get server audit log entries.</i>
-----------	--------------------------------------

Description

[Stable]

Usage

```
audit_get(
  action = NULL,
  start = NULL,
  end = NULL,
  limit = NULL,
  offset = NULL,
  url = Sys.getenv("ODKC_URL"),
  un = Sys.getenv("ODKC_UN"),
  pw = Sys.getenv("ODKC_PW"),
  retries = get_retries()
)
```

Arguments

action	string. The action to filter the logs, e.g. "user.create". See https://docs.getodk.org/central-api-system-endpoints/#getting-audit-log-entries for the full list of available actions.
start	string. The ISO8601 timestamp of the earliest log entry to return. E.g. 2000-01-01z or 2000-12-31T23:59.999z, 2000-01-01T12:12:12+08 or 2000-01-01+08.
end	string. The ISO8601 timestamp of the last log entry to return.
limit	integer. The max number of log entries to return.
offset	integer. The number of log entries to skip.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Parameters to filter the audit logs: `action=form.create&start=2000-01-01z&end=2000-12-31T23%3A59.999z`

Value

A tibble containing server audit logs. One row per audited action, columns are submission attributes:

- `actor_id`: integer. The ID of the actor, if any, that initiated the action.
- `action`: string. The action that was taken.
- `actee_id`: uuid, string. The ID of the permissioning object against which the action was taken.
- `details`: list. Additional details about the action that vary according to the type of action.
- `logged_at`: dtm. Time of action on server.

See Also

<https://docs.getodk.org/central-api-system-endpoints/#getting-audit-log-entries>

Other server-management: [config_public\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

logs <- audit_get()

# With search parameters
logs <- audit_get(
  action = "project.update",
  start = "2019-08-01Z",
  end = "2019-08-31Z",
  limit = 100,
  offset = 0
)

# With partial search parameters
logs <- audit_get(
  limit = 100,
  offset = 0
)

logs %>% knitr::kable(.)

# audit_get returns a tibble
class(logs)
# > c("tbl_df", "tbl", "data.frame")

# Audit details
names(logs)
```

```
# > "actor_id" "action" "actee_id" "details" "logged_at"

## End(Not run)
```

config_public	<i>Show publicly accessible server configuration.</i>
---------------	---

Description

[Experimental]

Usage

```
config_public(
  url = get_default_url(),
  un = NULL,
  pw = NULL,
  retries = get_retries()
)
```

Arguments

url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	(character) The ODK Central username. Optional: the endpoint needs no authentication. Default: NULL (anonymous).
pw	(character) The ODK Central password. Optional: the endpoint needs no authentication. Default: NULL (anonymous).
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

This endpoint returns all server configuration that is publicly accessible in a single object. This endpoint does not require authentication. The following configurations are publicly accessible: login-appearance, logo and hero-image. For configuration that stores binary data (e.g. logo), this endpoint only returns metadata about the configuration. This endpoint requires ODK Central 2026.1 or later.

Value

A list with the public server configuration as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-system-endpoints/#getting-all-public-configuration>

Other server-management: [audit_get\(\)](#)

Examples

```
## Not run:
cfg <- config_public(url = "https://my.odkcentral.org")

names(cfg)

## End(Not run)
```

drop_null_coords

Drop any NULL coordinates from a GeoJSON geometry.

Description

This helper patches a bug/feature in ODK Central (versions 0.7-0.9), where geotrace / geoshape GeoJSON contains a last coordinate pair with NULL lat/lon (no alt/acc), and WKT ends in , undefined NaN.

Usage

```
drop_null_coords(x)
```

Arguments

x A GeoJSON geometry parsed as nested list. E.g. geo_gj\$path_location_path_gps.

Details

While [split_geotrace](#) and [split_geoshape](#) modify the WKT inline, it is more maintainable to separate the GeoJSON cleaner into this function.

This helper drops the last element of a GeoJSON coordinate list if it is `list(NULL, NULL)`.

Value

The nested list minus the last element (if NULL).

See Also

Other utilities: [attachment_get\(\)](#), [attachment_link\(\)](#), [attachment_url\(\)](#), [form_schema_parse\(\)](#), [get_one_attachment\(\)](#), [get_one_submission\(\)](#), [get_one_submission_att_list\(\)](#), [get_one_submission_audit\(\)](#), [handle_ru_attachments\(\)](#), [handle_ru_datetimes\(\)](#), [handle_ru_geopoints\(\)](#), [handle_ru_geoshapes\(\)](#), [handle_ru_geotraces\(\)](#), [isodt_to_local\(\)](#), [odata_submission_rectangle\(\)](#), [predict_ruodk_name\(\)](#), [prepend_uuid\(\)](#), [ru_http_request\(\)](#), [split_geopoint\(\)](#), [split_geoshape\(\)](#), [split_geotrace\(\)](#), [strip_uuid\(\)](#), [tidyeval](#), [unnest_all\(\)](#)

Examples

```
# A snapshot of geo data with trailing empty coordinates.
data("geo_gj88")

len_coords <- length(geo_gj88$path_location_path_gps[[1]]$coordinates)

length(geo_gj88$path_location_path_gps[[1]]$coordinates[[len_coords]]) %>%
  testthat::expect_equal(2)

geo_gj88$path_location_path_gps[[1]]$coordinates[[len_coords]][[1]] %>%
  testthat::expect_null()

geo_gj88$path_location_path_gps[[1]]$coordinates[[len_coords]][[2]] %>%
  testthat::expect_null()

# The last coordinate pair is a list(NULL, NULL).
# Invalid coordinates like these are a choking hazard for geospatial
# packages. We should remove them before we can convert ODK data into native
# spatial formats, such as sf.
str(geo_gj88$path_location_path_gps[[1]]$coordinates[[len_coords]])

geo_gj_repaired <- geo_gj88 %>%
  dplyr::mutate(
    path_location_path_gps = path_location_path_gps %>%
      purrr::map(drop_null_coords)
  )

len_coords_repaired <- length(
  geo_gj_repaired$path_location_path_gps[[1]]$coordinates
)
testthat::expect_equal(len_coords_repaired + 1, len_coords)
```

encryption_key_list *List all encryption keys for a form.*

Description**[Maturing]****Usage**

```
encryption_key_list(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
```

```
tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

This endpoint provides a listing of all known encryption keys needed to decrypt all Submissions for a given Form. It will return at least the `base64RsaPublicKey` property (as column `public`) of all known versions of the form that have submissions against them. If managed keys are being used and a hint was provided, that will be returned as well.

Value

A tibble of encryption keys.

See Also

<https://docs.getodk.org/central-api-encryption/>

<https://docs.getodk.org/central-api-submission-management/#listing-encryption-keys>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attach`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_delete()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

x <- encryption_key_list(
  pid = Sys.getenv("ODKC_TEST_PID_ENC"),
  fid = Sys.getenv("ODKC_TEST_FID_ENC"),
  url = get_test_url(),
  un = get_test_un(),
  pw = get_test_pw()
)

names(x)
# > [1] "id" "public" "managed" "hint" "created_at"

## End(Not run)
```

entity_audits

Return server audit logs of one Entity.

Description

[Maturing]

Usage

```
entity_audits(
  pid = get_default_pid(),
  did = "",
  eid = "",
  url = get_default_url(),
```

```

un = get_default_un(),
pw = get_default_pw(),
retries = get_retries(),
odkc_version = get_default_odkc_version(),
orders = get_default_orders(),
tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
eid	(chr) The UUID of an Entity, which can be retrieved by <code>entity_list()</code>. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code>'s timezone can be set globally or per function.

Details

This returns **Server Audit Logs** relating to an Entity. The most recent log is returned first.

The authenticated user must have permissions on ODK Central to retrieve audit logs.

Value

A tibble with one row per audit log. See <https://docs.getodk.org/central-api-entity-management/#entity-audit-log> for the full schema. Top level list elements are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-entity-management/#entity-audit-log>

Other entity-management: [entity_bulk_delete\(\)](#), [entity_bulk_restore\(\)](#), [entity_changes\(\)](#), [entity_create\(\)](#), [entity_creators\(\)](#), [entity_delete\(\)](#), [entity_detail\(\)](#), [entity_geodata\(\)](#), [entity_geojson\(\)](#), [entity_list\(\)](#), [entity_restore\(\)](#), [entity_update\(\)](#), [entity_versions\(\)](#), [entitylist_create\(\)](#), [entitylist_delete\(\)](#), [entitylist_detail\(\)](#), [entitylist_download\(\)](#), [entitylist_list\(\)](#), [entitylist_property_create\(\)](#), [entitylist_property_delete\(\)](#), [entitylist_trash_download\(\)](#), [entitylist_update\(\)](#), [odata_entitylist_data_get\(\)](#), [odata_entitylist_metadata_get\(\)](#), [odata_entitylist_service_get\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

# Entity List name (Dataset ID, did)
did <- el$name[1]

# All Entities of Entity List
en <- entity_list(did = did)

ed <- entity_detail(did = did, eid = en$uuid[1])

e_label <- ed$current_version$label

# This example updates one field which exists in the example form.
# Your own Entity will have different fields to update.
e_data <- list(
  details = paste0(
    ed$current_version$data$details, ". Updated on ", Sys.time()
  )
)

# Update the Entity (implicitly forced update)
eu <- entity_update(
  did = did,
```

```

    eid = en$uuid[1],
    label = e_label,
    data = e_data
  )

  # Return the server audit logs
  ea <- entity_audits(did = did, eid = en$uuid[1])
  ea

  ## End(Not run)

```

entity_bulk_delete *Delete multiple Entities at once.*

Description

[Experimental]

Usage

```

entity_bulk_delete(
  pid = get_default_pid(),
  did = "",
  eids = c(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
eids	(character) The UUIDs of the Entities to delete.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code> 's timezone can be set globally or per function.

Details

This endpoint accepts an array of Entity UUIDs and performs soft-deletion on all of them.

Value

A list with a single element count holding the number of Entities that the bulk operation deleted or restored.

See Also

<https://docs.getodk.org/central-api-entity-management/#bulk-deleting-entities>

Other entity-management: `entity_audits()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")
```

```

el <- entitylist_list()
did <- el$name[1]

en <- entity_list(did = did)

entity_bulk_delete(did = did, eids = en$uuid[1:2])
# > $count
# > [1] 2

## End(Not run)

```

entity_bulk_restore *Restore multiple deleted Entities at once.*

Description

[Experimental]

Usage

```

entity_bulk_restore(
  pid = get_default_pid(),
  did = "",
  eids = c(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
eids	(character) The UUIDs of the deleted Entities to restore.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code> 's timezone can be set globally or per function.

Details

This endpoint accepts an array of Entity UUIDs and restores all of them that were previously deleted.

Value

A list with a single element count holding the number of Entities that the bulk operation deleted or restored.

See Also

<https://docs.getodk.org/central-api-entity-management/#bulk-restoring-entities>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")
```

```

el <- entitylist_list()
did <- el$name[1]

en <- entity_list(did = did)

entity_bulk_restore(did = did, eids = en$uuid[1:2])
# > $count
# > [1] 2

## End(Not run)

```

entity_changes

List changes to one Entity.

Description

[Maturing]

Usage

```

entity_changes(
  pid = get_default_pid(),
  did = "",
  eid = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
eid	(chr) The UUID of an Entity, which can be retrieved by <code>entity_list()</code>. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code> 's timezone can be set globally or per function.

Details

This returns the changes, or edits, between different versions of an Entity. These changes are returned as a list of lists. Between two Entities, there is a list of objects representing how each property changed. This change object contains the old and new values, as well as the property name.

Value

A tibble where rows correspond to each Entity update with three columns:

- old old value
- new new value
- propertyName name of changed Entity property

See Also

<https://docs.getodk.org/central-api-entity-management/#getting-changes-between-versions>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

# Entity List name (Dataset ID, did)
did <- el$name[1]

# All Entities of Entity List
en <- entity_list(did = did)

ed <- entity_detail(did = did, eid = en$uuid[1])

e_label <- ed$current_version$label

# This example updates one field which exists in the example form.
# Your own Entity will have different fields to update.
e_data <- list(
  details = paste0(
    ed$current_version$data$details, ". Updated on ", Sys.time()
  )
)

# Update the Entity (implicitly forced update)
eu <- entity_update(
  did = did,
  eid = en$uuid[1],
  label = e_label,
  data = e_data
)

ec <- entity_changes(did = did, eid = en$uuid[1])
ec

## End(Not run)
```

entity_create

Creates exactly one Entity in the Dataset.

Description**[Experimental]****Usage**

```
entity_create(
  pid = get_default_pid(),
```

```

    did = "",
    label = "",
    uuid = "",
    notes = "",
    data = list(),
    url = get_default_url(),
    un = get_default_un(),
    pw = get_default_pw(),
    retries = get_retries(),
    odkc_version = get_default_odkc_version(),
    orders = get_default_orders(),
    tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
label	(character) The Entity label which must be a non-empty string. If the label is given, a single Entity is created using data, notes, and uuid if given. If the label is kept at the default (or omitted), multiple Entities are created using data and notes and ignoring uuid. Default: "".
uuid	(character) A single UUID to assign to the Entity. Default: "". With the default, Central will create and assign a UUID. This parameter is only used when creating a single Entity (label non-empty) and ignored when creating multiple Entities (label empty).
notes	(character) Metadata about the request which can be retrieved using the Entity audit log. Default: "".
data	(list) A named list of Entity properties to create a single Entity, or a nested list with an array of Entity data to create multiple Entities. The nested lists representing individual Entities must be valid as in they must contain a label, valid data for the respective Entity properties, and can contain an optional UUID. See details and the ODK documentation for the exact format. Default: <code>list()</code>.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

Creating a single Entity:

For creating a single Entity, include

- An optional uuid. If skipped, Central will create a UUID for the Entity.
- The Entity label must be non-empty.
- A named list data representing the Entity's fields. The value type of all properties is string.

```
uuid = "... " label = "John Doe" data = list("firstName" = "John", "age" = "22")
```

This translates to JSON

```
{ "label": "John Doe", "data": { "firstName": "John", "age": "22" } }
```

Creating multiple Entities:

For creating multiple Entities in bulk, the request body takes an array `entities` containing a list of Entity objects as described above. The bulk Entity version also takes a source property with a required name field and optional size, for example to capture the file name and size of a bulk upload source (in MB).

```
data=list(
  "entities" = c(
    list("label" = "Entity 1", "data" = list("field1" = "value1")),
    list("label" = "Entity 2", "data" = list("field1" = "value2"))
  ),
  "source" = list("name" = "file.csv", "size" = 100)
)
```

This translates to JSON

```
{ "entities": [...], "source": { "name": "file.csv", "size": 100} }
```

You can provide notes to store the metadata about the request. The metadata is included in the POST request as header `X-Action-Notes` and can be retrieved using Entity Audit Log.

Value

A nested list identical to the return value of `entity_detail`. See <https://docs.getodk.org/central-api-entity-management/#creating-entities> for the full schema. Top level list elements are renamed from ODK's camelCase to snake_case. Nested list elements have the original camelCase.

See Also

<https://docs.getodk.org/central-api-entity-management/#creating-entities>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

# Entity List name (Dataset ID, did)
did <- el$name[1]

# All Entities of Entity List
en <- entity_list(did = did)

# Create a single Entity
ec <- entity_create(
  did = did,
  label = "Entity label",
  notes = "Metadata about the created Entity",
  data = list("field1" = "value1", "field2" = "value1")
)
ec

# Create multiple Entities, example: test form "problems"
label <- c(
  glue::glue(
    "Entity {nrow(en) + 1} created by ruODK package test on {Sys.time()}"
  ),
  glue::glue(
    "Entity {nrow(en) + 2} created by ruODK package test on {Sys.time()}"
  )
)
notes <- glue::glue("Two Entities created by ruODK package test on {Sys.time()}")
status <- c("needs_followup", "needs_followup")
```

```

details <- c("ruODK package test", "ruODK package test")
geometry <- c("-33.2 115.0 0.0 0.0", "-33.2 115.0 0.0 0.0")
data <- data.frame(status, details, geometry, stringsAsFactors = FALSE)
request_data <- list(
  "entities" = data.frame(label, data = I(data), stringsAsFactors = FALSE),
  "source" = list("name" = "file.csv", "size" = 100)
)

# Compare request_data to the schema expected by Central
# https://docs.getodk.org/central-api-entity-management/#creating-entities
# jsonlite::toJSON(request_data, pretty = TRUE, auto_unbox = TRUE)

ec <- entity_create(
  did = did,
  notes = notes,
  data = request_data
)

## End(Not run)

```

entity_creators

List all creators of Entities in one Entity List.

Description

[Experimental]

Usage

```

entity_creators(
  pid = get_default_pid(),
  did = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)

```

Arguments

pid The numeric ID of the project, e.g.: 2.
 Default: [get_default_pid](#).
 Set default pid through `ru_setup(pid="...")`.
 See `vignette("Setup", package = "ruODK")`.

did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

This endpoint returns a list of all Actors who have created Entities in the Dataset, sorted by display name.

Value

A tibble with one row per creating Actor and all Actor metadata as columns as per the ODK Central API docs. Column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-entity-management/#entities-creators>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

cr <- entity_creators(did = el$name[1])

cr |> knitr::kable()

## End(Not run)
```

entity_delete	<i>Delete one Entity.</i>
---------------	---------------------------

Description**[Maturing]****Usage**

```
entity_delete(
  pid = get_default_pid(),
  did = "",
  eid = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
eid	(chr) The UUID of an Entity, which can be retrieved by <code>entity_list()</code> . The function will error if this parameter is not given. Default: "".

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

This function soft-deletes one Entity, , which means it is still in Central's database and you can retrieve it via `entity_list(deleted=TRUE)`.

Value

A list with the key "success" (lgl) indicating whether the Entity was deleted.

See Also

<https://docs.getodk.org/central-api-entity-management/#deleting-an-entity>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

# Entity List name (Dataset ID, did)
did <- el$name[1]

# All Entities of Entity List
en <- entity_list(did = did)

# View details of one Entity
ed <- entity_detail(did = did, eid = en$uuid[1])

# Delete the Entity
ed_deleted <- entity_delete(did = did, eid = ed$id)

## End(Not run)
```

entity_detail

*Show metadata and current data of one Entity.***Description****[Maturing]****Usage**

```
entity_detail(
  pid = get_default_pid(),
  did = "",
  eid = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid The numeric ID of the project, e.g.: 2.
 Default: [get_default_pid](#).
 Set default pid through `ru_setup(pid="...")`.
 See `vignette("Setup", package = "ruODK")`.

did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
eid	(chr) The UUID of an Entity, which can be retrieved by <code>entity_list()</code> . The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code> 's timezone can be set globally or per function.

Details

This function returns the metadata and current data of an Entity.

The data and label of an Entity are found in the `current_version` columns under `data` and `label` respectively. The `current_version` column contains version-specific metadata fields such as `version`, `baseVersion`, details about conflicts that version introduced, etc. More details are available from `entity_versions()`. The Entity also contains top-level system metadata such as `uuid`, `created_at` and `updated_at` timestamps, and `creator_id` (or full creator if extended metadata is requested).

Conflicts:

An Entity's metadata also has a `conflict` field, which indicates the current conflict status of the Entity. The value of a conflict can either be:

- `null`. The Entity does not have conflicting versions.
- `soft`. The Entity has a version that was based on a version other than the version immediately prior to it. (The specified `base_version` was not the latest version on the server.)

- hard. The Entity has a version that was based on a version other than the version immediately prior to it. Further, that version updated the same property as another update.

If an Entity has a conflict, it can be marked as resolved. After that, the conflict field will be null until a new conflicting version is received.

Value

A nested list identical to the return value of `entity_update`. See <https://docs.getodk.org/central-api-entity-management/#getting-entity-details> for the full schema. Top level list elements are renamed from ODK's camelCase to snake_case. Nested list elements have the original camelCase.

See Also

<https://docs.getodk.org/central-api-entity-management/#entities-metadata>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

# Entity List name (Dataset ID, did)
did <- el$name[1]

# All Entities of Entity List
en <- entity_list(did = did)

ed <- entity_detail(did = did, eid = en$uuid[1])

# The current version of the first Entity
ev <- en$current_version_version[1]

## End(Not run)
```

entity_geodata	<i>Get Entity geodata as GeoJSON.</i>
----------------	---------------------------------------

Description

[Experimental]

Usage

```
entity_geodata(
  pid = get_default_pid(),
  did = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See vignette("Setup", package = "ruODK").
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>.

	Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>orders</code>	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
<code>tz</code>	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

This endpoint returns the geometry-interpretable value of the geometry attribute of Entities of a Dataset, in GeoJSON format. No particular ordering is implied.

Value

A list with the GeoJSON FeatureCollection as returned by Central: type and features with geometry and properties per Entity.

See Also

<https://docs.getodk.org/central-api-entity-management/#entities-geodata>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

g <- entity_geodata(did = el$name[1])

g$type
# > "FeatureCollection"

## End(Not run)
```

entity_geojson	<i>Get the GeoJSON of one Entity.</i>
----------------	---------------------------------------

Description

[Experimental]

Usage

```
entity_geojson(
  pid = get_default_pid(),
  did = "",
  eid = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
eid	(chr) The UUID of an Entity, which can be retrieved by <code>entity_list()</code>. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.

odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page /version.txt. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

Returns the GeoJSON representation of an Entity's geometry attribute, if available.

Value

A list with the GeoJSON FeatureCollection holding the Entity's geometry.

See Also

<https://docs.getodk.org/central-api-entity-management/#entity-geojson-representation>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()
did <- el$name[1]

en <- entity_list(did = did)

g <- entity_geojson(did = did, eid = en$uuid[1])

g$type
# > "FeatureCollection"

## End(Not run)
```

entity_list	<i>List all Entities of a kind.</i>
-------------	-------------------------------------

Description

[Maturing]

Usage

```
entity_list(
  pid = get_default_pid(),
  did = "",
  deleted = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
deleted	(lgl) Whether to get only deleted Entities (TRUE) or not (FALSE). Default: FALSE.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page /version.txt. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

This function returns a list of the Entities of a kind (belonging to an Entity List or Dataset). Please note that this endpoint only returns metadata of the Entities, not the data. If you want to get the data of all Entities then please refer to the OData Dataset Service.

You can get only deleted Entities with `deleted=TRUE`.

Value

A tibble with exactly one row for each Entity of the given project as per ODK Central API docs. Column names are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-entity-management/#entities-metadata>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

# Entity List name (Dataset ID)
did <- el$name[1]

# All Entities of Entity List
```

```

en <- entity_list(did = e1$name[1])

# The UUID of the first Entity
eid <- en$uuid[1]

# The current version of the first Entity
ev <- en$current_version_version[1]

## End(Not run)

```

entity_restore	<i>Restore a deleted Entity.</i>
----------------	----------------------------------

Description

[Experimental]

Usage

```

entity_restore(
  pid = get_default_pid(),
  did = "",
  eid = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
eid	(chr) The UUID of an Entity, which can be retrieved by <code>entity_list()</code>. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code> 's timezone can be set globally or per function.

Details

Entities that have been recently soft-deleted and not yet purged can be restored with this endpoint.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-entity-management/#restoring-a-deleted-entity>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()
```

```

did <- el$name[1]

en <- entity_list(did = did)

entity_delete(did = did, eid = en$uuid[1])

entity_restore(did = did, eid = en$uuid[1])
# > $success
# > [1] TRUE

## End(Not run)

```

entity_update	<i>Update one Entity.</i>
---------------	---------------------------

Description

[Experimental]

Usage

```

entity_update(
  pid = get_default_pid(),
  did = "",
  eid = "",
  label = "",
  data = list(),
  base_version = NULL,
  force = is.null(base_version),
  resolve = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".

eid	(chr) The UUID of an Entity, which can be retrieved by <code>entity_list()</code> . The function will error if this parameter is not given. Default: "".
label	(character) The Entity label which must be a non-empty string. Default: "".
data	(list) A named list of Entity properties to update. See details. Default: <code>list()</code> .
base_version	If given, must be the current version of the Entity on the server. Optional.
force	(lgl) Whether to force an update. Defaults to be FALSE if a <code>base_version</code> is specified, else defaults to TRUE. Using <code>force=TRUE</code> and specifying a <code>base_version</code> will emit a warning.
resolve	(lgl)
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code> 's timezone can be set globally or per function.

Details

This endpoint is used to update the label or the properties (passed as JSON in the request body) of an Entity. You only need to include the properties you wish to update. To unset the value of any property, you can set it to empty string (""). The label must be a non-empty string. Setting a property to null will throw an error. Attempting to update a property that doesn't exist in the Dataset will throw an error.

Specifying a base version:

You must either provide a `base_version` or use `force=TRUE` query parameter. You cannot cause a new Entity conflict via the API, which is why when specifying `base_version`, it must match the current version of the Entity on the server. This acts as a check to ensure you are not trying to update based on stale data. If you wish to update the Entity regardless of the current state, then you can use the force flag.

Resolving a conflict:

You can also use this endpoint to resolve an Entity conflict by passing `resolve=true`, in which case providing data is optional. When not providing new data, only the conflict status from the Entity will be cleared and no new version will be created. When providing data, the conflict will be cleared and an updated version of the Entity will be added.

Value

A nested list identical to the return value of `entity_detail`. See <https://docs.getodk.org/central-api-entity-management/#updating-an-entity> for the full schema. Top level list elements are renamed from ODK's camelCase to snake_case. Nested list elements have the original camelCase.

See Also

<https://docs.getodk.org/central-api-entity-management/#updating-an-entity>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

# Entity List name (Dataset ID, did)
did <- el$name[1]

# All Entities of Entity List
en <- entity_list(did = did)

ed <- entity_detail(did = did, eid = en$uuid[1])

e_label <- ed$current_version$label

# This example updates one field which exists in the example form.
```

```

# Your own Entity will have different fields to update.
e_data <- list(
  details = paste0(
    ed$current_version$data$details, ". Updated on ", Sys.time()
  )
)

# Update the Entity (implicitly forced update)
eu <- entity_update(
  did = did,
  eid = en$uuid[1],
  label = e_label,
  data = e_data
)
eu

## End(Not run)

```

entity_versions	<i>List versions of one Entity.</i>
-----------------	-------------------------------------

Description

[Maturing]

Usage

```

entity_versions(
  pid = get_default_pid(),
  did = "",
  eid = "",
  conflict = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
-----	--

did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
eid	(chr) The UUID of an Entity, which can be retrieved by <code>entity_list()</code> . The function will error if this parameter is not given. Default: "".
conflict	(lgl) Whether to return all versions (FALSE) or whether to returns the subset of past versions of an Entity that are relevant to the Entity's current conflict (TRUE). This includes the latest version, the base version, the previous server version, and any other versions since the last time the Entity was in a conflict-free state. If the Entity is not in conflict, zero versions are returned. Default: FALSE (return all versions)
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code> 's timezone can be set globally or per function.

Details

This returns the Entity metadata and data for every version of this Entity in ascending creation order. The ODK Central endpoint supports retrieving extended metadata which this function always returns.

There is an optional query flag `relevantToConflict` that returns the subset of past versions of an Entity that are relevant to the Entity's current conflict. This includes the latest version, the base version, the previous server version, and any other versions since the last time the Entity was in a conflict-free state. If the Entity is not in conflict, zero versions are returned.

Value

A tibble with one row per version. List columns contain unstructured data. See <https://docs.getodk.org/central-api-entity-management/#listing-versions> for the full schema. Top level list elements are renamed from ODK's camelCase to snake_case. Nested list elements have the original camelCase.

See Also

<https://docs.getodk.org/central-api-entity-management/#listing-versions>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_list()

# Entity List name (Dataset ID, did)
did <- el$name[1]

# All Entities of Entity List
en <- entity_list(did = did)

ed <- entity_detail(did = did, eid = en$uuid[1])

# The current version of the first Entity
ev <- en$current_version_version[1]

## End(Not run)
```

entitylist_create

Create an Entity List.

Description

[Experimental]

Usage

```
entitylist_create(
  pid = get_default_pid(),
  name = "",
  approval_required = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
name	(character) The desired name of the Entity List. The name must follow the same rules as XML identifiers and must not start with . or __.
approval_required	(lgl) Control whether a Submission should be approved before an Entity is created from it. Default: FALSE.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

orders (vector of character) Orders of datetime elements for lubridate.
 Default: `c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")`.

tz A timezone to convert dates and times to.
 Read `vignette("setup", package = "ruODK")` to learn how ruODK's timezone can be set globally or per function.

Details

You can create a Dataset with a specific name within a Project. This Dataset can then be populated with Entities via the API or via Forms and Submissions, and then used by other Forms. This endpoint allows a Dataset to be created programmatically without an input form.

The name of a Dataset is case-sensitive in that it will keep the capitalization provided (e.g. "Trees"). But Central will not allow a second Dataset with the same name but different capitalization to be created (e.g. "trees" when "Trees" already exists).

By default, the Dataset will have no properties, but each Entity will have a label and a unique ID (uuid).

This creates an empty Entity List. Add properties with the Dataset properties endpoint. Add Entities with `entity_create()`. Or publish a Form that defines the Entity List schema.

Value

A list of lists following the exact format and naming of the API response for `entitylist_detail`.

See Also

<https://docs.getodk.org/central-api-dataset-management/#creating-datasets>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

el <- entitylist_create(name = "Trees")

el$approvalRequired
# > FALSE

## End(Not run)
```

entitylist_delete	<i>Delete an Entity List.</i>
-------------------	-------------------------------

Description

[Experimental]

Usage

```
entitylist_delete(
  pid = get_default_pid(),
  did = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> .

Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central.
 Default: `get_default_odkc_version` or "2023.5.1" if unset.
 Set default `get_default_odkc_version` through `ru_setup(odkc_version="2023.5.1")`.
 See `vignette("Setup", package = "ruODK")`.

orders (vector of character) Orders of datetime elements for lubridate.
 Default: `c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")`.

tz A timezone to convert dates and times to.
 Read `vignette("setup", package = "ruODK")` to learn how ruODK's timezone can be set globally or per function.

Details

When a Dataset is deleted, it is soft-deleted and placed in the trash. After 30 days in the trash, the Dataset and all of its Entities are automatically purged. Unlike Forms, deleted Datasets cannot be restored. A Dataset cannot be deleted while any Forms still use it, either as a source of Entities or as a consumer through an attachment. Update or delete those Forms first. This endpoint requires ODK Central 2026.1 or later.

Value

A list with a single element `success` (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-dataset-management/#deleting-a-dataset>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

entitylist_delete(did = "trees")
# > $success
# > [1] TRUE

## End(Not run)
```

entitylist_detail	<i>Show Entity List details.</i>
-------------------	----------------------------------

Description

[Maturing]

Usage

```
entitylist_detail(
  pid = get_default_pid(),
  did = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> .

	Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central.
	Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset.
	Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> .
	See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate.
	Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to.
	Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Value

A list of lists following the exact format and naming of the API response. Since this nested list is so deeply nested and irregularly shaped it is not trivial to rectangle the result into a tibble.

See Also

<https://docs.getodk.org/central-api-dataset-management/#datasets>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

ds <- entitylist_list(pid = get_default_pid())

ds1 <- entitylist_detail(pid = get_default_pid(), did = ds$name[1])

ds1 |> listviewer::jsonedit()

ds1$linkeds_forms |>
  purrr::list_transpose() |>
  tibble::as_tibble()

ds1$source_forms |>
  purrr::list_transpose() |>
  tibble::as_tibble()

ds1$properties |>
  purrr::list_transpose() |>
```



```
tibble::as_tibble()

## End(Not run)
```

entitylist_download *Download an Entity List as CSV.*

Description

[Maturing]

Usage

```
entitylist_download(
  pid = get_default_pid(),
  did = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  local_dir = here::here(),
  filter = NULL,
  etag = NULL,
  overwrite = TRUE,
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz(),
  verbose = get_ru_verbose()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
local_dir	The local folder to save the downloaded files to, default: <code>here::here</code> . If the folder does not exist it will be created.
filter	(str) A valid filter string. Default: NULL (no filtering, all Entities returned).
etag	(str) The etag value from a previous call to <code>entitylist_download()</code> . The value must be stripped of the <code>W/\</code> and <code>\</code> , which is the format of the etag returned by <code>entitylist_download()</code> . If provided, only new Entities will be returned. If the same <code>local_dir</code> is chosen and <code>overwrite</code> is set to <code>TRUE</code> , the downloaded CSV will also be overwritten, losing the previously downloaded Entities. Default: NULL (no filtering, all Entities returned).
overwrite	Whether to overwrite previously downloaded file, default: <code>FALSE</code>
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See vignette("Setup", package = "ruODK").
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read vignette("setup", package = "ruODK") to learn how <code>ruODK</code> 's timezone can be set globally or per function.
verbose	Whether to display debug messages or not. Read vignette("setup", package = "ruODK") to learn how <code>ruODK</code> 's verbosity can be set globally or per function.

Details

CSV file:

The downloaded CSV file is named after the Entity List name. The download location defaults to the current workdir, but can be modified to a different folder path which will be created if it doesn't exist.

Entity Lists can be used as Attachments in other Forms, but they can also be downloaded directly as a CSV file.

The CSV format closely matches the OData Dataset (Entity List) Service format, with columns for system properties such as `__id` (the Entity UUID), `__createdAt`, `__creatorName`, etc., the Entity Label, and the Dataset (Entity List) or Entity Properties themselves. If any Property for an given Entity is blank (e.g. it was not captured by that Form or was left blank), that field of the CSV is blank.

Filter:

The ODK Central \$filter query string parameter can be used to filter on system-level properties, similar to how filtering in the OData Dataset (Entity List) Service works. Of the [OData filter specs](#)

ODK Central implements a [growing set of features](#). ruODK provides the parameter filter (str) which, if set, will be passed on to the ODK Central endpoint as is.

Resuming downloads through ETag:

The ODK Central endpoint supports the [ETag header](#), which can be used to avoid downloading the same content more than once. When an API consumer calls this endpoint, the endpoint returns a value in the ETag header. If you pass that value in the [If-None-Match header](#) of a subsequent request, then if the Entity List has not been changed since the previous request, you will receive 304 Not Modified response; otherwise you'll get the new data. ruODK provides the parameter etag which can be set from the output of a previous call to entitylist_download(). ruODK strips the W/\ and \" from the returned etag and expects the stripped etag as parameter.

Value

A list of four items:

- entities (tbl_df) The Entity List as tibble
- http_status (int) The HTTP status code of the response. 200 if OK, 304 if a given etag finds no new Entities created.
- etag (str) The ETag to use in subsequent calls to entitylist_download()
- downloaded_to (fs_path) The path to the downloaded CSV file
- downloaded_on (POSIXct) The time of download in the local timezone

See Also

<https://docs.getodk.org/central-api-dataset-management/#datasets>

Other entity-management: [entity_audits\(\)](#), [entity_bulk_delete\(\)](#), [entity_bulk_restore\(\)](#), [entity_changes\(\)](#), [entity_create\(\)](#), [entity_creators\(\)](#), [entity_delete\(\)](#), [entity_detail\(\)](#), [entity_geodata\(\)](#), [entity_geojson\(\)](#), [entity_list\(\)](#), [entity_restore\(\)](#), [entity_update\(\)](#), [entity_versions\(\)](#), [entitylist_create\(\)](#), [entitylist_delete\(\)](#), [entitylist_detail\(\)](#), [entitylist_list\(\)](#), [entitylist_property_create\(\)](#), [entitylist_property_delete\(\)](#), [entitylist_trash_download\(\)](#), [entitylist_update\(\)](#), [odata_entitylist_data_get\(\)](#), [odata_entitylist_metadata_get\(\)](#), [odata_entitylist_service_get\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

ds <- entitylist_list(pid = get_default_pid())

ds1 <- entitylist_download(pid = get_default_pid(), did = ds$name[1])
# ds1$entities
# ds1$etag
# ds1$downloaded_to
```

```

# ds1$downloaded_on

ds2 <- entitylist_download(
  pid = get_default_pid(),
  did = ds$name[1],
  etag = ds1$etag
)
# ds2$http_status == 304

newest_entity_date <- as.Date(max(ds1$entities$`__createdAt`))
ds3 <- entitylist_download(
  pid = get_default_pid(),
  did = ds$name[1],
  filter = glue::glue("__createdAt le {newest_entity_date}")
)

## End(Not run)

```

entitylist_list	<i>List all Entity Lists of one Project.</i>
-----------------	--

Description

[Maturing]

Usage

```

entitylist_list(
  pid = get_default_pid(),
  deleted = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
deleted	(lgl) If TRUE, list only deleted Entity Lists with their numeric IDs. Default: FALSE.

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

The returned list is useful to retrieve the valid name of an Entity List for further use by functions of the Entity Management family.

With `deleted = TRUE`, only deleted Entity Lists with their numeric IDs are returned, which can be used to export a deleted Entity List with `entitylist_trash_download()`.

Value

A tibble with exactly one row for each Entity List of the given Project as per ODK Central API docs. Column names are renamed from ODK Central's camelCase to snake_case. With `deleted = TRUE`, the tibble holds one row per deleted Entity List with its numeric ID.

See Also

<https://docs.getodk.org/central-api-dataset-management/#datasets>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`.

```
entitylist_property_create(), entitylist_property_delete(), entitylist_trash_download(),
entitylist_update(), odata_entitylist_data_get(), odata_entitylist_metadata_get(),
odata_entitylist_service_get()
```

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "... .svc", un = "me@email.com", pw = "...")

ds <- entitylist_list(pid = get_default_pid())

ds |> knitr::kable()

## End(Not run)
```

entitylist_property_create

Add a property to an Entity List.

Description

[Experimental]

Usage

```
entitylist_property_create(
  pid = get_default_pid(),
  did = "",
  property = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code> . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".

property	(character) The desired name of the Property.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

Creates a new Property with a specified name in the Dataset. The name of a Property is case-sensitive in that it will keep the capitalization provided (e.g. "Firstname"). But Central will not allow another Property with the same name but different capitalization to be created (e.g. "FIRST-NAME" when "Firstname" already exists). Property names follow the same rules as form field names (valid XML identifiers) and cannot use the reserved names of name or label, or begin with the reserved prefix `__`.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-dataset-management/#adding-properties>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`,

```
entity_versions(), entitylist_create(), entitylist_delete(), entitylist_detail(), entitylist_download(),
entitylist_list(), entitylist_property_delete(), entitylist_trash_download(), entitylist_update(),
odata_entitylist_data_get(), odata_entitylist_metadata_get(), odata_entitylist_service_get()
```

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

entitylist_property_create(did = "trees", property = "circumference")
# > $success
# > [1] TRUE

## End(Not run)
```

entitylist_property_delete

Delete a property from an Entity List.

Description

[Experimental]

Usage

```
entitylist_property_delete(
  pid = get_default_pid(),
  did = "",
  property = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code> . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".

property	(character) The name of the Property to delete.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page /version.txt. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

Deletes a published Property from a Dataset. The deletion is permanent, but a new Property with the same name can be added again. A Property cannot be deleted if any published or draft Forms write to it, or if any Entities hold a non-empty value for it. This endpoint requires ODK Central 2026.1 or later.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-dataset-management/#deleting-a-property>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

entitylist_property_delete(did = "trees", property = "circumference")
# > $success
# > [1] TRUE

## End(Not run)
```

entitylist_trash_download

Download Entities of a deleted Entity List.

Description

[Experimental]

Usage

```
entitylist_trash_download(
  pid = get_default_pid(),
  dataset_id,
  local_dir = tempdir(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz(),
  verbose = get_ru_verbose()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
dataset_id	The numeric ID of the deleted Dataset, e.g. from <code>entitylist_list(deleted = TRUE)</code> .
local_dir	The local folder to save the downloaded CSV to. If the folder does not exist it is created. Default: <code>tempdir()</code> .

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

Returns a CSV snapshot of the Entities belonging to a soft-deleted Dataset. This exports Entity data before the Dataset is permanently purged after 30 days in the trash. The CSV format is identical to `entitylist_download()`, except the filename carries a `-deleted` suffix. This endpoint uses the numeric Dataset ID rather than the Dataset name, because the name alone cannot identify a deleted Dataset unambiguously. The numeric ID comes from listing Datasets with `entitylist_list(deleted = TRUE)`. This endpoint requires ODK Central 2026.1 or later.

Value

A list of three items:

- `entities` (data.frame) The deleted Entities as parsed from CSV.
- `http_status` (int) The HTTP status code of the response.
- `downloaded_to` (fs_path) The path to the downloaded CSV file.

See Also

<https://docs.getodk.org/central-api-dataset-management/#downloading-deleted-dataset-entities>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

trash <- entitylist_list(deleted = TRUE)

dl <- entitylist_trash_download(dataset_id = trash$id[[1]])

dl$entities |> knitr::kable()

## End(Not run)
```

entitylist_update	<i>Update Entity List details.</i>
-------------------	------------------------------------

Description

[Maturing]

Usage

```
entitylist_update(
  pid = get_default_pid(),
  did = "",
  approval_required = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code>. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
approval_required	(lgl) The value to set <code>approvalRequired</code> to. If TRUE, a Submission must be approved before an Entity is created, if FALSE, an Entity is created as soon as the Submission is received by ODK Central. Default: FALSE.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code>. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: <code>get_default_un</code>. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: <code>get_default_pw</code>. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

You can only update `approvalRequired` using this endpoint. The `approvalRequired` flag controls the Entity creation flow; if it is true then the Submission must be approved before an Entity can be created from it and if it is false then an Entity is created as soon as the Submission is received by the ODK Central. By default `approvalRequired` is false for the Entity Lists created after v2023.3. Entity Lists created prior to that will have `approvalRequired` set to true.

Value

A list of lists following the exact format and naming of the API response for `entitylist_detail`. Since this nested list is so deeply nested and irregularly shaped it is not trivial to rectangle the result into a tibble.

See Also

<https://docs.getodk.org/central-api-dataset-management/#datasets>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

pid <- get_default_pid()

ds <- entitylist_list(pid = pid)

did <- ds$name[1]

ds1 <- entitylist_detail(pid = pid, did = did)
ds1$approvalRequired # FALSE

ds2 <- entitylist_update(pid = pid, did = did, approval_required = TRUE)
ds2$approvalRequired # TRUE

ds3 <- entitylist_update(pid = pid, did = did, approval_required = FALSE)
ds3$approvalRequired # FALSE

## End(Not run)
```

form_assignment_actors

List all Actors assigned some Form Role.

Description

[Experimental]

Usage

```

form_assignment_actors(
  pid = get_default_pid(),
  fid = get_default_fid(),
  role_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
role_id	(character or numeric) The Role ID, typically the integer ID of the Role. A Role system name can also be supplied if the Role has one.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

orders (vector of character) Orders of datetime elements for lubridate.
 Default: `c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")`.

tz A timezone to convert dates and times to.
 Read `vignette("setup", package = "ruODK")` to learn how ruODK's timezone can be set globally or per function.

Details

Given a Role, this endpoint lists all Actors that have been assigned that Role upon this particular Form.

Value

A tibble with one row for each Actor assigned the Role on the Form as per the ODK Central API docs. Column names are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-form-management/#listing-all-actors-assigned-some-form-role>

Other form-management: `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

aa <- form_assignment_actors(
  fid = fl$fid[[1]],
  role_id = "manager"
)

aa |> knitr::kable()

## End(Not run)
```

form_assignment_grant *Assign an Actor to a Form Role.*

Description

[Experimental]

Usage

```
form_assignment_grant(
  pid = get_default_pid(),
  fid = get_default_fid(),
  role_id,
  actor_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
role_id	(character or numeric) The Role ID, typically the integer ID of the Role. A Role system name can also be supplied if the Role has one.
actor_id	(numeric) The integer ID of the Actor.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Given a Role and an Actor, this endpoint assigns that Role to that Actor for this particular Form.
No request body data is required.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#assigning-an-actor-to-a-form-role>

Other form-management: `form_assignment_actors()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

form_assignment_grant(
  fid = fl$fid[[1]],
  role_id = "manager",
  actor_id = 14
)
# > $success
# > [1] TRUE

## End(Not run)
```

`form_assignment_list` *Summarize all Form Assignments of one Project.*

Description

[Experimental]

Usage

```
form_assignment_list(
  pid = get_default_pid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This returns a summary of all Form-specific Assignments within the Project in one transactional request. Each Assignment carries the `xmlFormId` of the Form it is attached to, as well as the `actor_id/role_id` pair.

Value

A tibble with one row per Form Assignment and the columns `actor_id`, `xml_form_id` and `role_id`.

See Also

<https://docs.getodk.org/central-api-project-management/#seeing-all-form-assignments-within-a-project>

Other user-management: [app_user_create\(\)](#), [app_user_delete\(\)](#), [app_user_list\(\)](#), [assignment_actors\(\)](#), [assignment_grant\(\)](#), [assignment_list\(\)](#), [assignment_revoke\(\)](#), [project_assignment_list\(\)](#), [role_detail\(\)](#), [role_list\(\)](#), [user_create\(\)](#), [user_delete\(\)](#), [user_detail\(\)](#), [user_list\(\)](#), [user_preference_project_delete\(\)](#), [user_preference_project_set\(\)](#), [user_preference_site_delete\(\)](#), [user_preference_site_set\(\)](#), [user_reset_password\(\)](#), [user_update\(\)](#), [user_update_password\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

al <- form_assignment_list()

al |> knitr::kable()

## End(Not run)
```

form_assignment_revoke

Revoke a Form Role Assignment from an Actor.

Description

[Experimental]

Usage

```
form_assignment_revoke(
  pid = get_default_pid(),
  fid = get_default_fid(),
  role_id,
  actor_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
role_id	(character or numeric) The Role ID, typically the integer ID of the Role. A Role system name can also be supplied if the Role has one.
actor_id	(numeric) The integer ID of the Actor.

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Given a Role and an Actor, this endpoint unassigns that Role from that Actor for this particular Form.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#revoking-a-form-role-assignment-from-an-actor>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

form_assignment_revoke(
  fid = fl$fid[[1]],
  role_id = "manager",
  actor_id = 14
```

```

)
# > $success
# > [1] TRUE

## End(Not run)

```

form_assignment_role_list

List Role-specific Form Assignments within a Project.

Description

[Experimental]

Usage

```

form_assignment_role_list(
  pid = get_default_pid(),
  role_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See vignette("Setup", package = "ruODK").
role_id	(numeric) The numeric ID of the Role.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.

`odkc_version` The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page `/version.txt`. Discard the "v". `ruODK` uses this parameter to adjust for breaking changes in ODK Central.

Default: `get_default_odkc_version` or "2023.5.1" if unset.

Set default `get_default_odkc_version` through `ru_setup(odkc_version="2023.5.1")`.

See `vignette("Setup", package = "ruODK")`.

Details

Like the Form Assignments summary, but filtered by some Role. Returned results include an `xmlFormId` field to specify which Form each Assignment is attached to. This endpoint requires ODK Central v0.7 or later.

Value

A tibble with one row for each Form Assignment of the Role as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-project-management/#seeing-role-specific-form-assignments-with>

Other project-management: `project_assignment_actors()`, `project_assignment_grant()`, `project_assignment_rev`, `project_create()`, `project_delete()`, `project_detail()`, `project_enable_encryption()`, `project_list()`, `project_replace()`, `project_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fa <- form_assignment_role_list(role_id = 2)

fa |> knitr::kable()

## End(Not run)
```

`form_attachment_download`

Download one Form attachment.

Description

[Experimental]

Usage

```

form_attachment_download(
  pid = get_default_pid(),
  fid = get_default_fid(),
  filename,
  dest = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
filename	(character) The name of the file to download, as given by <code>form_attachment_list()</code> .
dest	(character) The local file path to save the attachment to. Default: NULL, which saves to a temporary file with the same extension.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

To download a single file, use this endpoint. The Content-Type is either the type supplied at upload time, a type derived from the file extension, or application/octet-stream.

Value

The local file path the attachment was saved to.

See Also

<https://docs.getodk.org/central-api-form-management/#downloading-a-form-attachment>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

al <- form_attachment_list(fid = fid)

path <- form_attachment_download(fid = fid, filename = al$name[[1]])

file.info(path)$size

## End(Not run)
```

`form_attachment_list` *List all attachments of one Form.*

Description

[Experimental]

Usage

```
form_attachment_list(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This endpoint fetches the list of expected attachment files, and tells whether the server holds each file or not, and a hash of the file contents. To modify an attachment, create a Draft. The property `datasetExists` tells whether the attachment is a link to a Dataset instead of an actual file.

Value

A tibble with one row per expected Form attachment and the columns `name`, `type`, `exists`, `blob_exists`, `dataset_exists`, `size`, `hash` and `updated_at`, as far as Central returns them. Column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-form-management/#listing-form-attachments>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

al <- form_attachment_list(fid = fl$fid[[1]])

al |> knitr::kable()

## End(Not run)
```

form_create

Create a new Form.

Description

[Experimental]

Usage

```
form_create(
  pid = get_default_pid(),
  xml = NULL,
  file = NULL,
  publish = FALSE,
  ignore_warnings = FALSE,
  xls_form_id_fallback = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
xml	(character) The XForms XML definition as a single string. Exactly one of <code>xml</code> and <code>file</code> must be given. Default: <code>NULL</code> .
file	(character) Path to a local <code>.xml</code> , <code>.xls</code> or <code>.xlsx</code> file holding the Form definition. Exactly one of <code>xml</code> and <code>file</code> must be given. Default: <code>NULL</code> .
publish	(lgl) If <code>TRUE</code> , the Form skips the Draft state and is published immediately. Default: <code>FALSE</code> .

ignore_warnings	(lg1) If TRUE, the Form is created even if the XLSForm conversion results in warnings. Conversion errors always fail the request. Only used for .xls/.xlsx uploads. Default: FALSE.
xls_form_id_fallback	(character) The form ID to use if an uploaded spreadsheet does not specify one. Sent as the X-XlsForm-FormId-Fallback header. Only used for .xls/.xlsx uploads. Default: NULL (no header sent).
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

When creating a Form, the only required data is the actual XForms XML or XLSForm itself.

Forms will by default be created in Draft state, accessible under `/projects/.../forms/.../draft`. The Form itself will not have a public XML definition, and will not appear for download onto mobile devices. You will need to publish the Form to finalize it for data collection. To disable this behaviour, and force the new Form to be immediately ready, pass `publish = TRUE`.

The API will check the XML's structure in order to extract the information it needs about it, but ODK Central does not run comprehensive validation on the full contents of the XML. ODK Validate finds problems in the Form before upload. Publish the Draft after the Form passes validation.

Value

A list with the new Form's metadata as per the ODK Central API docs. Top level list elements are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-form-management/#creating-a-new-form>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`,

```

form_schema_ext(), form_update(), form_version_attachment_download(), form_version_attachment_list(),
form_version_detail(), form_version_list(), form_version_xlsx(), form_version_xml(),
form_xlsx(), form_xml(), public_link_create(), public_link_delete(), public_link_detail(),
public_link_list(), public_link_update()

```

Examples

```

## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

xml <- paste0(
  '<h:html xmlns="http://www.w3.org/2002/xforms" ',
  'xmlns:h="http://www.w3.org/1999/xhtml">',
  "<h:head><h:title>Simple</h:title><model><instance>",
  '<data id="simple"><meta><instanceID/></meta><name/></data>',
  "</instance>",
  '<bind nodeset="/data/meta/instanceID" type="string" readonly="true()" ',
  '<calculate="concat(\'uuid:\', uuid())"/>',
  '<bind nodeset="/data/name" type="string"/>',
  "</model></h:head>",
  '<h:body><input ref="/data/name"><label>Name</label></input></h:body>',
  "</h:html>"
)

f <- form_create(xml = xml, publish = TRUE)

f$xml_form_id
# > "simple"

## End(Not run)

```

form_dataset_diff

Show Datasets affected by one Form.

Description

[Experimental]

Usage

```

form_dataset_diff(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This endpoint lists the name and Properties of a Dataset that are affected by a Form. The list of Properties includes all published Properties of that Dataset, but each Property has the `inForm` flag to note whether or not that Form fills it in.

Value

A list with one entry per affected Dataset, each holding the Dataset name and its properties with `inForm` flags.

See Also

<https://docs.getodk.org/central-api-form-management/#published-form-related-datasets>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

dd <- form_dataset_diff(fid = fl$fid[[1]])

dd |> listviewer::jsonedit()

## End(Not run)
```

form_delete

*Delete a Form.***Description****[Experimental]****Usage**

```
form_delete(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

When a Form is deleted, it goes into the Trash section, but it can now be restored from the Trash with `form_restore()`. After 30 days in the Trash, the Form and all of its resources and Submissions will be automatically purged. If the goal is to prevent the Form from showing up on survey clients like ODK Collect, consider setting its state to closing or closed instead (see `form_update()`).

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#deleting-a-form>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

form_delete(fid = fl$fid[[1]])
# > $success
# > [1] TRUE

## End(Not run)
```

form_detail	<i>Show details for one form.</i>
-------------	-----------------------------------

Description**[Stable]****Usage**

```
form_detail(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.

Value

A tibble with one row and all form metadata as columns.

See Also

<https://docs.getodk.org/central-api-form-management/#getting-form-details>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# With explicit credentials, see tests
fl <- form_list()

# The first form in the test project
f <- form_detail(fid = fl$fid[[1]])

# form_detail returns exactly one row
nrow(f)
# > 1

# form_detail returns all form metadata as columns: name, xmlFormId, etc.
names(f)

# > "name" "fid" "version" "state" "submissions" "created_at"
# > "created_by_id" "created_by" "updated_at" "published_at"
# > "last_submission" "hash"

## End(Not run)
```

form_draft_attachment_delete

Clear a Draft Form attachment.

Description

[Experimental]

Usage

```
form_draft_attachment_delete(
  pid = get_default_pid(),
  fid = get_default_fid(),
  filename,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
filename	(character) The name of the expected file slot, as given by <code>form_draft_attachment_list()</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

A DELETE to a Draft attachment's endpoint clears the uploaded bytes while the expected file slot remains listed.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#clearing-a-draft-form-attachment>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

form_draft_attachment_delete(fid = fid, filename = "cities.csv")
# > $success
# > [1] TRUE

## End(Not run)
```

form_draft_attachment_download

Download one Draft Form attachment.

Description

[Experimental]

Usage

```
form_draft_attachment_download(
  pid = get_default_pid(),
  fid = get_default_fid(),
  filename,
  dest = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
filename	(character) The name of the file to download, as given by <code>form_draft_attachment_list()</code> .
dest	(character) The local file path to save the attachment to. Default: NULL, which saves to a temporary file with the same extension.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

To download a single file from a Draft Form, use this endpoint.

Value

The local file path the attachment was saved to.

See Also

<https://docs.getodk.org/central-api-form-management/#downloading-a-draft-form-attachment>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

al <- form_draft_attachment_list(fid = fid)

path <- form_draft_attachment_download(fid = fid, filename = al$name[[1]])

file.info(path)$size

## End(Not run)
```

form_draft_attachment_link

Link a Dataset to a Draft Form attachment.

Description

[Experimental]

Usage

```
form_draft_attachment_link(
  pid = get_default_pid(),
  fid = get_default_fid(),
  filename,
  dataset = TRUE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

filename	(character) The name of the attachment, e.g. "people.csv" for a Dataset named "people".
dataset	(lgl) Link (TRUE) or unlink (FALSE) the Dataset.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This endpoint updates a Form Attachment's link to a Dataset. Linking only happens if the Attachment type is file and a Dataset with the exact name of the Attachment (excluding the .csv extension) exists in the Project. Letter case and spaces must match exactly. When a Dataset is linked, any attached file is removed. This endpoint requires ODK Central v2022.3 or later.

Value

A list with the updated attachment's metadata as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#linking-a-dataset-to-a-draft-form-attachment>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")
```

```

fl <- form_list()
fid <- fl$fid[[1]]

form_draft_attachment_link(fid = fid, filename = "people.csv", dataset = TRUE)

## End(Not run)

```

form_draft_attachment_list

List all expected attachments of a Draft Form.

Description

[Experimental]

Usage

```

form_draft_attachment_list(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See vignette("Setup", package = "ruODK").

retries The number of attempts to retrieve a web resource.
 This parameter is given to `RETRY(times = retries)`.
 Default: 3.

Details

When a Draft is created, the expected Form Attachments are computed and slots are created, as with a new Form. Attachments that match existing ones on the published Form are copied over to the new Draft.

Value

A tibble with one row per expected Draft attachment and the columns name, type, exists, blob_exists, dataset_exists and updated_at, as far as Central returns them. Column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-form-management/#listing-expected-draft-form-attachments>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

al <- form_draft_attachment_list(fid = fl$fid[[1]])

al |> knitr::kable()

## End(Not run)
```

form_draft_attachment_upload

Upload a Draft Form attachment.

Description

[Experimental]

Usage

```
form_draft_attachment_upload(
  pid = get_default_pid(),
  fid = get_default_fid(),
  filename,
  file,
  content_type = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
filename	(character) The name of the expected file slot, as given by <code>form_draft_attachment_list()</code> .
file	(character) Path to a local file whose bytes are uploaded to the expected file slot.
content_type	(character) The MIME type sent as Content-Type header. Default: NULL, which sends "application/octet-stream".
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").

pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

To upload a binary to an expected file slot, POST the binary to its endpoint. Supply a `content_type` if you have one. If a Dataset is linked to this attachment, the upload unlinks and replaces it with the uploaded file.

Value

A list with the updated attachment's metadata as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#uploading-a-draft-form-attachment>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

al <- form_draft_attachment_list(fid = fid)

form_draft_attachment_upload(
  fid = fid,
  filename = al$name[[1]],
  file = "/path/to/file.csv"
)

## End(Not run)
```

form_draft_create	Create a Draft Form.
-------------------	----------------------

Description

[Experimental]

Usage

```
form_draft_create(
  pid = get_default_pid(),
  fid = get_default_fid(),
  xml = NULL,
  file = NULL,
  ignore_warnings = FALSE,
  xls_form_id_fallback = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
xml	(character) The XForms XML definition as a single string. If neither xml nor file is given, the Draft copies the published definition. Default: NULL.
file	(character) Path to a local .xml, .xls or .xlsx file holding the Form definition. Default: NULL.
ignore_warnings	(lgl) If TRUE, the Draft is created even if the XLSForm conversion results in warnings. Only used for .xls/.xlsx uploads. Default: FALSE.
xls_form_id_fallback	(character) The form ID to use if an uploaded spreadsheet does not specify one. Only used for .xls/.xlsx uploads. Default: NULL (no header sent).
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

POSTing here creates a new Draft Form on the given Form. For the most part, it takes the same parameters as the Create Form request: you can submit XML or Excel files, and you can provide `ignore_warnings`. Additionally, a call with neither `xml` nor `file` creates a Draft Form with a copy of the published definition, if there is one. This is useful to change Form Attachments without updating the Form definition itself. Even if a Draft exists, you can always replace it by calling this function again. The `xmlFormId` must exactly match that of the Form overall, or the request is rejected.

Value

A list with a single element `success` (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#creating-a-draft-form>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

form_draft_create(fid = fl$fid[[1]])
# > $success
# > [1] TRUE

## End(Not run)
```

form_draft_dataset_diff

Show Dataset changes of a Draft Form.

Description

[Experimental]

Usage

```
form_draft_dataset_diff(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.

Details

This endpoint reflects the change to a Dataset that takes effect once the Form is published. Like the published endpoint, it lists the Dataset name and Properties, but it also includes the `isNew` flag on both the Dataset and each individual Property. This flag is true only if the Dataset or Property is new and publishing the Draft Form creates it.

Value

A list with one entry per affected Dataset, each holding the Dataset name, its `isNew` flag, and its properties with `inForm` and `isNew` flags.

See Also

<https://docs.getodk.org/central-api-form-management/#draft-form-dataset-diff>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

dd <- form_draft_dataset_diff(fid = fl$fid[[1]])

dd |> listviewer::jsonedit()

## End(Not run)
```

form_draft_delete

Delete a Draft Form.

Description

[Experimental]

Usage

```
form_draft_delete(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.

Details

Once a Draft Form is deleted, its definition and any Form Attachments associated with it are removed. A Draft cannot be deleted if the Form has no published version.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#deleting-a-draft-form>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

form_draft_delete(fid = fl$fid[[1]])
# > $success
# > [1] TRUE

## End(Not run)
```

form_draft_detail	<i>Show details of a Draft Form.</i>
-------------------	--------------------------------------

Description

[Experimental]

Usage

```
form_draft_detail(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

The response includes standard overall Form metadata, like `xmlFormId`, in addition to the Draft-specific information.

Value

A list with the Draft Form's metadata as per the ODK Central API docs, including the `draft_token`. Top level list elements are renamed from ODK's `camelCase` to `snake_case`.

See Also

<https://docs.getodk.org/central-api-form-management/#getting-draft-form-details>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

d <- form_draft_detail(fid = fl$fid[[1]])

d$xml_form_id

## End(Not run)
```

form_draft_publish	<i>Publish a Draft Form.</i>
--------------------	------------------------------

Description**[Experimental]****Usage**

```
form_draft_publish(
  pid = get_default_pid(),
  fid = get_default_fid(),
  version = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
version	(character) The version to associate with the Draft once it is published. Default: NULL (the Draft keeps its version).

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This publishes the current Draft Form and makes it the active Form definition (and attachments). If the Draft version conflicts with an older version of the Form, the publish operation fails, unless the new version string is given with version and Central sets it on publish. Once the Draft is published, there is no longer a Draft version of the Form. Publishing a Draft Form that defines a Dataset schema also publishes the Dataset.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#publishing-a-draft-form>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "....svc", un = "me@email.com", pw = "...")

fl <- form_list()

form_draft_publish(fid = fl$fid[[1]], version = "2")
```

```
# > $success
# > [1] TRUE

## End(Not run)
```

form_draft_xlsx

Download the XLSForm of a Draft Form.

Description

[Experimental]

Usage

```
form_draft_xlsx(
  pid = get_default_pid(),
  fid = get_default_fid(),
  dest = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
dest	(character) The local file path to save the spreadsheet to. Default: NULL, which saves to a temporary .xlsx file.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

retries The number of attempts to retrieve a web resource.
 This parameter is given to `RETRY`(times = retries).
 Default: 3.

Details

If a Draft Form was created with an Excel file (.xls or .xlsx), this function gets that file back. Draft Forms created from XForms XML have no spreadsheet source, and Central answers those requests with 404.

Value

The local file path the spreadsheet was saved to.

See Also

<https://docs.getodk.org/central-api-form-management/#retrieving-draft-form-xls-x>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

xlsx <- form_draft_xlsx(fid = fl$fid[[1]])

file.info(xlsx)$size

## End(Not run)
```

form_draft_xml

Show the XML of a Draft Form.

Description

[Experimental]

Usage

```
form_draft_xml(
  parse = TRUE,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

parse	Whether to parse the XML into a nested list, default: TRUE
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

To get the XML of the Draft Form, this function requests the Draft resource with `.xml` appended.

Value

The Draft Form XML as a nested list, or the raw response if `parse` is FALSE.

See Also

<https://docs.getodk.org/central-api-form-management/#retrieving-draft-form-xml>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

dx <- form_draft_xml(fid = fl$fid[[1]])

names(dx)

## End(Not run)
```

form_link

Show Form details by Form Link ID.

Description

[Experimental]

Usage

```
form_link(
  form_link_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```


Arguments

form_link_id	(character) The Form Link ID, e.g. the Enketo ID of a published Form.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Returns Form metadata for the provided Form Link ID without needing the Project ID or Form ID.

Value

A tibble with one row holding the Form's metadata as columns as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#getting-form-details-by-formlinkid>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

f <- form_link(form_link_id = fl$enketo_id[[1]])
```

```
f$fid

## End(Not run)
```

form_list	<i>List all forms.</i>
-----------	------------------------

Description

[Stable]

Usage

```
form_list(
  pid = get_default_pid(),
  deleted = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
deleted	(lgl) If TRUE, list only deleted Forms with their numeric form IDs, which can be used to restore a deleted Form with <code>form_restore()</code>. Default: FALSE.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.

orders (vector of character) Orders of datetime elements for lubridate.
Default: c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd").

tz A timezone to convert dates and times to.
Read vignette("setup", package = "ruODK") to learn how ruODK's timezone can be set globally or per function.

Value

A tibble with one row per form and all given form metadata as cols. Column names are sanitized into snake_case. Nested columns (review start and created by) are flattened and prefixed. The column xml_form_id is replicated as fid according to ruODK naming standards. With deleted = TRUE, the tibble holds one row per deleted Form with its numeric form ID.

See Also

<https://docs.getodk.org/central-api-form-management/#list-all-forms>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# With default pid
fl <- form_list()

# With explicit pid
fl <- form_list(pid = 1)

class(fl)
# > c("tbl_df", "tbl", "data.frame")

# Filter out draft forms (published_at=NA)
only_published_forms <- fl %>% dplyr::filter(is.na(published_at))

# Note: older ODK Central versions < 1.1 have published_at = NA for both
# published and draft forms. Drafts have NA for version and hash.
only_published_forms <- fl %>% dplyr::filter(is.na(version) & is.na(hash))

## End(Not run)
```

form_restore	<i>Restore a deleted Form.</i>
--------------	--------------------------------

Description

[Experimental]

Usage

```
form_restore(
  pid = get_default_pid(),
  id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
id	The numeric ID of the deleted Form, e.g. from <code>form_list(deleted = TRUE)</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

Deleted Forms can be restored as long as they have been in the Trash less than 30 days and have not been purged. A deleted Form with the same `xmlFormId` as an active Form cannot be restored while that other Form is active. This function uses the numeric ID of the Form (returned by `form_list(deleted = TRUE)`) rather than the `xmlFormId` to restore unambiguously. This endpoint requires ODK Central v1.4 or later.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#restoring-a-form>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

trash <- form_list(deleted = TRUE)

form_restore(id = trash$id[[1]])
# > $success
# > [1] TRUE

## End(Not run)
```

form_schema

Show the schema of one form.

Description

[Stable]

Usage

```
form_schema(
  flatten = FALSE,
  odata = FALSE,
  parse = TRUE,
  draft = FALSE,
  version = NULL,
  pid = get_default_pid(),
```

```

    fid = get_default_fid(),
    url = get_default_url(),
    un = get_default_un(),
    pw = get_default_pw(),
    odkc_version = get_default_odkc_version(),
    retries = get_retries(),
    verbose = get_ru_verbose()
)

```

Arguments

flatten	Whether to flatten the resulting list of lists (TRUE) or not (FALSE, default). Only applies to ODK Central version < 0.8.
odata	Whether to sanitise the field names to match the way they will be outputted for OData. While the original field names as given in the XForms definition may be used as-is for CSV output, OData has some restrictions related to the domain-qualified identifier syntax it uses. Only applies to ODK Central version < 0.8. Default: FALSE.
parse	Whether to parse the form schema into a tibble of form field type and name. This uses form_schema_parse internally. If used together with <code>flatten=TRUE</code> , form_schema will raise a warning and return the unparsed, flattened form schema. Only applies to ODK Central version < 0.8. Default: TRUE.
draft	Whether the form is published (FALSE) or a draft (TRUE). Default: TRUE.
version	(character) The published Form version whose schema fields to return, e.g. from <code>form_version_list()</code> . Pass <code>___</code> for a blank version. If given, the version path is used instead of the published or draft path. Default: NULL.
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").

odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

ODK Central has introduced a new API endpoint in version 0.8 which returns a parsed and flattened list of fields. This replaces the nested form schema which is challenging to parse.

While users of newer ODK Central versions (> 0.8) can ignore the legacy support for ODK Central's earlier form schema API, users of ODK Central version < 0.8 can set an environment variable `ODKC_VERSION` to their ODKC's version in format <major>.<minor> e.g. 0.7. This variable caters for future breaking changes.

Either way, `form_schema` will always return a tibble with columns `name`, `type`, `path` and `ruodk_name`.

Value

A tibble or nested list (v0.7) containing the form definition. At the lowest nesting level, each form field consists of a list of two nodes, `name` (the underlying field name) and `type` (the XForms field type, as in "string", "select1", "geopoint", "binary" and so on). These fields are nested in lists of tuples `name` (the XForms screen name), `children` (the fields as described above), `type` ("structure" for non- repeating screens, "repeat" for repeating screens). A list with name "meta" may precede the structure, if several metadata fields are captured (e.g. "instanceId", form start datetimes etc.). In all cases for ODK Central 0.8, and with default parameters (`parse=TRUE`) for ODK Central 0.7, `form_schema` returns a tibble with the columns:

- `name` The field name as given in the form schema.
- `type` The field type, e.g. "string", "select1", etc.
- `path` The XForms path of the field,
- `selectMultiple` Whether a field of type "select" is a select multiple (TRUE). Any other types are NA.
- `ruodk_name` The predicted field name as generated by `odata_submission_get`, prefixed by the path, additionally cleaned with `make_clean_names` to match the cleaned column names from `odata_submission_rectangle`.

See Also

<https://docs.getodk.org/central-api-form-management/#getting-form-schema-fields>

<https://docs.getodk.org/central-api-form-management/#getting-form-version-schema-fields>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# With explicit pid and fid
fs_defaults <- form_schema(pid = 1, fid = "build_xformsId")

# With current ODK Central (v0.8)
fs <- form_schema()

# With defaults, ODK Central v0.7
fs_nested <- form_schema(
  flatten = FALSE,
  odata = FALSE,
  parse = FALSE,
  odkc_version = 0.7
)
listviewer::jsonedit(fs_nested)

fs_flattened <- form_schema(
  flatten = TRUE,
  odata = FALSE,
  parse = FALSE,
  odkc_version = 0.7
)
listviewer::jsonedit(fs_flattened)

# form_schema returns a nested list. There's nothing to change about that.
class(fs_nested)
# > "list"

class(fs_flattened)
# > "list"
```



```

# This assumes knowledge of that exact form being tested.
# First node: type "structure" (a field group) named "meta".
fs_nested[[1]]$type
# > "structure"

fs_nested[[1]]$name
# > "meta"

# The first node contains children, which means it's an XForms field group.
names(fs_nested[[1]])
# > "name" "children" "type"

# Next node: a "meta" field of type "string" capturing the "instanceId".
# First child node of "meta": type "string", name "instanceId".
fs_nested[[1]]$children[[1]]$type
# > "string"
fs_nested[[1]]$children[[1]]$name
# > "instanceId"

# In the flattened version, the field's and it's ancestors' names are the
# components of "path".
fs_flattened[[1]]$path
# > "meta". "instanceId"

fs_flattened[[1]]$type
# > "string"

# Last node: a "meta" field capturing the datetime of form completion
fs_flattened[[length(fs_flattened)]]$type
# > "dateTime"
fs_nested[[length(fs_nested)]]$type
# > "dateTime"

# Parsed into a tibble of form field type/name:
# Useful to inform further parsing of submission data (attachments, dates)
fs <- form_schema(parse = TRUE, odkc_version = 0.7)
fs <- form_schema(odkc_version = 0.8)

# Attachments: used by handle_ru_attachments
fs %>% dplyr::filter(type == "binary")

# dateTime: used by handle_ru_datetimes
fs %>% dplyr::filter(type == "dateTime")

# Point location: used by handle_ru_geopoints
fs %>% dplyr::filter(type == "geopoint")

## End(Not run)

```

Description**[Experimental]****Usage**

```
form_schema_ext(
  flatten = FALSE,
  odata = FALSE,
  parse = TRUE,
  version = NULL,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  odkc_version = get_default_odkc_version(),
  retries = get_retries(),
  verbose = get_ru_verbose()
)
```

Arguments

flatten	Whether to flatten the resulting list of lists (TRUE) or not (FALSE, default). Only applies to ODK Central version < 0.8.
odata	Whether to sanitise the field names to match the way they will be outputted for OData. While the original field names as given in the XForms definition may be used as-is for CSV output, OData has some restrictions related to the domain-qualified identifier syntax it uses. Only applies to ODK Central version < 0.8. Default: FALSE.
parse	Whether to parse the form schema into a tibble of form field type and name. This uses form_schema_parse internally. If used together with <code>flatten=TRUE</code> , form_schema will raise a warning and return the unparsed, flattened form schema. Only applies to ODK Central version < 0.8. Default: TRUE.
version	(character) The published Form version whose schema fields to return, e.g. from <code>form_version_list()</code> . Pass <code>___</code> for a blank version. If given, the version path is used instead of the published or draft path. Default: NULL.
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").

url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code> 's verbosity can be set globally or per function.

Details

ODK Central has introduced a new API endpoint in version 0.8 which returns a parsed and flattened list of fields. This replaces the nested form schema which is challenging to parse. This list is returned by [form_schema](#).

However this still misses important elements, in particular labels, hints and choice_lists.

[form_schema_ext](#) returns the same object as [form_schema](#) adding labels, hints and choice lists in all languages available. This is done by using the return object from [form_xml](#).

It has the exact function signature as [form_schema](#). In that sense, any call to [form_schema](#) can be replaced by [form_schema_ext](#)

This function, however, has been prepared with ODK Central version 0.8 or higher. If you use it with an earlier version, a warning will be given.

Value

A tibble containing the form definition. For ODK Central 0.8, and with default parameters (`parse=TRUE`) for ODK Central 0.7, [form_schema](#) returns a tibble with the columns:

- name The field name as given in the form schema.
- type The field type, e.g. "string", "select1", etc.
- path The XForms path of the field,

- `ruodk_name` The predicted field name as generated by `odata_submission_get`, prefixed by the path, additionally cleaned with `make_clean_names` to match the cleaned column names from `odata_submission_rectangle`.
- `label` The field label as given in the form schema. If specific languages are available, this column will return the default language or it will be empty if this is not specified.
- `label_lang` The field label in language `_lang` as given in the form schema.
- `hint` The field hint as given in the form schema. If specific languages are available, this column will return the default language or it will be empty if this is not specified.
- `hint_lang` The field hint in language `_lang` as given in the form schema.
- `choices` A list of lists containing at least values and, if available, labels of the choices as given in the form schema. If specific languages are available, this column will return the default language or it will be empty if this is not specified. Please notice that whenever choice filters are applied, this will return the unfiltered choice list.
- `choices_lang` A list of lists containing at least values and, if available, labels of the choices in language `_lang` as given in the form schema. Please notice that whenever choice filters are applied, this will return the unfiltered choice list.

See Also

<https://docs.getodk.org/central-api-form-management/#getting-form-schema-fields>

<https://docs.getodk.org/central-api-form-management/#getting-form-version-schema-fields>

<https://docs.getodk.org/central-api-form-management/#retrieving-form-xml>

<https://docs.getodk.org/central-api-form-management/#retrieving-form-version-xml>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# With current ODK Central (>0.7)
# get extended schema:
fsx <- form_schema_ext()

# print choice list in english:
fsx[fsx$name == "test_yn", "choices_english_(en)"][[1]]
```

```
# view the extended schema:
fsx

# The extended schema of one published Form version:
v1 <- form_version_list()
fsx_v1 <- form_schema_ext(version = v1$version[[1]])

## End(Not run)
```

form_schema_parse	<i>Parse a form_schema into a tibble of fields with name, type, and path.</i>
-------------------	---

Description

[Stable]

Usage

```
form_schema_parse(fs, path = "Submissions", verbose = get_ru_verbose())
```

Arguments

fs	The output of form_schema as nested list
path	The base path for form fields. Default: "Submissions". form_schema_parse recursively steps into deeper nesting levels, which are reflected as separate OData tables. The returned value in path reflects the XForms group name, which translates to separate screens in ODK Collect. Non-repeating form groups will be flattened out into the main Submissions table. Repeating groups are available as separate OData tables.
verbose	Whether to display debug messages or not. Read vignette("setup", package = "ruODK") to learn how ruODK's verbosity can be set globally or per function.

Details

This function is used by [form_schema](#) for older versions of ODK Central (pre 0.8). These return the form schema as XML, requiring the quite involved code of [form_schema_parse](#), while newer ODK Central versions return JSON, which is parsed directly in [form_schema](#).

The form_schema returned from ODK Central versions < 0.8 is a nested list of lists containing the form definition. The form definition consists of fields (with a type and name), and form groups, which are rendered as separate ODK Collect screens. Form groups in turn can also contain form fields.

[form_schema_parse](#) recursively unpacks the form and extracts the name and type of each field. This information then informs [handle_ru_attachments](#), [handle_ru_datetimes](#), [handle_ru_geopoints](#), [handle_ru_geotraces](#), and [handle_ru_geoshapes](#).

See Also

Other utilities: [attachment_get\(\)](#), [attachment_link\(\)](#), [attachment_url\(\)](#), [drop_null_coords\(\)](#), [get_one_attachment\(\)](#), [get_one_submission\(\)](#), [get_one_submission_att_list\(\)](#), [get_one_submission_audit\(\)](#), [handle_ru_attachments\(\)](#), [handle_ru_datetimes\(\)](#), [handle_ru_geopoints\(\)](#), [handle_ru_geoshapes\(\)](#), [handle_ru_geotraces\(\)](#), [isodt_to_local\(\)](#), [odata_submission_rectangle\(\)](#), [predict_ruodk_name\(\)](#), [prepend_uuid\(\)](#), [ru_http_request\(\)](#), [split_geopoint\(\)](#), [split_geoshape\(\)](#), [split_geotrace\(\)](#), [strip_uuid\(\)](#), [tidyeval](#), [unnest_all\(\)](#)

Examples

```
## Not run:
# Option 1: in two steps, ODKC Version 0.7
fs <- form_schema(flatten = FALSE, parse = FALSE, odkc_version = 0.7)
fsp <- form_schema_parse(fs)

# Option 2: in one go
fsp <- form_schema(parse = TRUE)

fsp

## End(Not run)
```

form_update

Modify a Form's state.

Description

[Experimental]

Usage

```
form_update(
  pid = get_default_pid(),
  fid = get_default_fid(),
  state = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid The numeric ID of the project, e.g.: 2.
 Default: [get_default_pid](#).
 Set default pid through `ru_setup(pid="...")`.
 See `vignette("Setup", package = "ruODK")`.

fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
state	(character) The new lifecycle state of the Form. One of "open", "closing" or "closed".
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

It is currently possible to modify only one thing about a Form: its state, which governs whether it is available for download onto survey clients and whether it accepts new Submissions.

Only the properties you supply are changed. Anything you do not supply remains untouched. This changes only the state. It does not change other properties.

Value

A list with the updated Form's metadata as per the ODK Central API docs. Top level list elements are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-form-management/#modifying-a-form>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

f <- form_update(fid = fl$fid[[1]], state = "closing")

f$state
# > "closing"

## End(Not run)
```

form_version_attachment_download

Download one attachment of a published Form version.

Description

[Experimental]

Usage

```
form_version_attachment_download(
  pid = get_default_pid(),
  fid = get_default_fid(),
  version,
  filename,
  dest = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

version	(character) The version of the Form version being referenced. Pass <code>___</code> for a blank version.
filename	(character) The name of the file to download, as given by <code>form_version_attachment_list()</code> .
dest	(character) The local file path to save the attachment to. Default: <code>NULL</code> , which saves to a temporary file with the same extension.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Attachments are specific to each version of a Form. If the server holds a copy of the expected file, append its filename to the version attachments URL to download only that file.

Value

The local file path the attachment was saved to.

See Also

<https://docs.getodk.org/central-api-form-management/#downloading-a-form-version-attachment>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")
```

```

fl <- form_list()
fid <- fl$fid[[1]]

vl <- form_version_list(fid = fid)
version <- vl$version[[1]]

al <- form_version_attachment_list(fid = fid, version = version)

path <- form_version_attachment_download(
  fid = fid,
  version = version,
  filename = al$name[[1]]
)

file.info(path)$size

## End(Not run)

```

form_version_attachment_list

List expected attachments of one published Form version.

Description

[Experimental]

Usage

```

form_version_attachment_list(
  pid = get_default_pid(),
  fid = get_default_fid(),
  version,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

version	(character) The version of the Form version being referenced. Pass <code>___</code> for a blank version.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This lists the expected attachments of a published Form version with a boolean flag telling whether the server holds a copy of each file or not.

Value

A tibble with one row per expected attachment and the columns name and exists.

See Also

<https://docs.getodk.org/central-api-form-management/#listing-form-version-attachments>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

vl <- form_version_list(fid = fid)
```

```

al <- form_version_attachment_list(fid = fid, version = vl$version[[1]])

al |> knitr::kable()

## End(Not run)

```

form_version_detail *Show details of one published Form version.*

Description

[Experimental]

Usage

```

form_version_detail(
  pid = get_default_pid(),
  fid = get_default_fid(),
  version,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
version	(character) The version of the Form version being referenced. Pass <code>___</code> for a blank version.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Since the XForms specification allows blank strings as versions (and Central treats the lack of a version as a blank string), pass the special value `___` (three underscores) as the version to retrieve the blank version.

Value

A tibble with one row holding the Form version's metadata as columns as per the ODK Central API docs. Column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-form-management/#getting-form-version-details>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

v1 <- form_version_list(fid = fid)

vd <- form_version_detail(fid = fid, version = v1$version[[1]])

vd |> knitr::kable()

## End(Not run)
```

form_version_list	List all published versions of one Form.
-------------------	--

Description

[Experimental]

Usage

```
form_version_list(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.

tz A timezone to convert dates and times to.
 Read vignette("setup", package = "ruODK") to learn how ruODK's timezone can be set globally or per function.

Details

Each entry of the version listing contains some of the same duplicate keys with basic information about the Form: xmlFormId and createdAt, for example. This matches the data received when each version is requested separately.

Value

A tibble with one row per published Form version and all version metadata as columns as per the ODK Central API docs. Column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-form-management/#listing-published-form-versions>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

vl <- form_version_list(fid = fl$fid[[1]])

vl |> knitr::kable()

## End(Not run)
```

form_version_xlsx

Download the XLSForm of one published Form version.

Description

[Experimental]

Usage

```
form_version_xlsx(
  pid = get_default_pid(),
  fid = get_default_fid(),
  version,
  dest = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
version	(character) The version of the Form version being referenced. Pass <code>___</code> for a blank version.
dest	(character) The local file path to save the spreadsheet to. Default: <code>NULL</code> , which saves to a temporary <code>.xlsx</code> file.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").

retries The number of attempts to retrieve a web resource.
 This parameter is given to `RETRY`(times = retries).
 Default: 3.

Details

If a Form Version was created with an Excel file (.xls or .xlsx), this function gets that file back. Versions created from XForms XML have no spreadsheet source, and Central answers those requests with 404.

Value

The local file path the spreadsheet was saved to.

See Also

<https://docs.getodk.org/central-api-form-management/#retrieving-form-version-xls-x>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "....svc", un = "me@email.com", pw = "...")

f1 <- form_list()
fid <- f1$fid[[1]]

v1 <- form_version_list(fid = fid)

xlsx <- form_version_xlsx(fid = fid, version = v1$version[[1]])

file.info(xlsx)$size

## End(Not run)
```

form_version_xml

*Show the XML of one published Form version.***Description****[Experimental]****Usage**

```
form_version_xml(
  parse = TRUE,
  pid = get_default_pid(),
  fid = get_default_fid(),
  version,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

parse	Whether to parse the XML into a nested list, default: TRUE
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
version	(character) The version of the Form version being referenced. Pass <code>___</code> for a blank version.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

To get the XML of a published Form version, this function requests the version resource with `.xml` appended. Since the XForms specification allows blank strings as versions, pass the special value `___` (three underscores) as the version to retrieve the blank version.

Value

The Form version XML as a nested list, or the raw response if `parse` is `FALSE`.

See Also

<https://docs.getodk.org/central-api-form-management/#retrieving-form-version-xml>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

vl <- form_version_list(fid = fid)

vx <- form_version_xml(fid = fid, version = vl$version[[1]])

names(vx)

## End(Not run)
```

form_xlsx

Download the XLSForm of one Form.

Description

[Experimental]

Usage

```
form_xlsx(
  pid = get_default_pid(),
  fid = get_default_fid(),
  dest = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
dest	(character) The local file path to save the spreadsheet to. Default: NULL, which saves to a temporary .xlsx file.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

If a Form was created with an Excel file (.xls or .xlsx), this function gets that file back. Forms created from XForms XML have no spreadsheet source, and Central answers those requests with 404.

Value

The local file path the spreadsheet was saved to.

See Also

<https://docs.getodk.org/central-api-form-management/#retrieving-form-xls-x>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

xlsx <- form_xlsx(fid = fl$fid[[1]])

file.info(xlsx)$size

## End(Not run)
```

form_xml

Show the XML representation of one form as list.

Description

[Stable]

Usage

```
form_xml(
  parse = TRUE,
  pid = get_default_pid(),
  fid = get_default_fid(),
  version = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

parse	Whether to parse the XML into a nested list, default: TRUE
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
version	(character) The version of the Form version being referenced. Pass <code>___</code> for a blank version. If NULL (default), the published Form XML is returned.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

To get the XML of the Form, add `.xml` to the end of the request URL. To get the XML of one published Form version, request the version resource with `.xml` appended. Since the XForms specification allows blank strings as versions, pass the special value `___` (three underscores) as the version to retrieve the blank version.

Value

The form XML as a nested list.

See Also

<https://docs.getodk.org/central-api-form-management/#retrieving-form-xml>

<https://docs.getodk.org/central-api-form-management/#retrieving-form-version-xml>

Other form-management: [form_assignment_actors\(\)](#), [form_assignment_grant\(\)](#), [form_assignment_revoke\(\)](#), [form_attachment_download\(\)](#), [form_attachment_list\(\)](#), [form_create\(\)](#), [form_dataset_diff\(\)](#), [form_delete\(\)](#), [form_detail\(\)](#), [form_draft_attachment_delete\(\)](#), [form_draft_attachment_download\(\)](#), [form_draft_attachment_link\(\)](#), [form_draft_attachment_list\(\)](#), [form_draft_attachment_upload\(\)](#),

```

form_draft_create(), form_draft_dataset_diff(), form_draft_delete(), form_draft_detail(),
form_draft_publish(), form_draft_xlsx(), form_draft_xml(), form_link(), form_list(),
form_restore(), form_schema(), form_schema_ext(), form_update(), form_version_attachment_download(),
form_version_attachment_list(), form_version_detail(), form_version_list(), form_version_xlsx(),
form_version_xml(), form_xlsx(), public_link_create(), public_link_delete(), public_link_detail(),
public_link_list(), public_link_update()

```

Examples

```

## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# With explicit pid and fid
fxml_defaults <- form_xml(1, "build_xformsId")

# With defaults
fxml <- form_xml()
listviewer::jsonedit(fxml)

# form_xml returns a nested list
class(fxml)
# > "list"

# The XML of one published Form version
v1 <- form_version_list()
fxml_v1 <- form_xml(version = v1$version[[1]], parse = FALSE)

## End(Not run)

```

fq_attachments

A tibble of submission attachments.

Description

[Stable]

Usage

```
fq_attachments
```

Format

A tibble of submission attachments.

Source

The output of `attachment_list` run on submissions of the test form system.file("extdata", "FloraQuadrat04.xml", package = "ruODK").

See Also

Other included: [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

<code>fq_data</code>	<i>Parsed submission data for an ODK Central form.</i>
----------------------	--

Description

[Stable]

Usage

`fq_data`

Format

The output of [odata_submission_get](#) for a set of example data. A tidy tibble referencing the attachments included in the vignettes and documentation at a relative path at `attachments/media/<filename>.<ext>`.

Details

The parsed OData response for the submissions of an ODK Central form. This form represents a Flora Quadrat, which is a ca 50 by 50 m quadrat of a uniform plant community.

The XML and .odkbuild versions for this form are available as `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")` and `system.file("extdata", "FloraQuadrat04.odkbuild", package = "ruODK")`, respectively.

This data is kept up to date with the data used in vignettes and package tests. The data is comprised of test records with nonsensical data. The forms used to capture this data are development versions of real-world forms.

Source

See `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")` and [odata_submission_get](#).

See Also

Other included: [fq_attachments](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

fq_data_strata

*Parsed submission data for a subgroup of an ODK Central form.***Description****[Stable]****Usage**

fq_data_strata

Format

The output of [odata_submission_get](#) for a set of example data. A tidy tibble referencing the attachments included in the vignettes and documentation at a relative path `attachments/media/<filename>.<ext>`.

Details

The parsed OData response for the subgroup of an ODK Central form.

This subgroup represents vegetation strata as per the NVIS classification. A vegetation stratum is a layer of plants with the same height, and dominated by one or few plant taxa. Plant communities can be made of up to five strata, with two to three being most common.

This data is kept up to date with the data used in vignettes and package tests. The data is comprised of test records with nonsensical data. The forms used to capture this data are development versions of real-world forms.

Source

See `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")` and [odata_submission_get](#).

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

fq_data_taxa

*Parsed submission data for a subgroup of an ODK Central form.***Description****[Stable]****Usage**

fq_data_taxa

Format

The output of [odata_submission_get](#) for a set of example data. A tidy tibble referencing the attachments included in the vignettes and documentation at a relative path attachments/media/<filename>.<ext>.

Details

The parsed OData response for a subgroup of an ODK Central form.

This subgroup represents an individual plant taxon which is encountered by the enumerators. Typically, one voucher specimen is taken for each distinct encountered plant taxon. A field name is allocated by the enumerators, which can be the proper canonical name (if known) or any other moniker. The voucher specimens are later determined by taxonomic experts, who then provide the real, terminal taxonomic name for a given voucher specimen.

This data is kept up to date with the data used in vignettes and package tests. The data is comprised of test records with nonsensical data. The forms used to capture this data are development versions of real-world forms.

Source

See `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")` and [odata_submission_get](#).

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

fq_form_detail	<i>A tibble of form metadata.</i>
----------------	-----------------------------------

Description**[Stable]****Usage**

```
fq_form_detail
```

Format

A tibble of form metadata.

Source

The output of `form_detail` run on submissions of the test form system. `file("extdata", "FloraQuadrat04.xml", package = "ruODK")`.

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_list`, `fq_form_schema`, `fq_form_xml`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_submissions`, `fq_svc`, `fq_zip_data`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7`, `fs_v7_raw`, `geo_fs`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt88`, `geo_wkt_raw`

fq_form_list	<i>A tibble of forms.</i>
--------------	---------------------------

Description**[Stable]****Usage**

```
fq_form_list
```

Format

A tibble of forms

Source

The output of `form_list`. run on the project.

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

fq_form_schema

JSON form schema for an ODK Central form.

Description

[Stable]

Usage

```
fq_form_schema
```

Format

The output of `ruODK::form_schema()`, a tibble with columns "type", "name" and "path" and one row per form field.

Details

The parsed form schema of an ODK Central form.

This data is kept up to date with the data used in vignettes and package tests. The data is comprised of test records with nonsensical data. The forms used to capture this data are development versions of real-world forms.

This data is used to build vignettes offline and without the need for credentials to an ODK Central server. The test suite ensures that the "canned" data is identical to the "live" data.

Source

See `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")` and `ruODK::form_schema()`.

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

fq_form_xml*A nested list of a form definition.*

Description**[Stable]****Usage**

fq_form_xml

Format

A nested list of a form definition.

Source

The output of `form_xml` run on the test form system. `file("extdata", "FloraQuadrat04.xml", package = "ruODK")`.

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_detail`, `fq_form_list`, `fq_form_schema`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_submissions`, `fq_svc`, `fq_zip_data`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7`, `fs_v7_raw`, `geo_fs`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt88`, `geo_wkt_raw`

fq_meta*OData metadata document for an ODK Central form.*

Description**[Stable]****Usage**

fq_meta

Format

A list of lists

Details

The OData response for the metadata of an ODK Central form.

This data is kept up to date with the data used in vignettes and package tests. The data is comprised of test records with nonsensical data. The forms used to capture this data are development versions of real-world forms.

Source

See `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")`

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

<code>fq_project_detail</code>	<i>A tibble of project metadata.</i>
--------------------------------	--------------------------------------

Description

[Stable]

Usage

`fq_project_detail`

Format

A tibble of project metadata.

Source

The output of [project_detail](#) run on the project containing the test form `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")`.

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

fq_project_list	<i>A tibble of project metadata.</i>
-----------------	--------------------------------------

Description**[Stable]****Usage**

fq_project_list

Format

A tibble of project metadata.

SourceThe output of [project_list](#) run on all projects on the configured ODK Central server.**See Also**

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

fq_raw	<i>OData submission data for an ODK Central form.</i>
--------	---

Description**[Stable]****Usage**

fq_raw

Format

A list of lists

Details

The OData response for the submissions of an ODK Central form. This form represents a Flora Quadrat, which is a ca 50 by 50 m quadrat of a uniform plant community.

The XML and .odkbuild versions for this form are available as `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")` and `system.file("extdata", "FloraQuadrat04.odkbuild", package = "ruODK")`, respectively.

This data is kept up to date with the data used in vignettes and package tests. The data is comprised of test records with nonsensical data. The forms used to capture this data are development versions of real-world forms.

Source

See `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")`

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

<code>fq_raw_strata</code>	<i>OData submission data for a subgroup of an ODK Central form.</i>
----------------------------	---

Description

[Stable]

Usage

`fq_raw_strata`

Format

A list of lists

Details

The OData response for the subgroup of an ODK Central form.

This subgroup represents vegetation strata as per the NVIS classification. A vegetation stratum is a layer of plants with the same height, and dominated by one or few plant taxa. Plant communities can be made of up to five strata, with two to three being most common.

This data is kept up to date with the data used in vignettes and package tests. The data is comprised of test records with nonsensical data. The forms used to capture this data are development versions of real-world forms.

Source

```
See system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")
```

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

fq_raw_taxa

*OData submission data for a subgroup of an ODK Central form.***Description****[Stable]****Usage**

fq_raw_taxa

Format

A list of lists

Details

The OData response for a subgroup of an ODK Central form.

This subgroup represents an individual plant taxon which is encountered by the enumerators. Typically, one voucher specimen is taken for each distinct encountered plant taxon. A field name is allocated by the enumerators, which can be the proper canonical name (if known) or any other moniker. The voucher specimens are later determined by taxonomic experts, who then provide the real, terminal taxonomic name for a given voucher specimen.

This data is kept up to date with the data used in vignettes and package tests. The data is comprised of test records with nonsensical data. The forms used to capture this data are development versions of real-world forms.

Source

```
See system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")
```

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

<code>fq_submission_list</code>	<i>A tibble of submission metadata.</i>
---------------------------------	---

Description

[Stable]

Usage

`fq_submission_list`

Format

A tibble of submission metadata.

Source

The output of [submission_list](#) run on the test form system. `file("extdata", "FloraQuadrat04.xml", package = "ruODK")`.

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

<code>fq_submissions</code>	<i>A nested list of submission data.</i>
-----------------------------	--

Description

[Stable]

Usage

`fq_submissions`

Format

A nested list of submission data.

Source

The output of `submission_get` run on the test form system.`file("extdata", "FloraQuadrat04.xml", package = "ruODK")` using submission instance IDs from `submission_list`.

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_detail`, `fq_form_list`, `fq_form_schema`, `fq_form_xml`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_svc`, `fq_zip_data`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7`, `fs_v7_raw`, `geo_fs`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt88`, `geo_wkt_raw`

fq_svc

*OData service document for an ODK Central form.***Description**

[Stable]

Usage

```
fq_svc
```

Format

A tibble with one row per submission data endpoint.

Details

The OData response for the metadata of an ODK Central form.

This data is kept up to date with the data used in vignettes and package tests. The data is comprised of test records with nonsensical data. The forms used to capture this data are development versions of real-world forms.

Source

OData service document for `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")`

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_detail`, `fq_form_list`, `fq_form_schema`, `fq_form_xml`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_submissions`, `fq_zip_data`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7`, `fs_v7_raw`, `geo_fs`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt88`, `geo_wkt_raw`

<code>fq_zip_data</code>	<i>A tibble of the main data table of records from a test form.</i>
--------------------------	---

Description

[Stable]

Usage

`fq_zip_data`

Format

A tibble of main records from a test form.

Source

`submission_export` run on the test form `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")`.

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_detail`, `fq_form_list`, `fq_form_schema`, `fq_form_xml`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_submissions`, `fq_svc`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7`, `fs_v7_raw`, `geo_fs`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt88`, `geo_wkt_raw`

<code>fq_zip_strata</code>	<i>A tibble of a repeated sub-group of records from a test form.</i>
----------------------------	--

Description

[Stable]

Usage

`fq_zip_strata`

Format

A tibble of repeated sub-group of records from a test form.

Source

`submission_export` run on the test form `system.file("extdata", "FloraQuadrat04.xml", package = "ruODK")`.

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

`fq_zip_taxa`*A tibble of a repeated sub-group of records from a test form.*

Description

[Stable]

Usage

```
fq_zip_taxa
```

Format

A tibble of repeated sub-group of records from a test form.

Source

`submission_export` run on the test form system.file("extdata", "FloraQuadrat04.xml",
package = "ruODK").

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

`fs_v7`*The parsed XML form_schema of a form from ODK Central v0.6.*

Description

[Stable]

Usage

```
fs_v7
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 12 rows and 3 columns.

Source

`form_schema_parse(fs_v7_raw)`

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_detail`, `fq_form_list`, `fq_form_schema`, `fq_form_xml`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_submissions`, `fq_svc`, `fq_zip_data`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7_raw`, `geo_fs`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt88`, `geo_wkt_raw`

fs_v7_raw	<i>The unparsed XML form_schema of a form from ODK Central v0.6 as nested list.</i>
-----------	---

Description

[Stable]

Usage

`fs_v7_raw`

Format

An object of class `list` of length 6.

Source

`form_schema(odkc_version = 0.7, parse = FALSE)`

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_detail`, `fq_form_list`, `fq_form_schema`, `fq_form_xml`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_submissions`, `fq_svc`, `fq_zip_data`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7`, `geo_fs`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt88`, `geo_wkt_raw`

`geo_fs`*The form_schema of a form containing geofields in GeoJSON.*

Description**[Stable]****Usage**`geo_fs`**Format**

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 19 rows and 6 columns.

Source

`form_schema` run on the test form system. `file("extdata", "Locations.xml", package = "ruODK")`.

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_detail`, `fq_form_list`, `fq_form_schema`, `fq_form_xml`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_submissions`, `fq_svc`, `fq_zip_data`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7`, `fs_v7_raw`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt88`, `geo_wkt_raw`

`geo_gj`*The parsed submissions of a form containing geofields in GeoJSON.*

Description**[Stable]****Usage**`geo_gj`**Format**

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 1 rows and 59 columns.

Source

`odata_submission_get`(`wkt=FALSE`, `parse=TRUE`) run on the test form system. `file("extdata", "Locations.xml", package = "ruODK")`.

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

geo_gj_raw	<i>The unparsed submissions of a form containing geofields in GeoJSON.</i>
------------	--

Description

[Stable]

Usage

geo_gj_raw

Format

An object of class list of length 2.

Source

`odata_submission_get(wkt=FALSE, parse=FALSE)` run on the test form system.`file("extdata", "Locations.xml", package = "ruODK")`.

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

geo_gj88	<i>The parsed submissions of a form containing geofields in GeoJSON with trailing empty coordinates present.</i>
----------	--

Description

[Stable]

Usage

geo_gj88

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 1 rows and 51 columns.

Details

This issue was fixed in #88. ODK Central versions 0.7 - 0.9 export geotracess and geoshapes with trailing empty coordinates. ruODK has a patch to drop trailing empty coordinates. This Dataset is used to test the patch in ruODK.

Source

`odata_submission_get(wkt=FALSE, parse=TRUE)` run on the test form system. `file("extdata", "Locations.xml", package = "ruODK")`.

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj_raw](#), [geo_wkt](#), [geo_wkt88](#), [geo_wkt_raw](#)

geo_wkt

The parsed submissions of a form containing geofields in WKT.

Description

[Stable]

Usage

`geo_wkt`

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 1 rows and 56 columns.

Source

`odata_submission_get(wkt=TRUE, parse=TRUE)` run on the test form system. `file("extdata", "Locations.xml", package = "ruODK")`.

See Also

Other included: [fq_attachments](#), [fq_data](#), [fq_data_strata](#), [fq_data_taxa](#), [fq_form_detail](#), [fq_form_list](#), [fq_form_schema](#), [fq_form_xml](#), [fq_meta](#), [fq_project_detail](#), [fq_project_list](#), [fq_raw](#), [fq_raw_strata](#), [fq_raw_taxa](#), [fq_submission_list](#), [fq_submissions](#), [fq_svc](#), [fq_zip_data](#), [fq_zip_strata](#), [fq_zip_taxa](#), [fs_v7](#), [fs_v7_raw](#), [geo_fs](#), [geo_gj](#), [geo_gj88](#), [geo_gj_raw](#), [geo_wkt88](#), [geo_wkt_raw](#)

geo_wkt_raw	<i>The unparsed submissions of a form containing geofields in WKT.</i>
-------------	--

Description

[Stable]

Usage

geo_wkt_raw

Format

An object of class list of length 2.

Source

`odata_submission_get(wkt=TRUE, parse=FALSE)` run on the test form system. `file("extdata", "Locations.xml", package = "ruODK")`.

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_detail`, `fq_form_list`, `fq_form_schema`, `fq_form_xml`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_submissions`, `fq_svc`, `fq_zip_data`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7`, `fs_v7_raw`, `geo_fs`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt88`

geo_wkt88	<i>The parsed submissions of a form containing geofields in WKT with trailing empty coordinates present.</i>
-----------	--

Description

[Stable]

Usage

geo_wkt88

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 1 rows and 48 columns.

Details

This issue was fixed in #88. ODK Central versions 0.7 - 0.9 export geotraces and geoshapes with trailing empty coordinates. ruODK has a patch to drop trailing empty coordinates. This Dataset is used to test the patch in ruODK.

Source

`odata_submission_get(wkt=TRUE, parse=TRUE)` run on the test form system. `file("extdata", "Locations.xml", package = "ruODK")`.

See Also

Other included: `fq_attachments`, `fq_data`, `fq_data_strata`, `fq_data_taxa`, `fq_form_detail`, `fq_form_list`, `fq_form_schema`, `fq_form_xml`, `fq_meta`, `fq_project_detail`, `fq_project_list`, `fq_raw`, `fq_raw_strata`, `fq_raw_taxa`, `fq_submission_list`, `fq_submissions`, `fq_svc`, `fq_zip_data`, `fq_zip_strata`, `fq_zip_taxa`, `fs_v7`, `fs_v7_raw`, `geo_fs`, `geo_gj`, `geo_gj88`, `geo_gj_raw`, `geo_wkt`, `geo_wkt_raw`

<code>get_one_attachment</code>	<i>Download one media attachment.</i>
---------------------------------	---------------------------------------

Description

[Stable]

Usage

```
get_one_attachment(
  pth,
  fn,
  src,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  verbose = get_ru_verbose()
)
```

Arguments

<code>pth</code>	A local file path to save the attachment to.
<code>fn</code>	The attachment filename, as per ODK form submission. If NA, no file will be downloaded, but NA will be returned. If the file does not exist in ODK Central, a warning will be emitted.

src	The attachment's download URL, generated by attachment_url . The src must contain the uuid: prefix. Note that the main Submissions table contains the submission id in the field id, whereas nested sub-tables contain the submission id in the field submissions_id.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

This is a helper function used by [attachment_get](#). This function is not vectorised, but mapped by [attachment_get](#) to a tibble of input parameters.

Value

The relative local path to the downloaded attachment or NA.

See Also

Other utilities: [attachment_get\(\)](#), [attachment_link\(\)](#), [attachment_url\(\)](#), [drop_null_coords\(\)](#), [form_schema_parse\(\)](#), [get_one_submission\(\)](#), [get_one_submission_att_list\(\)](#), [get_one_submission_audit\(\)](#), [handle_ru_attachments\(\)](#), [handle_ru_datetimes\(\)](#), [handle_ru_geopoints\(\)](#), [handle_ru_geoshapes\(\)](#), [handle_ru_geotraces\(\)](#), [isodt_to_local\(\)](#), [odata_submission_rectangle\(\)](#), [predict_ruodk_name\(\)](#), [prepend_uuid\(\)](#), [ru_http_request\(\)](#), [split_geopoint\(\)](#), [split_geoshape\(\)](#), [split_geotrace\(\)](#), [strip_uuid\(\)](#), [tidyeval](#), [unnest_all\(\)](#)

Examples

```
## Not run:
# Step 1: Setup ruODK with OData Service URL (has url, pid, fid)
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# Step 2: Construct attachment_url
att_url <- ruODK::attachment_url(
  "uuid:d3bcefea-32a8-4dbc-80ca-4ecb0678e2b0",
```

```

    "filename.jpg"
  )

  # Step 3: Get one attachment
  local_fn <- get_one_attachment("media/filename.jpg", "filename.jpg", att_url)

  # In real life: done in bulk behind the scenes during odata_submission_get()

  ## End(Not run)

```

get_one_submission	<i>Download one submission.</i>
--------------------	---------------------------------

Description

This function is the workhorse for the vectorised function `submission_get`, which gets all submissions for a list of submission IDs.

Usage

```

get_one_submission(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

<code>iid</code>	The instance_id, a UUID, as returned by <code>submission_list</code> .
<code>pid</code>	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code> . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>fid</code>	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: <code>get_default_fid</code> . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>url</code>	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Note this function returns a nested list containing any repeating subgroups. As the presence and length of repeating subgroups is non-deterministic and entirely depends on the completeness of the submission data, we cannot rectangle them any further here. Rectangling requires knowledge of the form schema and the completeness of submission data.

[Stable]

Value

A nested list of submission data.

See Also

<https://docs.getodk.org/central-api-submission-management/#retrieving-submission-xml>

Other utilities: `attachment_get()`, `attachment_link()`, `attachment_url()`, `drop_null_coords()`, `form_schema_parse()`, `get_one_attachment()`, `get_one_submission_att_list()`, `get_one_submission_audit()`, `handle_ru_attachments()`, `handle_ru_datetimes()`, `handle_ru_geopoints()`, `handle_ru_geoshapes()`, `handle_ru_geotraces()`, `isodt_to_local()`, `odata_submission_rectangle()`, `predict_ruodk_name()`, `prepend_uuid()`, `ru_http_request()`, `split_geopoint()`, `split_geoshape()`, `split_geotrace()`, `strip_uuid()`, `tidyeval`, `unnest_all()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# With explicit credentials, see tests
sl <- submission_list()

sub <- get_one_submission(sl$instance_id[[1]])
listviewer::jsonedit(sub)

# The details for one submission depend on the form fields
length(sub)
# > 11

# The items are the field names. Repeated groups have the same name.
names(sub)
```

```
# > "meta"                "encounter_start_datetime" "reporter"
# > "device_id"           "location"                "habitat"
# > "vegetation_structure" "perimeter"                "taxon_encounter"
# > "taxon_encounter"     "encounter_end_datetime"

## End(Not run)
```

get_one_submission_att_list

List all attachments of one submission.

Description

[Stable]

Usage

```
get_one_submission_att_list(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

When a Submission is created, either over the OpenRosa or the REST interface, its XML data is analysed to determine which file attachments it references: these may be photos or video taken as part of the survey, or an audit/timing log, among other things. Each reference is an expected attachment, and these expectations are recorded permanently alongside the Submission. With this subresource, you can list the expected attachments, see whether the server actually has a copy or not, and download, upload, re-upload, or clear binary data for any particular attachment.

You can retrieve the list of expected Submission attachments at this route, along with a boolean flag indicating whether the server actually has a copy of the expected file or not. If the server has a file, you can then append its filename to the request URL to download only that file.

Value

A tibble containing some high-level details of the submission attachments. One row per submission attachment, columns are submission attributes:

- * name: The attachment filename, e.g. 12345.jpg
- * exists: Whether the attachment for that submission exists on the server.

See Also

<https://docs.getodk.org/central-api-submission-management/#listing-expected-submission-attachments>

<https://docs.getodk.org/central-api-form-management/#listing-form-attachments>

Other utilities: `attachment_get()`, `attachment_link()`, `attachment_url()`, `drop_null_coords()`, `form_schema_parse()`, `get_one_attachment()`, `get_one_submission()`, `get_one_submission_audit()`, `handle_ru_attachments()`, `handle_ru_datetimes()`, `handle_ru_geopoints()`, `handle_ru_geoshapes()`, `handle_ru_geotraces()`, `isodt_to_local()`, `odata_submission_rectangle()`, `predict_ruodk_name()`, `prepend_uuid()`, `ru_http_request()`, `split_geopoint()`, `split_geoshape()`, `split_geotrace()`, `strip_uuid()`, `tidyeval`, `unnest_all()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

al <- get_one_submission_att_list(sl$instance_id[[1]])
al %>% knitr::kable(.)
```



```
# attachment_list returns a tibble
class(al)
# > c("tbl_df", "tbl", "data.frame")

# Submission attributes are the tibble's columns
names(al)
# > "name" "exists"

## End(Not run)
```

```
get_one_submission_audit
```

Download server audit logs for one submission.

Description

[Experimental]

Usage

```
get_one_submission_audit(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This function is the workhorse for the vectorised function `submission_audit_get`, which gets all server audit logs for a list of submission IDs.

Note this function returns a nested list containing any repeating subgroups. As the presence and length of repeating subgroups is non-deterministic and entirely depends on the completeness of the submission data, we cannot rectangle them any further here. Rectangling requires knowledge of the form schema and the completeness of submission data.

[Stable]

Value

A nested list of submission data.

See Also

<https://docs.getodk.org/central-api-submission-management/#retrieving-audit-logs>

Other utilities: `attachment_get()`, `attachment_link()`, `attachment_url()`, `drop_null_coords()`, `form_schema_parse()`, `get_one_attachment()`, `get_one_submission()`, `get_one_submission_att_list()`, `handle_ru_attachments()`, `handle_ru_datetimes()`, `handle_ru_geopoints()`, `handle_ru_geoshapes()`, `handle_ru_geotraces()`, `isodt_to_local()`, `odata_submission_rectangle()`, `predict_ruodk_name()`, `prepend_uuid()`, `ru_http_request()`, `split_geopoint()`, `split_geoshape()`, `split_geotrace()`, `strip_uuid()`, `tidyeval`, `unnest_all()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# With explicit credentials, see tests
sl <- submission_list()

sub <- get_one_submission_audit(sl$instance_id[[1]])
listviewer::jsonedit(sub)

# The details for one submission depend on the form fields
length(sub)
# > 11
```

```
# The items are the field names. Repeated groups have the same name.
names(sub)
# > "meta"                "encounter_start_datetime" "reporter"
# > "device_id"           "location"                "habitat"
# > "vegetation_structure" "perimeter"          "taxon_encounter"
# > "taxon_encounter"     "encounter_end_datetime"

## End(Not run)
```

handle_ru_attachments *Download and link submission attachments according to a form schema.*

Description

[Stable]

Usage

```
handle_ru_attachments(
  data,
  form_schema,
  local_dir = "media",
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  verbose = get_ru_verbose()
)
```

Arguments

data	Submissions rectangled into a tibble. E.g. the output of ruODK::odata_submission_get(parse = FALSE) %>% ruODK::odata_submission_rectangle()
form_schema	The form_schema for the submissions. E.g. the output of ruODK::form_schema().
local_dir	The local folder to save the downloaded files to, default: "media".
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through ru_setup(pid="..."). See vignette("Setup", package = "ruODK").

fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

For a given tibble of submissions, download and link attachments for all columns which are marked in the form schema as type "binary".

Value

The submissions tibble with all attachments downloaded and linked to a `local_dir`.

See Also

Other utilities: [attachment_get\(\)](#), [attachment_link\(\)](#), [attachment_url\(\)](#), [drop_null_coords\(\)](#), [form_schema_parse\(\)](#), [get_one_attachment\(\)](#), [get_one_submission\(\)](#), [get_one_submission_att_list\(\)](#), [get_one_submission_audit\(\)](#), [handle_ru_datetimes\(\)](#), [handle_ru_geopoints\(\)](#), [handle_ru_geoshapes\(\)](#), [handle_ru_geotraces\(\)](#), [isodt_to_local\(\)](#), [odata_submission_rectangle\(\)](#), [predict_ruodk_name\(\)](#), [prepend_uuid\(\)](#), [ru_http_request\(\)](#), [split_geopoint\(\)](#), [split_geoshape\(\)](#), [split_geotrace\(\)](#), [strip_uuid\(\)](#), [tidyeval](#), [unnest_all\(\)](#)

Examples

```
## Not run:
library(magrittr)
data("fq_raw")
data("fq_form_schema")
t <- tempdir()
fs::dir_ls(t) %>% fs::file_delete()
fq_with_att <- fq_raw %>%
```

```

ruODK::odata_submission_rectangle() %>%
ruODK::handle_ru_attachments(
  form_schema = fq_form_schema,
  local_dir = t,
  pid = ruODK::get_test_pid(),
  fid = ruODK::get_test_fid(),
  url = ruODK::get_test_url(),
  un = ruODK::get_test_un(),
  pw = ruODK::get_test_pw(),
  verbose <- ruODK::get_ru_verbose()
)
# There should be files in local_dir
testthat::expect_true(fs::dir_ls(t) %>% length() > 0)

## End(Not run)

```

handle_ru_datetimes	<i>Parse datetimes of submission data according to a form schema.</i>
---------------------	---

Description

[Stable]

Usage

```

handle_ru_datetimes(
  data,
  form_schema,
  orders = get_default_orders(),
  tz = get_default_tz(),
  verbose = get_ru_verbose()
)

```

Arguments

data	Submissions rectangled into a tibble. E.g. the output of ruODK::odata_submission_get(parse = FALSE) %>% ruODK::odata_submission_rectangle()
form_schema	The form_schema for the submissions. E.g. the output of ruODK::form_schema().
orders	(vector of character) Orders of datetime elements for lubridate. Default: c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd").
tz	A timezone to convert dates and times to. Read vignette("setup", package = "ruODK") to learn how ruODK's timezone can be set globally or per function.
verbose	Whether to display debug messages or not. Read vignette("setup", package = "ruODK") to learn how ruODK's verbosity can be set globally or per function.

Details

For a given tibble of submissions, parse all columns which are marked in the form schema as type "date" or "dateTime" using a set of lubridate orders and a given timezone.

Value

The submissions tibble with all date/dateTime columns mutated as lubridate datetimes.

See Also

Other utilities: `attachment_get()`, `attachment_link()`, `attachment_url()`, `drop_null_coords()`, `form_schema_parse()`, `get_one_attachment()`, `get_one_submission()`, `get_one_submission_att_list()`, `get_one_submission_audit()`, `handle_ru_attachments()`, `handle_ru_geopoints()`, `handle_ru_geoshapes()`, `handle_ru_geotraces()`, `isodt_to_local()`, `odata_submission_rectangle()`, `predict_ruodk_name()`, `prepend_uuid()`, `ru_http_request()`, `split_geopoint()`, `split_geoshape()`, `split_geotrace()`, `strip_uuid()`, `tidyeval`, `unnest_all()`

Examples

```
## Not run:
library(magrittr)
data("fq_raw")
data("fq_form_schema")

fq_with_dates <- fq_raw %>%
  ruODK::odata_submission_rectangle() %>%
  ruODK::handle_ru_datetimes(form_schema = fq_form_schema)

dplyr::glimpse(fq_with_dates)

## End(Not run)
```

handle_ru_geopoints	<i>Split all geopoints of a submission tibble into their components.</i>
---------------------	--

Description

[Stable]

Usage

```
handle_ru_geopoints(data, form_schema, wkt = FALSE, verbose = get_ru_verbose())
```

Arguments

data	Submissions rectangled into a tibble. E.g. the output of <code>ruODK::odata_submission_get(parse = FALSE) %>% ruODK::odata_submission_rectangle()</code>
form_schema	The form_schema for the submissions. E.g. the output of <code>ruODK::form_schema()</code> .
wkt	Whether geofields are GeoJSON (if FALSE) or WKT strings (if TRUE), default: FALSE.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

For a given tibble of submissions, find all columns which are listed in the form schema as type `geopoint`, and extract their components. Extracted components are longitude (X), latitude (Y), altitude (Z, where given), and accuracy (M, where given).

The original column is retained to allow parsing into other spatially enabled formats.

Value

The submissions tibble with all geopoints retained in their original format, plus columns of their coordinate components as provided by `split_geopoint`.

See Also

Other utilities: `attachment_get()`, `attachment_link()`, `attachment_url()`, `drop_null_coords()`, `form_schema_parse()`, `get_one_attachment()`, `get_one_submission()`, `get_one_submission_att_list()`, `get_one_submission_audit()`, `handle_ru_attachments()`, `handle_ru_datetimes()`, `handle_ru_geoshapes()`, `handle_ru_geotraces()`, `isodt_to_local()`, `odata_submission_rectangle()`, `predict_ruodk_name()`, `prepend_uuid()`, `ru_http_request()`, `split_geopoint()`, `split_geoshape()`, `split_geotrace()`, `strip_uuid()`, `tidyeval`, `unnest_all()`

Examples

```
library(magrittr)
data("geo_fs")
data("geo_gj_raw")
data("geo_wkt_raw")

# GeoJSON
geo_gj_parsed <- geo_gj_raw %>%
  ruODK::odata_submission_rectangle(form_schema = geo_fs) %>%
  ruODK::handle_ru_geopoints(form_schema = geo_fs, wkt = FALSE)

dplyr::glimpse(geo_gj_parsed)

# WKT
geo_wkt_parsed <- geo_wkt_raw %>%
```

```

ruODK::odata_submission_rectangle(form_schema = geo_fs) %>%
ruODK::handle_ru_geopoints(form_schema = geo_fs, wkt = TRUE)

dplyr::glimpse(geo_wkt_parsed)

```

handle_ru_geoshapes	<i>Split all geoshapes of a submission tibble into their components.</i>
---------------------	--

Description

[Stable]

Usage

```

handle_ru_geoshapes(
  data,
  form_schema,
  wkt = FALSE,
  odkc_version = get_default_odkc_version(),
  verbose = get_ru_verbose()
)

```

Arguments

data	Submissions rectangled into a tibble. E.g. the output of <pre>ruODK::odata_submission_get(parse = FALSE) %>% ruODK::odata_submission_rectangle(form_schema = ...)</pre>
form_schema	The form_schema for the submissions. E.g. the output of <code>ruODK::form_schema()</code> .
wkt	Whether geofields are GeoJSON (if FALSE) or WKT strings (if TRUE), default: FALSE.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

For a given tibble of submissions, find all columns which are listed in the form schema as type geoshape, and extract their components. Extracted components are longitude (X), latitude (Y), altitude (Z, where given), and accuracy (M, where given) of the first point of the geoshape.

The original column is retained to allow parsing into other spatially enabled formats.

Value

The submissions tibble with all geoshapes retained in their original format, plus columns of their first point's coordinate components as provided by [split_geoshape](#).

See Also

Other utilities: [attachment_get\(\)](#), [attachment_link\(\)](#), [attachment_url\(\)](#), [drop_null_coords\(\)](#), [form_schema_parse\(\)](#), [get_one_attachment\(\)](#), [get_one_submission\(\)](#), [get_one_submission_att_list\(\)](#), [get_one_submission_audit\(\)](#), [handle_ru_attachments\(\)](#), [handle_ru_datetimes\(\)](#), [handle_ru_geopoints\(\)](#), [handle_ru_geotraces\(\)](#), [isodt_to_local\(\)](#), [odata_submission_rectangle\(\)](#), [predict_ruodk_name\(\)](#), [prepend_uuid\(\)](#), [ru_http_request\(\)](#), [split_geopoint\(\)](#), [split_geoshape\(\)](#), [split_geotrace\(\)](#), [strip_uuid\(\)](#), [tidyeval](#), [unnest_all\(\)](#)

Examples

```
## Not run:
library(magrittr)
data("geo_fs")
data("geo_wkt_raw")
data("geo_gj_raw")

# GeoJSON
geo_gj_parsed <- geo_gj_raw %>%
  ruODK::odata_submission_rectangle(form_schema = geo_fs) %>%
  ruODK::handle_ru_geoshapes(form_schema = geo_fs, wkt = FALSE)

dplyr::glimpse(geo_gj_parsed)

# WKT
geo_wkt_parsed <- geo_wkt_raw %>%
  ruODK::odata_submission_rectangle(form_schema = geo_fs) %>%
  ruODK::handle_ru_geoshapes(form_schema = geo_fs, wkt = TRUE)

dplyr::glimpse(geo_wkt_parsed)

## End(Not run)
```

handle_ru_geotraces	<i>Split all geotraces of a submission tibble into their components.</i>
---------------------	--

Description

[Stable]

Usage

```
handle_ru_geotraces(
  data,
  form_schema,
```

```

wkt = FALSE,
odkc_version = get_default_odkc_version(),
verbose = get_ru_verbose()
)

```

Arguments

data	Submissions rectangled into a tibble. E.g. the output of <pre> ruODK::odata_submission_get(parse = FALSE) %>% ruODK::odata_submission_rectangle(form_schema = ...) </pre>
form_schema	The form_schema for the submissions. E.g. the output of <code>ruODK::form_schema()</code> .
wkt	Whether geofields are GeoJSON (if FALSE) or WKT strings (if TRUE), default: FALSE.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

For a given tibble of submissions, find all columns which are listed in the form schema as type geotrace, and extract their components. Extracted components are longitude (X), latitude (Y), altitude (Z, where given), and accuracy (M, where given) of the first point of the geotrace.

The original column is retained to allow parsing into other spatially enabled formats.

Value

The submissions tibble with all geotraces retained in their original format, plus columns of their first point's coordinate components as provided by `split_geotrace`.

See Also

Other utilities: `attachment_get()`, `attachment_link()`, `attachment_url()`, `drop_null_coords()`, `form_schema_parse()`, `get_one_attachment()`, `get_one_submission()`, `get_one_submission_att_list()`, `get_one_submission_audit()`, `handle_ru_attachments()`, `handle_ru_datetimes()`, `handle_ru_geopoints()`, `handle_ru_geoshapes()`, `isodt_to_local()`, `odata_submission_rectangle()`, `predict_ruodk_name()`, `prepend_uuid()`, `ru_http_request()`, `split_geopoint()`, `split_geoshape()`, `split_geotrace()`, `strip_uuid()`, `tidyeval`, `unnest_all()`

Examples

```
## Not run:
library(magrittr)
data("geo_fs")
data("geo_wkt_raw")
data("geo_gj_raw")

# GeoJSON
geo_gj_parsed <- geo_gj_raw %>%
  ruODK::odata_submission_rectangle(form_schema = geo_fs) %>%
  ruODK::handle_ru_geotraces(form_schema = geo_fs, wkt = FALSE)

dplyr::glimpse(geo_gj_parsed)

# WKT
geo_wkt_parsed <- geo_wkt_raw %>%
  ruODK::odata_submission_rectangle(form_schema = geo_fs) %>%
  ruODK::handle_ru_geotraces(form_schema = geo_fs, wkt = TRUE)

dplyr::glimpse(geo_wkt_parsed)

## End(Not run)
```

odata_entitylist_data_get

Get the Data Document from the OData Dataset Service.

Description**[Experimental]****Usage**

```
odata_entitylist_data_get(
  pid = get_default_pid(),
  did = "",
  query = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
query	An optional named list of query parameters, e.g. <pre>list("\$filter" = "__system/createdAt le 2024-11-05", "\$orderby" = "__system/creatorId ASC, __system/conflict DESC", "\$top" = "100", "\$skip" = "3", "\$count" = "true")</pre> No validation is conducted by ruODK on the query list prior to passing it to ODK Central. If omitted, no filter query is sent. Note that the behaviour of this parameter differs from the implementation of <code>odata_submission_get()</code> in that query here accepts a list of all possible OData query parameters and <code>odata_submission_get()</code> offers individual function parameters matching supported OData query parameters. Default: NULL
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.

tz A timezone to convert dates and times to.
 Read vignette("setup", package = "ruODK") to learn how ruODK's timezone can be set globally or per function.

Details

All the Entities in a Dataset.

The `$top` and `$skip` querystring parameters, specified by OData, apply limit and offset operations to the data, respectively.

The `$count` parameter, also an OData standard, will annotate the response data with the total row count, regardless of the scoping requested by `$top` and `$skip`. If `$top` parameter is provided in the request then the response will include `@odata.nextLink` that you can use as is to fetch the next set of data. As of ODK Central v2023.4, `@odata.nextLink` contains a `$skiptoken` (an opaque cursor) to better paginate around deleted Entities.

The `$filter` querystring parameter can be used to filter certain data fields in the system-level schema, but not the Dataset properties. The operators `lt`, `le`, `eq`, `ne`, `ge`, `gt`, `not`, `and`, and `or` and the built-in functions `now`, `year`, `month`, `day`, `hour`, `minute`, `second` are supported.

The fields you can query against are as follows:

Entity Metadata: OData Field Name Entity Creator Actor ID: `__system/creatorId` Entity Timestamp: `__system/createdAt` Entity Update Timestamp: `__system/updatedAt` Entity Conflict: `__system/conflict`

Note that `createdAt` and `updatedAt` are time components. This means that any comparisons you make need to account for the full time of the Entity. It might seem like `$filter=__system/createdAt le 2020-01-31` would return all results on or before 31 Jan 2020, but in fact only Entities made before midnight of that day would be accepted. To include all of the month of January, you need to filter by either `$filter=__system/createdAt le 2020-01-31T23:59:59.999Z` or `$filter=__system/createdAt lt 2020-02-01`. Remember also that you can query by a specific timezone.

Please see the OData documentation on `$filter` [operations](#) and [functions](#) for more information.

The `$select` query parameter will return just the fields you specify and is supported on `__id`, `__system`, `__system/creatorId`, `__system/createdAt` and `__system/updatedAt`, as well as on user defined properties.

The `$orderby` query parameter will return Entities sorted by different fields, which come from the same list used by `$filter`, as noted above. The order can be specified as `ASC` (ascending) or `DESC` (descending), which are case-insensitive. Multiple sort expressions can be used together, separated by commas, e.g. `$orderby=__system/creatorId ASC, __system/conflict DESC`.

As the vast majority of clients only support the JSON OData format, that is the only format ODK Central offers.

Value

An S3 class `odata_entitylist_data_get` with two list items:

- `context` The URL for the OData metadata document
- `value` A tibble of EntitySets available in this Entity List, with names cleaned by `janitor::clean_names()` and unnested list columns (`__system`).

See Also

<https://docs.getodk.org/central-api-odata-endpoints/#id3>

http://docs.oasis-open.org/odata/odata/v4.01/odata-v4.01-part1-protocol.html#_Toc31358948

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_metadata_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

ds <- entitylist_list(pid = get_default_pid())

ds1 <- odata_entitylist_data_get(pid = get_default_pid(), did = ds$name[1])

ds1
ds1$context
ds1$value

qry <- list(
  "$filter" = "__system/createdAt le 2024-11-05",
  "$orderby" = "__system/creatorId ASC, __system/conflict DESC",
  "$top" = "100",
  "$skip" = "3",
  "$count" = "true"
)
ds2 <- odata_entitylist_data_get(
  pid = get_default_pid(),
  did = ds$name[1],
  query = qry
)

ds2

## End(Not run)
```

odata_entitylist_metadata_get

Get the Metadata Document from the OData Dataset Service.

Description

[Experimental]

Usage

```
odata_entitylist_metadata_get(
  pid = get_default_pid(),
  did = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

The Metadata Document describes, in **EDMX CSDL**, the schema of all the data you can retrieve from the OData Dataset Service in question. Essentially, these are the Dataset properties, or the schema of each Entity, translated into the OData format.

Value

An S3 class `odata_entitylist_metadata_get` and `list` containing the Metadata document following the DDMX CSDL standard

- `version` The EDMX version, e.g. "4.0"
- `complex_types`
- `entity_types`
- `containers`

See Also

<https://docs.getodk.org/central-api-odata-endpoints/#id2>

<https://docs.oasis-open.org/odata/odata-csdl-xml/v4.01/odata-csdl-xml-v4.01.html>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_service_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

ds <- entitylist_list(pid = get_default_pid())

dm1 <- odata_entitylist_metadata_get(pid = get_default_pid(), did = ds$name[1])

# Overview
print(dm1)

# Get all property names for an Entity type
names(dm1$entity_types$Entities$properties)

# Check what properties are non-filterable
dm1$containers$trees$entity_sets$Entities$capabilities

# Get complex type definitions
dm1$complex_types$metadata$properties

## End(Not run)
```

odata_entitylist_service_get

Get the Service Document from the OData Dataset Service.

Description

[Experimental]

Usage

```
odata_entitylist_service_get(
  pid = get_default_pid(),
  did = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
did	(chr) The name of the Entity List, internally called Dataset. The function will error if this parameter is not given. Default: "".
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.

odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page /version.txt. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

ODK Central presents one OData service for every Dataset (Entity List) as a way to get an OData feed of Entities. To access the OData service, add `.svc` to the resource URL for the given Dataset (Entity List).

The Service Document provides a link to the main source of information in this OData service: the list of Entities in this Dataset, as well as the Metadata Document describing the schema of this information.

This document is available only in JSON format.

Value

An S3 class `odata_entitylist_service_get` with two list items:

- `context` The URL for the OData metadata document
- `value` A tibble of EntitySets available in this EntityList

See Also

<https://docs.getodk.org/central-api-odata-endpoints/#odata-dataset-service>

Other entity-management: `entity_audits()`, `entity_bulk_delete()`, `entity_bulk_restore()`, `entity_changes()`, `entity_create()`, `entity_creators()`, `entity_delete()`, `entity_detail()`, `entity_geodata()`, `entity_geojson()`, `entity_list()`, `entity_restore()`, `entity_update()`, `entity_versions()`, `entitylist_create()`, `entitylist_delete()`, `entitylist_detail()`, `entitylist_download()`, `entitylist_list()`, `entitylist_property_create()`, `entitylist_property_delete()`, `entitylist_trash_download()`, `entitylist_update()`, `odata_entitylist_data_get()`, `odata_entitylist_metadata_get()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

ds <- entitylist_list(pid = get_default_pid())
```

```

ds1 <- odata_entitylist_service_get(pid = get_default_pid(), did = ds$name[1])

ds1
ds1$context
ds1$value

## End(Not run)

```

odata_metadata_get	<i>Retrieve metadata from an OData URL ending in .svc as list of lists.</i>
--------------------	---

Description

[Stable]

Usage

```

odata_metadata_get(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See vignette("Setup", package = "ruODK").

retries The number of attempts to retrieve a web resource.
 This parameter is given to `RETRY(times = retries)`.
 Default: 3.

Value

A nested list containing Edmx (Dataset schema definition) and .attrs (Version).

See Also

<https://docs.getodk.org/central-api-odata-endpoints/#metadata-document>

Other odata-api: `odata_service_get()`, `odata_submission_get()`, `warn_on_schema_drift()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

meta <- odata_metadata_get()
listviewer::jsonedit(meta)

## End(Not run)
```

<code>odata_service_get</code>	<i>Retrieve service metadata from an OData URL ending in .svc as tibble.</i>
--------------------------------	--

Description

[Stable]

Usage

```
odata_service_get(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.

Value

A tibble with one row per submission data endpoint. Columns: name, kind, url.

See Also

<https://docs.getodk.org/central-api-odata-endpoints/#service-document>

Other odata-api: [odata_metadata_get\(\)](#), [odata_submission_get\(\)](#), [warn_on_schema_drift\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "....svc", un = "me@email.com", pw = "...")

svc <- odata_service_get()
svc

## End(Not run)
```

odata_submission_get	<i>Retrieve and rectangle form submissions, parse dates, geopoints, download and link attachments.</i>
----------------------	--

Description

[Stable]

Usage

```
odata_submission_get(  
  table = "Submissions",  
  skip = NULL,  
  top = NULL,  
  count = FALSE,  
  wkt = FALSE,  
  expand = FALSE,  
  filter = NULL,  
  parse = TRUE,  
  download = TRUE,  
  names_sep = "_",  
  clean_names = TRUE,  
  orders = get_default_orders(),  
  local_dir = "media",  
  pid = get_default_pid(),  
  fid = get_default_fid(),  
  url = get_default_url(),  
  un = get_default_un(),  
  pw = get_default_pw(),  
  odkc_version = get_default_odkc_version(),  
  tz = get_default_tz(),  
  retries = get_retries(),  
  verbose = get_ru_verbose()  
)
```

Arguments

table	The submission EntityType, or in plain words, the table name. Default: Submissions (the main table). Change to Submissions.GROUP_NAME for repeating form groups. The group name can be found through odata_service_get .
skip	The number of rows to be omitted from the results. Example: 10, default: NA (none skipped).
top	The number of rows to return. Example: 100, default: NA (all returned).
count	If TRUE, an @odata.count property will be returned in the response from ODK Central. Default: FALSE.

wkt	If TRUE, geospatial data will be returned as WKT (Well Known Text) strings. Default: FALSE, returns GeoJSON structures. Note that accuracy is only returned through GeoJSON.
expand	If TRUE, all subtables will be expanded and included with column names containing the number of the repeat, the group name, and the field name. This parameter is supported from ODK Central v1.2 onwards and will be ignored on earlier versions of ODK Central. The version is inferred from the parameter <code>odkc_version</code> . Default: FALSE.
filter	If provided, will filter responses to those matching the query. For an <code>odkc_version</code> below 1.1, this parameter will be discarded. As of ODK Central v1.5, the fields <code>system/submitterId</code> , <code>system/submissionDate</code> , <code>__system/updatedAt</code> and <code>__system/reviewState</code> are available to reference. The operators <code>lt</code> , <code>lte</code> , <code>eq</code> , <code>neq</code> , <code>gte</code> , <code>gt</code> , <code>not</code> , <code>and</code> , and <code>or</code> are supported, and the built-in functions <code>now</code> , <code>year</code> , <code>month</code> , <code>day</code> , <code>hour</code> , <code>minute</code> , <code>second</code> . <code>ruODK</code> does not validate the query string given to <code>filter</code> . It is highly recommended to refer to the ODK Central API documentation as well as the OData spec on filters . for filter options and capabilities.
parse	Whether to parse submission data based on form schema. Dates and datetimes will be parsed into local time. Attachments will be downloaded, and the field updated to the local file path. Point locations will be split into components; GeoJSON (<code>wkt=FALSE</code>) will be split into latitude, longitude, altitude and accuracy (with anonymous field names), while WKT will be split into longitude, latitude, and altitude (missing accuracy) prefixed by the original field name. See details for the handling of geotracers and geoshapes. Default: TRUE.
download	Whether to download attachments to <code>local_dir</code> or not. If in the future ODK Central supports hot-linking attachments, this parameter will replace attachment file names with their fully qualified attachment URL. Default: TRUE.
names_sep	Used internally in <code>tidyr::unnest_wider(names_sep=names_sep)</code> when unnesting form groups. By default, nested form fields are prefixed with the group name and separated by <code>names_sep</code> . To omit the group name as prefix, use <code>names_sep=NULL</code> . Default: <code>"_"</code>
clean_names	Whether to run <code>janitor::clean_names()</code> after unnesting form groups. Set <code>clean_names=FALSE</code> to preserve any non-standard <code>names_sep</code> . Default: TRUE.
orders	(vector of character) Orders of datetime elements for <code>lubridate</code> . Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
local_dir	The local folder to save the downloaded files to, default: <code>"media"</code> .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. <code>"build_Spotlighting-0-8_1559885147"</code> . Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code>'s timezone can be set globally or per function.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code>'s verbosity can be set globally or per function.

Details

[odata_submission_get](#) downloads submissions from (default) the main form group (submission table) including any non-repeating form groups, or from any other table as specified by parameter `table`.

With parameter `parse=TRUE` (default), submission data is parsed into a tibble. Any fields of type `dateTime` or `date` are parsed into dates, with an optional parameter `tz` to specify the local timezone.

A parameter `local_dir` (default: `media`) specifies a local directory for downloaded attachment files. Already existing, previously downloaded attachments will be retained.

With parameter `wkt=TRUE`, spatial fields will be returned as WKT, rather than GeoJSON. In addition, fields of type `geopoint` will be split into latitude, longitude, and altitude, prefixed with the original field name. E.g. a field `start_location` of type `geopoint` will be split into `start_location_latitude`, `start_location_longitude`, and `start_location_altitude`. The field name prefix will allow multiple fields of type `geopoint` to be split into their components without naming conflicts.

Geotraces (lines) and gepshapes (polygons) will be retained in their original format, plus columns of their first point's coordinate components as provided by [split_geotrace](#) and [split_geoshape](#), respectively.

Entirely unpopulated form fields, as well as notes and form groups, will be excluded from the resulting tibble. Submitting at least one complete form instance will prevent the accidental exclusion of an otherwise mostly empty form field.

Submissions keep the values entered under the Form version used at entry. The OData feed, however, uses only the current Form definition. Fields that a previous version held at another path (removed, renamed, or moved to another group or repeat) do not appear in the download. `odata_submission_get` warns when previous published versions hold such fields. To include fields from all versions, use `submission_export` with `deleted_fields = TRUE`.

The only remaining manual step is to optionally join any sub-tables to the master table.

The parameter `verbose` enables diagnostic messages along the download and parsing process.

With parameter `parse=FALSE`, submission data is presented as nested list, which is the R equivalent of the JSON structure returned from the API. From there, `odata_submission_rectangle` can rectangle the data into a tibble, and subsequent lines of `handle_ru_datetimes`, `handle_ru_attachments`, `handle_ru_geopoints`, `handle_ru_geotraces`, and `handle_ru_geoshapes` parse dates, download and link file attachments, and extract coordinates from geofields. ruODK offers this manual and explicit pathway as an option to investigate and narrow down unexpected or unwanted behaviour.

Value

A list of lists.

- `value` contains the submissions as list of lists.
- `@odata.context` is the URL of the metadata.
- `@odata.count` is the total number of rows in the table.

See Also

<https://docs.getodk.org/central-api-odata-endpoints/#odata-form-service>

<https://docs.getodk.org/central-api-odata-endpoints/#data-document>

Other odata-api: `odata_metadata_get()`, `odata_service_get()`, `warn_on_schema_drift()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

form_tables <- ruODK::odata_service_get()
data <- odata_submission_get() # default: main data table
data <- odata_submission_get(table = form_tables$url[1]) # same, explicitly
data_sub1 <- odata_submission_get(table = form_tables$url[2]) # sub-table 1
data_sub2 <- odata_submission_get(table = form_tables$url[3]) # sub-table 2

# Skip one row, return the next 1 rows (top), include total row count
data <- odata_submission_get(
  table = form_tables$url[1],
  skip = 1,
  top = 1,
```

```

    count = TRUE
  )

  # Filter submissions
  data <- odata_submission_get(
    table = form_tables$url[1],
    filter = "year(__system/submissionDate) lt year(now())"
  )
  data <- odata_submission_get(
    table = form_tables$url[1],
    filter = "year(__system/submissionDate) lt 2020"
  )

  # To include all of the month of January, you need to filter by either
  # filter = "__system/submissionDate le 2020-01-31T23:59:59.999Z"
  # or
  # filter = "__system/submissionDate lt 2020-02-01".
  # Instead of timezone UTC ("Z"), you can also filter by any other timezone.

  ## End(Not run)

```

odata_submission_rectangle

Rectangle the output of `odata_submission_get`(parse=FALSE) into a tidy tibble and unnest all levels.

Description

[Stable]

Usage

```

odata_submission_rectangle(
  data,
  names_repair = "universal",
  names_sep = "_",
  form_schema = NULL,
  clean_names = TRUE,
  verbose = get_ru_verbose()
)

```

Arguments

data	A nested list of lists as given by <code>odata_submission_get</code> .
names_repair	The argument <code>names_repair</code> for <code>tidyr::unnest_wider</code> , default: "universal".
names_sep	The argument <code>names_sep</code> for <code>tidyr::unnest_wider</code> , default: "_". Un-nested variables inside a list column will be prefixed by the list column name, separated by <code>names_sep</code> . This avoids unsightly repaired names such as <code>latitude...1</code> .

form_schema	An optional form_schema, like the output of form_schema . If a form schema is supplied, location fields will not be unnested. While WKT location fields contain plain text and will never be unnested, GeoJSON location fields would cause errors during unnesting.
clean_names	Whether to run <code>janitor::clean_names()</code> . Set <code>clean_names=FALSE</code> to preserve any non-standard names_sep. Default: TRUE.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

This function cleans names with `janitor::clean_names()` and drops the prefix `value_`.

Value

The submissions as un-nested tibble

See Also

Other utilities: [attachment_get\(\)](#), [attachment_link\(\)](#), [attachment_url\(\)](#), [drop_null_coords\(\)](#), [form_schema_parse\(\)](#), [get_one_attachment\(\)](#), [get_one_submission\(\)](#), [get_one_submission_att_list\(\)](#), [get_one_submission_audit\(\)](#), [handle_ru_attachments\(\)](#), [handle_ru_datetimes\(\)](#), [handle_ru_geopoints\(\)](#), [handle_ru_geoshapes\(\)](#), [handle_ru_geotraces\(\)](#), [isodt_to_local\(\)](#), [predict_ruodk_name\(\)](#), [prepend_uuid\(\)](#), [ru_http_request\(\)](#), [split_geopoint\(\)](#), [split_geoshape\(\)](#), [split_geotrace\(\)](#), [strip_uuid\(\)](#), [tidyeval](#), [unnest_all\(\)](#)

Examples

```
## Not run:
# Using canned data
data_parsed <- odata_submission_rectangle(fq_raw, verbose = TRUE)
# Field "device_id" is known part of fq_raw
testthat::expect_equal(
  data_parsed$device_id[[1]],
  fq_raw$value[[1]]$device_id
)

# fq_raw has two submissions
testthat::expect_equal(length(fq_raw$value), nrow(data_parsed))

## End(Not run)
```

odata_svc_parse	<i>Retrieve URL, project ID, and form ID from an ODK Central OData service URL.</i>
-----------------	---

Description**[Stable]****Usage**

```
odata_svc_parse(svc)
```

Arguments

svc	(character) The OData service URL of a form as provided by the ODK Central form submissions tab. Example: "https://URL/v1/projects/PID/forms/FID.svc"
-----	---

Value

A named list with three components (all of type character):

- url The ODK Central base URL.
- pid The project ID.
- fid The form ID.

See Also

Other ru_settings: [parse_odkc_version\(\)](#), [ru_settings\(\)](#), [ru_setup\(\)](#), [semver_gt\(\)](#), [semver_lt\(\)](#), [yell_if_error\(\)](#), [yell_if_missing\(\)](#)

parse_odkc_version	<i>Parse a given ODK Central version string or number into a semver.</i>
--------------------	--

Description**[Stable]****Usage**

```
parse_odkc_version(v, env_var = "ODKC_VERSION")
```

Arguments

v	A string (e.g. "ODKC_VERSION=") or number (e.g. 0.8, 1.5) of a complete or partial semver, or a semver of class "svlist".
env_var	A string indicating which environment variable was targeted. This is used in the warning to advise which environment variable to update. If set to an empty string, the warning message is suppressed. Default: "ODKC_VERSION".

Details

Past versions of ruODK advised to set ODKC_VERSION to a floating point number indicating major and minor versions, e.g. ODKC_VERSION=0.7 or ODKC_VERSION=1.5.

ODK Central has since switched to semantic versioning, e.g. ODKC_VERSION=ODKC_VERSION=.

To preserve backwards compatibility, ruODK handles both formats gracefully, but emits a helpful warning to update the version string if the older format is detected, or the version string is missing the minor or patch version.

Value

semver::svlist The version as semver of major, minor, and patch.

See Also

Other ru_settings: [odata_svc_parse\(\)](#), [ru_settings\(\)](#), [ru_setup\(\)](#), [semver_gt\(\)](#), [semver_lt\(\)](#), [yell_if_error\(\)](#), [yell_if_missing\(\)](#)

Examples

```
parse_odkc_version("1.2.3")

# Warn: too short
parse_odkc_version("1")
parse_odkc_version("1.2")

# Warn: too long
parse_odkc_version("1.2.3.4")

# Warn: otherwise invalid
parse_odkc_version("1.2.")
parse_odkc_version(".2.3")
```

project_assignment_actors

List all Actors assigned some Project Role.

Description

[Experimental]

Usage

```
project_assignment_actors(
  pid = get_default_pid(),
  role_id,
  url = get_default_url(),
```

```

un = get_default_un(),
pw = get_default_pw(),
retries = get_retries(),
odkc_version = get_default_odkc_version(),
orders = get_default_orders(),
tz = get_default_tz()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
role_id	(character or numeric) The Role ID, typically the integer ID of the Role. A Role system name can also be supplied if the Role has one.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". <code>ruODK</code> uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how <code>ruODK</code>'s timezone can be set globally or per function.

Details

Given a Role, this endpoint lists all Actors that have been assigned that Role upon this particular Project. This endpoint requires ODK Central v0.5 or later.

Value

A tibble with one row for each Actor assigned the Role on the Project as per the ODK Central API docs. Column names are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-project-management/#listing-all-actors-assigned-some-project-r>

Other project-management: [form_assignment_role_list\(\)](#), [project_assignment_grant\(\)](#), [project_assignment_rev](#), [project_create\(\)](#), [project_delete\(\)](#), [project_detail\(\)](#), [project_enable_encryption\(\)](#), [project_list\(\)](#), [project_replace\(\)](#), [project_update\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

pa <- project_assignment_actors(role_id = "manager")

pa |> knitr::kable()

## End(Not run)
```

project_assignment_grant

Assign an Actor to a Project Role.

Description

[Experimental]

Usage

```
project_assignment_grant(
  pid = get_default_pid(),
  role_id,
  actor_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
role_id	(character or numeric) The Role ID, typically the integer ID of the Role. A Role system name can also be supplied if the Role has one.
actor_id	(numeric) The integer ID of the Actor.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

Details

Given a Role and an Actor, this endpoint assigns that Role to that Actor for this particular Project. No request body data is required. This endpoint requires ODK Central v0.5 or later.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-project-management/#assigning-an-actor-to-a-project-role>

Other project-management: [form_assignment_role_list\(\)](#), [project_assignment_actors\(\)](#), [project_assignment_revoke\(\)](#), [project_create\(\)](#), [project_delete\(\)](#), [project_detail\(\)](#), [project_enable_encryption\(\)](#), [project_list\(\)](#), [project_replace\(\)](#), [project_update\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

project_assignment_grant(role_id = "manager", actor_id = 14)
# > $success
# > [1] TRUE

## End(Not run)
```

project_assignment_list

List all Assignments of one Project.

Description

[Experimental]

Usage

```
project_assignment_list(
  pid = get_default_pid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This lists every Assignment upon the given Project in the form of actor_id/role_id pairs. Assigning an Actor a Role grants that Actor the Role's verbs anywhere within the Project.

Value

A tibble with one row per Assignment and the columns actor_id and role_id.

See Also

<https://docs.getodk.org/central-api-project-management/#listing-all-project-assignments>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

al <- project_assignment_list()

al |> knitr::kable()

## End(Not run)
```

project_assignment_revoke

Revoke a Project Role Assignment from an Actor.

Description

[Experimental]

Usage

```
project_assignment_revoke(
  pid = get_default_pid(),
  role_id,
  actor_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
role_id	(character or numeric) The Role ID, typically the integer ID of the Role. A Role system name can also be supplied if the Role has one.
actor_id	(numeric) The integer ID of the Actor.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

Details

Given a Role and an Actor, this endpoint unassigns that Role from that Actor for this particular Project. This endpoint requires ODK Central v0.5 or later.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-project-management/#revoking-a-project-role-assignment-from-an>

Other project-management: [form_assignment_role_list\(\)](#), [project_assignment_actors\(\)](#), [project_assignment_grant\(\)](#), [project_create\(\)](#), [project_delete\(\)](#), [project_detail\(\)](#), [project_enable_encrypt\(\)](#), [project_list\(\)](#), [project_replace\(\)](#), [project_update\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

project_assignment_revoke(role_id = "manager", actor_id = 14)
# > $success
# > [1] TRUE

## End(Not run)
```

project_create	Create a new project.
----------------	-----------------------

Description

[Experimental]

Usage

```
project_create(
  name,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

name	The desired name of the project. Can contain whitespace.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Value

A tibble with one row holding the new Project's metadata as columns, as per ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-project-management/#creating-a-project>

Other project-management: [form_assignment_role_list\(\)](#), [project_assignment_actors\(\)](#), [project_assignment_grant\(\)](#), [project_assignment_revoke\(\)](#), [project_delete\(\)](#), [project_detail\(\)](#), [project_enable_encryption\(\)](#), [project_list\(\)](#), [project_replace\(\)](#), [project_update\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

p <- project_create("Test Project")
knitr::kable(p)

# project_create returns a tibble
class(p)
# > "tbl_df" "tbl" "data.frame"

# columns are project metadata
names(p)
# > "id" "name" "archived"

## End(Not run)
```

project_delete

Delete a Project.

Description

[Experimental]

Usage

```
project_delete(
  pid = get_default_pid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code> . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Deleting a Project removes it from the management interface and makes it permanently inaccessible. Do not do this unless you are certain you will never need any of its data again.

Value

A list with a single element `success` (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-project-management/#deleting-a-project>

Other project-management: `form_assignment_role_list()`, `project_assignment_actors()`, `project_assignment_grant()`, `project_assignment_revoke()`, `project_create()`, `project_detail()`, `project_enable_encryption()`, `project_list()`, `project_replace()`, `project_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

p <- project_create("Temporary Project")

project_delete(p$id)
# > $success
# > [1] TRUE

## End(Not run)
```

project_detail	List all details of one project.
----------------	----------------------------------

Description

While the API endpoint will return all details for one project, [project_detail](#) will fail with incorrect or missing authentication.

Usage

```
project_detail(
  pid = get_default_pid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.

Details

[Stable]

Value

A tibble with exactly one row for the project and all project metadata as columns as per ODK Central API docs. Column names are renamed from ODK's camelCase to snake_case. Values differ to values returned by ODK Central API:

- archived: FALSE (if NULL) else TRUE
- dates: NA if NULL

See Also

<https://docs.getodk.org/central-api-project-management/#getting-project-details>

Other project-management: `form_assignment_role_list()`, `project_assignment_actors()`, `project_assignment_grant()`, `project_assignment_revoke()`, `project_create()`, `project_delete()`, `project_enable_encryption()`, `project_list()`, `project_replace()`, `project_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

pd <- project_detail()

pd %>%
  dplyr::select(-"verbs") %>%
  knitr::kable(.)

## End(Not run)
```

project_enable_encryption

Enable Project Managed Encryption.

Description

[Experimental]

Usage

```
project_enable_encryption(
  pid = get_default_pid(),
  passphrase,
  hint = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version()
)
```


Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
passphrase	(character) The encryption passphrase. If this passphrase is lost, the data will be irrecoverable.
hint	(character) A reminder about the passphrase. This is primarily useful when multiple encryption keys and passphrases are being used, to tell them apart. Default: NULL (no hint).
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

Details

Project Managed Encryption can be enabled via the API. To do this, POST with the passphrase and optionally a reminder hint about the passphrase. If managed encryption is already enabled, a 409 error response will be returned. Enabling managed encryption will modify all unencrypted Forms in the Project, and as a result the version of all Forms within the Project will also be modified. It is therefore best to enable managed encryption before devices are in the field. Any Forms in the Project that already have self-supplied encryption keys will be left alone. This endpoint requires ODK Central v0.6 or later.

Value

A list with a single element `success (TRUE)` as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-project-management/#enabling-project-managed-encryption>

Other project-management: [form_assignment_role_list\(\)](#), [project_assignment_actors\(\)](#), [project_assignment_grant\(\)](#), [project_assignment_revoke\(\)](#), [project_create\(\)](#), [project_delete\(\)](#), [project_detail\(\)](#), [project_list\(\)](#), [project_replace\(\)](#), [project_update\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

p <- project_create(name = "Encrypted Project")

project_enable_encryption(
  pid = p$id,
  passphrase = "super duper secret",
  hint = "it was a secret"
)
# > $success
# > [1] TRUE

## End(Not run)
```

project_list

List all projects.

Description

While the API endpoint will return all projects, [project_list](#) will fail with incorrect or missing authentication.

Usage

```
project_list(
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

url The ODK Central base URL without trailing slash.
 Default: [get_default_url](#).
 Set default url through `ru_setup(url="...")`.
 See `vignette("Setup", package = "ruODK")`.

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details**[Stable]****Value**

A tibble with one row per project and all project metadata as columns as per ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-project-management/#listing-projects>

Other project-management: `form_assignment_role_list()`, `project_assignment_actors()`, `project_assignment_grant()`, `project_assignment_revoke()`, `project_create()`, `project_delete()`, `project_detail()`, `project_enable_encryption()`, `project_replace()`, `project_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

p1 <- project_list()
knitr::kable(p1)

# project_list returns a tibble
class(p1)
# > "tbl_df" "tbl" "data.frame"

# columns are project metadata
names(p1)
# > "id" "name" "forms" "app_users" "created_at" "updated_at"
# > "last_submission" "archived"

## End(Not run)
```

project_replace	<i>Replace a Project.</i>
-----------------	---------------------------

Description

[Experimental]

Usage

```
project_replace(
  pid = get_default_pid(),
  name,
  description = "",
  archived = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
name	(character) The desired name of the Project.
description	(character) The desired description of the Project.
archived	(lgl) Archive the Project.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

This endpoint allows a deep update of Project metadata, Form metadata, and Form Assignment metadata at once and transactionally using a nested data format. This function replaces the top-level Project metadata. Each call sends the complete name, description and archived flag. Omitted Form detail is left as-is, but this function sends no Form detail. Form states and assignments need a direct call to the ODK Central API.

Value

A tibble with one row holding the replaced Project's metadata as columns, as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-project-management/#deep-updating-project-and-form-details>

Other project-management: `form_assignment_role_list()`, `project_assignment_actors()`, `project_assignment_grant()`, `project_assignment_revoke()`, `project_create()`, `project_delete()`, `project_detail()`, `project_enable_encryption()`, `project_list()`, `project_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

p <- project_create("Test Project")

p <- project_replace(
  pid = p$id,
  name = "Renamed Project",
  description = "A test project.",
  archived = FALSE
)

p$name
# > "Renamed Project"

## End(Not run)
```

project_update

Modify a Project.

Description

[Experimental]

Usage

```
project_update(
  pid = get_default_pid(),
  name = NULL,
  description = NULL,
  archived = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
name	(character) The desired name of the Project. Default: NULL (unchanged).
description	(character) The desired description of the Project. Default: NULL (unchanged).
archived	(lgl) Archive the Project. Default: NULL (unchanged).
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

The Project name may be updated, as well as the Project description and the archived flag. By default, archived is not set, which is equivalent to false. If archived is set to true, the Project will be sorted to the bottom of the list, and in the web management application the Project will become effectively read-only. API write access is not affected. Only the properties you supply are changed. Anything you do not supply remains untouched.

Value

A tibble with one row holding the updated Project's metadata as columns, as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-project-management/#updating-project-details>

Other project-management: [form_assignment_role_list\(\)](#), [project_assignment_actors\(\)](#), [project_assignment_grant\(\)](#), [project_assignment_revoke\(\)](#), [project_create\(\)](#), [project_delete\(\)](#), [project_detail\(\)](#), [project_enable_encryption\(\)](#), [project_list\(\)](#), [project_replace\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

p <- project_create("Test Project")

p <- project_update(pid = p$id, description = "A test project.")

p$description
# > "A test project."

## End(Not run)
```

public_link_create	Create a Public Access Link for one Form.
--------------------	---

Description

[Experimental]

Usage

```
public_link_create(
  pid = get_default_pid(),
  fid = get_default_fid(),
  display_name,
  once = FALSE,
  properties = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
-----	--

fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: <code>get_default_fid</code> . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
display_name	(character) The name of the Link, for keeping track of. This name shows on the Central administration website but not to survey respondents.
once	(lgl) Create an Enketo single-submission survey instead of a standard one. Default: FALSE.
properties	(list) An optional named list mapping pre-registered Actor Property names to string values. Default: NULL (not sent).
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

To create a new Public Access Link to a Form, send at least a `display_name` for the resulting Actor. Set `once` to TRUE for a link that each respondent can only fill once. This setting is enforced by Enketo using local device tracking. An optional `properties` object sets pre-registered Actor Property values on the Link at creation time.

Value

A tibble with one row holding the new Link's metadata as columns as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#creating-a-link>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`,


```
form_version_attachment_list(), form_version_detail(), form_version_list(), form_version_xlsx(),
form_version_xml(), form_xlsx(), form_xml(), public_link_delete(), public_link_detail(),
public_link_list(), public_link_update()
```

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

pl <- public_link_create(fid = fl$fid[[1]], display_name = "Survey link")

pl$token

## End(Not run)
```

public_link_delete	<i>Delete a Public Access Link.</i>
--------------------	-------------------------------------

Description

[Experimental]

Usage

```
public_link_delete(
  pid = get_default_pid(),
  fid = get_default_fid(),
  link_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

link_id	The numeric ID of the Link.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

A DELETE to a Link resource removes the Link from the system entirely. To only revoke the Link's access and prevent future Submissions without removing its record, terminate its session token instead.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#deleting-a-link>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_detail()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

pl <- public_link_create(fid = fid, display_name = "Temporary link")
```

```

public_link_delete(fid = fid, link_id = pl$id)
# > $success
# > [1] TRUE

## End(Not run)

```

public_link_detail *Show details of one Public Access Link.*

Description

[Experimental]

Usage

```

public_link_detail(
  pid = get_default_pid(),
  fid = get_default_fid(),
  link_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
link_id	The numeric ID of the Link.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Returns the details of a single Public Access Link by its numeric ID.

Value

A tibble with one row holding the Link's metadata as columns as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-form-management/#getting-link-details>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_list()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

pl <- public_link_list(fid = fid)

pd <- public_link_detail(fid = fid, link_id = pl$fid[[1]])

pd |> knitr::kable()

## End(Not run)
```

public_link_list	List all Public Access Links of one Form.
------------------	---

Description

[Experimental]

Usage

```
public_link_list(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.

tz A timezone to convert dates and times to.
 Read vignette("setup", package = "ruODK") to learn how ruODK's timezone can be set globally or per function.

Details

This lists every Public Access Link upon the given Form. Anybody in possession of a Public Access Link for a Form can use that link to submit data to that Form.

Value

A tibble with one row per Public Access Link and all Link metadata as columns as per the ODK Central API docs. Column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-form-management/#listing-all-links>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

pl <- public_link_list(fid = fl$fid[[1]])

pl |> knitr::kable()

## End(Not run)
```

public_link_update	<i>Set Actor Property values on a Public Access Link.</i>
--------------------	---

Description

[Experimental]

Usage

```
public_link_update(
  pid = get_default_pid(),
  fid = get_default_fid(),
  link_id,
  properties = list(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
link_id	The numeric ID of the Link.
properties	(list) A named list mapping Actor Property names to string values.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.

Details

Sets pre-registered Actor Property values on a Public Access Link. The body holds a `properties` object mapping property names to string values. Unsetting a property needs a JSON null, which this function cannot send. Only set properties with this function. This endpoint requires the `public_link.update` permission.

Value

A list with the updated Link's metadata as per the ODK Central API docs, including a properties object with all currently set property values.

See Also

<https://docs.getodk.org/central-api-form-management/#setting-actor-property-values-on-a-link>

Other form-management: `form_assignment_actors()`, `form_assignment_grant()`, `form_assignment_revoke()`, `form_attachment_download()`, `form_attachment_list()`, `form_create()`, `form_dataset_diff()`, `form_delete()`, `form_detail()`, `form_draft_attachment_delete()`, `form_draft_attachment_download()`, `form_draft_attachment_link()`, `form_draft_attachment_list()`, `form_draft_attachment_upload()`, `form_draft_create()`, `form_draft_dataset_diff()`, `form_draft_delete()`, `form_draft_detail()`, `form_draft_publish()`, `form_draft_xlsx()`, `form_draft_xml()`, `form_link()`, `form_list()`, `form_restore()`, `form_schema()`, `form_schema_ext()`, `form_update()`, `form_version_attachment_download()`, `form_version_attachment_list()`, `form_version_detail()`, `form_version_list()`, `form_version_xlsx()`, `form_version_xml()`, `form_xlsx()`, `form_xml()`, `public_link_create()`, `public_link_delete()`, `public_link_detail()`, `public_link_list()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "....svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

pl <- public_link_list(fid = fid)

public_link_update(
  fid = fid,
  link_id = pl$id[[1]],
  properties = list("region" = "North")
)

## End(Not run)
```

role_detail

Show details of one Role.

Description

[Experimental]

Usage

```
role_detail(
  role_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

role_id	The numeric ID of the Role, or its system name.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

Getting an individual Role reveals no additional information over listing all Roles, but allows direct lookup of a specific Role. The ID accepts the numeric ID of the Role, or a system name if the Role has one (e.g. admin for the Administrator Role). There are no authorization restrictions upon this endpoint.

Value

A tibble with one row holding the Role's metadata as columns as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#getting-role-details>

Other user-management: [app_user_create\(\)](#), [app_user_delete\(\)](#), [app_user_list\(\)](#), [assignment_actors\(\)](#), [assignment_grant\(\)](#), [assignment_list\(\)](#), [assignment_revoke\(\)](#), [form_assignment_list\(\)](#), [project_assignment_list\(\)](#), [role_list\(\)](#), [user_create\(\)](#), [user_delete\(\)](#), [user_detail\(\)](#), [user_list\(\)](#), [user_preference_project_delete\(\)](#), [user_preference_project_set\(\)](#), [user_preference_site_delete\(\)](#), [user_preference_site_set\(\)](#), [user_reset_password\(\)](#), [user_update\(\)](#), [user_update_password\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

r <- role_detail("admin")

r$verbs

## End(Not run)
```

role_list	<i>List all Roles.</i>
-----------	------------------------

Description

[Experimental]

Usage

```
role_list(
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .

tz A timezone to convert dates and times to.
 Read `vignette("setup", package = "ruODK")` to learn how ruODK's timezone can be set globally or per function.

Details

The Roles API lists and describes each known Role within the system. Each Role holds the verbs it allows its assignees to perform. There are no authorization restrictions upon this endpoint: anybody is allowed to list all Role information at any time.

Value

A tibble with one row per Role and all Role metadata as columns as per the ODK Central API docs. The verbs column is a list of character vectors. Other column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#listing-all-roles>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

r1 <- role_list()

r1 |> dplyr::select(id, name, system) |> knitr::kable()

## End(Not run)
```

ru_msg_abort	<i>Abort with an error message.</i>
--------------	-------------------------------------

Description

[Stable]

Usage

```
ru_msg_abort(message)
```

Arguments

message (chr) A message to print

Details

Errors are never silenced, neither by verbose nor by options(ruODK.quiet = TRUE).

See Also

Other messaging: [ru_msg_info\(\)](#), [ru_msg_noop\(\)](#), [ru_msg_success\(\)](#), [ru_msg_warn\(\)](#)

Examples

```
## Not run:
ru_msg_abort("This is an error, abort.")

## End(Not run)
```

ru_msg_info	<i>Print an info message.</i>
-------------	-------------------------------

Description

[Stable]

Usage

```
ru_msg_info(message, verbose = get_ru_verbose())
```

Arguments

message (chr) A message to print

verbose Whether to display debug messages or not.
Read vignette("setup", package = "ruODK") to learn how ruODK's verbosity can be set globally or per function.

Details

Set options(ruODK.quiet = TRUE) to silence this message regardless of verbose.

See Also

Other messaging: [ru_msg_abort\(\)](#), [ru_msg_noop\(\)](#), [ru_msg_success\(\)](#), [ru_msg_warn\(\)](#)

Examples

```
ru_msg_info("This is an info message.")
```

ru_msg_noop	<i>Print a noop message.</i>
-------------	------------------------------

Description**[Stable]****Usage**

```
ru_msg_noop(message, verbose = get_ru_verbose())
```

Arguments

message	(chr) A message to print
verbose	Whether to display debug messages or not. Read vignette("setup", package = "ruODK") to learn how ruODK's verbosity can be set globally or per function.

Details

Set options(ruODK.quiet = TRUE) to silence this message regardless of verbose.

See Also

Other messaging: [ru_msg_abort\(\)](#), [ru_msg_info\(\)](#), [ru_msg_success\(\)](#), [ru_msg_warn\(\)](#)

Examples

```
ru_msg_noop("This is a noop message.")
```

ru_msg_success	<i>Print a success message.</i>
----------------	---------------------------------

Description**[Stable]****Usage**

```
ru_msg_success(message, verbose = get_ru_verbose())
```

Arguments

message	(chr) A message to print
verbose	Whether to display debug messages or not. Read vignette("setup", package = "ruODK") to learn how ruODK's verbosity can be set globally or per function.

Details

Set `options(ruODK.quiet = TRUE)` to silence this message regardless of verbose.

See Also

Other messaging: [ru_msg_abort\(\)](#), [ru_msg_info\(\)](#), [ru_msg_noop\(\)](#), [ru_msg_warn\(\)](#)

Examples

```
ru_msg_success("This is a success message.")
```

ru_msg_warn	<i>Signal a warning message.</i>
-------------	----------------------------------

Description

[Stable]

Usage

```
ru_msg_warn(message, verbose = get_ru_verbose())
```

Arguments

message	(chr) A message to print
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

Warnings are never silenced by `options(ruODK.quiet = TRUE)`.

See Also

Other messaging: [ru_msg_abort\(\)](#), [ru_msg_info\(\)](#), [ru_msg_noop\(\)](#), [ru_msg_success\(\)](#)

Examples

```
## Not run:
ru_msg_warn("This is a warning.")

## End(Not run)
```

ru_settings	<i>Get or set ruODK settings.</i>
-------------	-----------------------------------

Description**[Stable]****Usage**

```
ru_settings()
get_default_pid()
get_default_fid()
get_default_url()
get_default_un()
get_default_pw()
get_default_pp()
get_default_tz()
get_default_orders()
get_test_url()
get_test_un()
get_test_pw()
get_test_pid()
get_test_fid()
get_test_fid_zip()
get_test_fid_att()
get_test_fid_gap()
get_test_fid_wkt()
get_test_pp()
```

```

get_ru_verbose()

get_default_odkc_version()

get_test_odkc_version()

get_retries()

```

Value

[ru_settings](#) prints your default ODK Central project ID, form ID, url, username, and password, corresponding optional test server as well as verbosity and HTTP request settings. [ru_setup](#) sets your production and test settings, while `get_(default/test)_*` get each of those respective settings.

See Also

[ru_setup](#), [get_default_pid](#), [get_default_fid](#), [get_default_url](#), [get_default_un](#), [get_default_pw](#), [get_default_pp](#), [get_default_tz](#), [get_default_odkc_version](#), [get_retries](#), [get_test_pid](#), [get_test_fid](#), [get_test_fid_zip](#), [get_test_fid_att](#), [get_test_fid_gap](#), [get_test_fid_wkt](#), [get_test_url](#), [get_test_un](#), [get_test_pw](#), [get_test_pp](#), [get_test_odkc_version](#), [get_ru_verbose](#).

Other `ru_settings`: [odata_svc_parse\(\)](#), [parse_odkc_version\(\)](#), [ru_setup\(\)](#), [semver_gt\(\)](#), [semver_lt\(\)](#), [yell_if_error\(\)](#), [yell_if_missing\(\)](#)

Examples

```
ru_settings()
```

ru_setup	<i>Configure default ruODK settings.</i>
----------	--

Description

Settings are returned invisibly and additionally printed depending on [get_ru_verbose](#).

Usage

```

ru_setup(
  svc = NULL,
  pid = NULL,
  fid = NULL,
  url = NULL,
  un = NULL,
  pw = NULL,
  pp = NULL,
  tz = NULL,
  odkc_version = NULL,

```



```

    retries = NULL,
    verbose = NULL,
    test_svc = NULL,
    test_pid = NULL,
    test_fid = NULL,
    test_fid_zip = NULL,
    test_fid_att = NULL,
    test_fid_gap = NULL,
    test_fid_wkt = NULL,
    test_url = NULL,
    test_un = NULL,
    test_pw = NULL,
    test_pp = NULL,
    test_odkc_version = NULL
)

```

Arguments

svc	(optional, character) The OData service URL of a form. This parameter will set pid, fid, and url. It is sufficient to supply svc, un, and pw.
pid	(optional, character) The ID of an existing project on url. This will override the project ID from svc. A numeric value for pid will be converted to character.
fid	(optional, character) The alphanumeric ID of an existing form in pid. This will override the form ID from svc.
url	An ODK Central URL, e.g. "https://sandbox.central.getodk.org". This will override the ODK Central base URL from svc.
un	An ODK Central username which is the email of a "web user" in the specified ODK Central instance url (optional, character).
pw	The password for user un (optional, character).
pp	The passphrase (optional, character) for an encrypted form.
tz	Global default time zone. ruODK's time zone is determined in order of precedence: • Function parameter: e.g. <code>odata_submission_get(tz = "Australia/Perth")</code> • ruODK setting: <code>ru_setup(tz = "Australia/Perth")</code> • Environment variable RU_TIMEZONE (e.g. set in .Renvirom) • UTC (GMT+00)
odkc_version	The ODK Central version as major/minor version, e.g. 1.1.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
verbose	Global default for ruODK verbosity. ruODK verbosity is determined in order of precedence: • Function parameter: e.g. <code>odata_submission_get(verbose = TRUE)</code> • ruODK setting: <code>ru_setup(verbose = TRUE)</code>

	• Environment variable RU_VERBOSE (e.g. set in .Renvirom) • FALSE.
test_svc	(optional, character) The OData service URL of a test form. This parameter will set test_pid, test_fid, and test_url. It is sufficient to supply test_svc, test_un, and test_pw to configure testing.
test_pid	(optional, character) The numeric ID of an existing project on test_url. This will override the project ID from test_svc. A numeric value for test_pid will be converted to character.
test_fid	(optional, character) The alphanumeric ID of an existing form in test_pid. This will override the form ID from test_svc. This form is used as default form in all tests, examples, vignettes, data, and Rmd templates.
test_fid_zip	(optional, character) The alphanumeric ID of an existing form in test_pid. This will override the form ID from test_svc. Provide the form ID of a form with few submissions and without attachments. This form is used to test the repeated download of all form submissions.
test_fid_att	(optional, character) The alphanumeric ID of an existing form in test_pid. This will override the form ID from test_svc. Provide the form ID of a form with few submissions and few attachments. This form is used to test downloading and linking attachments.
test_fid_gap	(optional, character) The alphanumeric ID of an existing form in test_pid. This will override the form ID from test_svc. Provide the form ID of a form with gaps in the first submission. This form is used to test parsing incomplete submissions.
test_fid_wkt	(optional, character) The alphanumeric ID of an existing form in test_pid. This will override the form ID from test_svc. Provide the form ID of a form with geopoints, geotraces, and geoshapes.
test_url	(optional, character) A valid ODK Central URL for testing. This will override the ODK Central base URL from svc.
test_un	(optional, character) A valid ODK Central username (email) privileged to view the test project(s) at test_url.
test_pw	(optional, character) The valid ODK Central password for test_un.
test_pp	(optional, character) The valid passphrase to decrypt the data of encrypted project test_pid for download, only used for tests.
test_odkc_version	The ODK Central test server's version as major/minor version, e.g. 1.1.

Details

[Stable]

[ru_setup](#) sets ODK Central connection details. [ruODK](#)'s functions default to use the default project ID, form ID, URL, username, and password unless specified explicitly.

Any parameters not specified will remain unchanged. It is therefore possible to set up username and password initially with `ru_setup(un="XXX", pw="XXX")`, and switch between forms with

`ru_setup(svc="XXX")`, supplying the form's OData service URL. ODK Central conveniently provides the OData service URL in the form submission tab, which in turn contains base URL, project ID, and form ID.

`ruODK`'s automated tests require a valid ODK Central URL, and a privileged username and password of a "web user" on that ODK Central instance, as well as an existing project and form.

See Also

Other `ru_settings`: `odata_svc_parse()`, `parse_odkc_version()`, `ru_settings()`, `semver_gt()`, `semver_lt()`, `yell_if_error()`, `yell_if_missing()`

Examples

```
# `ruODK` users only need default settings to their ODK Central:
ru_setup(url = "https://my-odkc.com", un = "me@email.com", pw = "...")

# `ruODK` contributors and maintainers need specific ODK Central
# instances to run tests and build vignettes, see contributing guide:
ru_setup(
  url = "https://odkcentral.dbca.wa.gov.au",
  un = "me@email.com",
  pw = "...",
  pp = "...",
  test_url = "https://sandbox.central.getodk.org",
  test_un = "me@email.com",
  test_pw = "...",
  test_pp = "...",
  test_pid = 14,
  test_fid = "build_Flora-Quadrat-0-2_1558575936",
  test_fid_zip = "build_Spotlighting-0-6_1558333698",
  test_fid_att = "build_Flora-Quadrat-0-1_1558330379",
  test_fid_gap = "build_Turtle-Track-or-Nest-1-0_1569907666",
  test_fid_wkt = "build_Locations_1589344221",
  retries = 3,
  verbose = TRUE
)
```

`semver_gt`

Show whether a given semver is greater than a baseline version.

Description

Show whether a given semver is greater than a baseline version.

Usage

```
semver_gt(sv = get_default_odkc_version(), to = "1.5.0")
```

Arguments

sv	The semver to compare as character ("2023.5.1", "1.5.0", "1.5"), or numeric (1.5). The value is always parsed with <code>semver::parse_semver()</code> . Default: <code>get_default_odkc_version()</code> .
to	The semver to compare to as string. Although semver can parse complete version strings, to is still parsed by <code>parse_odkc_version()</code> to ensure it is complete with major, minor, and patch version components.

Value

A boolean indicating whether the given semver sv is greater than the baseline semver to.

See Also

Other ru_settings: [odata_svc_parse\(\)](#), [parse_odkc_version\(\)](#), [ru_settings\(\)](#), [ru_setup\(\)](#), [semver_lt\(\)](#), [yell_if_error\(\)](#), [yell_if_missing\(\)](#)

Examples

```
get_default_odkc_version() |> semver_gt("0.8.0")
"2024.1.1" |> semver_gt("2024.1.0")
"2024.1.1" |> semver_gt("2024.1.1")
"2024.1.1" |> semver_gt("2024.1.2")
```

semver_lt	<i>Show whether a given semver is lesser than a baseline version.</i>
-----------	---

Description

Show whether a given semver is lesser than a baseline version.

Usage

```
semver_lt(sv = get_default_odkc_version(), to = "1.5.0")
```

Arguments

sv	The semver to compare as character ("2023.5.1", "1.5.0", "1.5"), or numeric (1.5). The value is always parsed with <code>semver::parse_semver()</code> . Default: <code>get_default_odkc_version()</code> .
to	The semver to compare to as string. Although semver can parse complete version strings, to is still parsed by <code>parse_odkc_version()</code> to ensure it is complete with major, minor, and patch version components.

Value

A boolean indicating whether the given semver sv is greater than the baseline semver to.

See Also

Other `ru_settings`: [odata_svc_parse\(\)](#), [parse_odkc_version\(\)](#), [ru_settings\(\)](#), [ru_setup\(\)](#), [semver_gt\(\)](#), [yell_if_error\(\)](#), [yell_if_missing\(\)](#)

Examples

```
get_default_odkc_version() |> semver_lt("0.8.0")
"2024.1.1" |> semver_lt("2024.1.0")
"2024.1.1" |> semver_lt("2024.1.1")
"2024.1.1" |> semver_lt("2024.1.2")
```

<code>split_geopoint</code>	<i>Annotate a dataframe containing a geopoint column with lon, lat, alt.</i>
-----------------------------	--

Description

[Stable]

Usage

```
split_geopoint(data, colname, wkt = FALSE)
```

Arguments

<code>data</code>	(dataframe) A dataframe with a geopoint column.
<code>colname</code>	(chr) The name of the geopoint column. This column will be retained.
<code>wkt</code>	Whether geofields are GeoJSON (if FALSE) or WKT strings (if TRUE), default: FALSE.

Details

This function is used by [handle_ru_geopoints](#) on all geopoint fields as per [form_schema](#).

Value

The given dataframe with the WKT POINT column `colname`, plus three new columns, `colname_longitude`, `colname_latitude`, `colname_altitude`. The three new columns are prefixed with the original `colname` to avoid naming conflicts with any other geopoint columns.

See Also

Other utilities: [attachment_get\(\)](#), [attachment_link\(\)](#), [attachment_url\(\)](#), [drop_null_coords\(\)](#), [form_schema_parse\(\)](#), [get_one_attachment\(\)](#), [get_one_submission\(\)](#), [get_one_submission_att_list\(\)](#), [get_one_submission_audit\(\)](#), [handle_ru_attachments\(\)](#), [handle_ru_datetimes\(\)](#), [handle_ru_geopoints\(\)](#), [handle_ru_geoshapes\(\)](#), [handle_ru_geotraces\(\)](#), [isodt_to_local\(\)](#), [odata_submission_rectangle\(\)](#), [predict_ruodk_name\(\)](#), [prepend_uuid\(\)](#), [ru_http_request\(\)](#), [split_geoshape\(\)](#), [split_geotrace\(\)](#), [strip_uuid\(\)](#), [tidyeval](#), [unnest_all\(\)](#)

Examples

```
## Not run:
df_wkt <- tibble::tibble(
  stuff = c("asd", "sdf", "sdf"),
  loc = c(
    "POINT (115.99 -32.12 20.01)",
    "POINT (116.12 -33.34 15.23)",
    "POINT (114.01 -31.56 23.56)"
  )
)
df_wkt_split <- df |> split_geopoint("loc", wkt = TRUE)
testthat::expect_equal(
  names(df_wkt_split),
  c("stuff", "loc", "loc_longitude", "loc_latitude", "loc_altitude")
)

# With package data
data("geo_fs")
data("geo_wkt_raw")
data("geo_gj_raw")

# Find variable names of geopoints
geo_fields <- geo_fs |>
  dplyr::filter(type == "geopoint") |>
  magrittr::extract2("ruodk_name")
geo_fields[1] # First geotrace in data: point_location_point_gps

# Rectangle but don't parse submission data (GeoJSON and WKT)
geo_gj_rt <- geo_gj_raw |>
  odata_submission_rectangle(form_schema = geo_fs)
geo_wkt_rt <- geo_wkt_raw |>
  odata_submission_rectangle(form_schema = geo_fs)

# Data with first geopoint split
gj_first_gt <- split_geopoint(geo_gj_rt, geo_fields[1], wkt = FALSE)
gj_first_gt$point_location_point_gps_longitude

wkt_first_gt <- split_geopoint(geo_wkt_rt, geo_fields[1], wkt = TRUE)
wkt_first_gt$point_location_point_gps_longitude

## End(Not run)
```

split_geoshape

Annotate a dataframe containing a geoshape column with lon, lat, alt of the geotrace's first point.

Description
[Stable]

Usage

```
split_geoshape(
  data,
  colname,
  wkt = FALSE,
  odkc_version = get_default_odkc_version()
)
```

Arguments

data	(dataframe) A dataframe with a geoshape column.
colname	(chr) The name of the geoshape column. This column will be retained.
wkt	Whether geofields are GeoJSON (if FALSE) or WKT strings (if TRUE), default: FALSE.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

Details

This function is used by `handle_ru_geopoints` on all geopoint fields as per `form_schema`.

Value

The given dataframe with the geoshape column `colname`, plus three new columns, `colname_longitude`, `colname_latitude`, `colname_altitude`. The three new columns are prefixed with the original `colname` to avoid naming conflicts with any other geoshape columns.

See Also

Other utilities: `attachment_get()`, `attachment_link()`, `attachment_url()`, `drop_null_coords()`, `form_schema_parse()`, `get_one_attachment()`, `get_one_submission()`, `get_one_submission_att_list()`, `get_one_submission_audit()`, `handle_ru_attachments()`, `handle_ru_datetimes()`, `handle_ru_geopoints()`, `handle_ru_geoshapes()`, `handle_ru_geotraces()`, `isodt_to_local()`, `odata_submission_rectangle()`, `predict_ruodk_name()`, `prepend_uuid()`, `ru_http_request()`, `split_geopoint()`, `split_geotrace()`, `strip_uuid()`, `tidyeval`, `unnest_all()`

Examples

```
## Not run:
library(magrittr)
data("geo_fs")
data("geo_wkt_raw")
data("geo_gj_raw")
```

```

# Find variable names of geoshapes
geo_fields <- geo_fs %>%
  dplyr::filter(type == "geoshape") %>%
  magrittr::extract2("ruodk_name")
geo_fields[1] # First geoshape in data: shape_location_shape_gps

# Rectangle but don't parse submission data (GeoJSON and WKT)
geo_gj_rt <- geo_gj_raw %>%
  odata_submission_rectangle(form_schema = geo_fs)
geo_wkt_rt <- geo_wkt_raw %>%
  odata_submission_rectangle(form_schema = geo_fs)

# Data with first geoshape split
gj_first_gt <- split_geoshape(geo_gj_rt, geo_fields[1], wkt = FALSE)
cn_gj <- names(gj_first_gt)
testthat::expect_true("shape_location_shape_gps_longitude" %in% cn_gj)
testthat::expect_true("shape_location_shape_gps_latitude" %in% cn_gj)
testthat::expect_true("shape_location_shape_gps_altitude" %in% cn_gj)
testthat::expect_true(
  is.numeric(gj_first_gt$shape_location_shape_gps_longitude)
)
testthat::expect_true(
  is.numeric(gj_first_gt$shape_location_shape_gps_latitude)
)
testthat::expect_true(
  is.numeric(gj_first_gt$shape_location_shape_gps_altitude)
)

wkt_first_gt <- split_geoshape(geo_wkt_rt, geo_fields[1], wkt = TRUE)
cn_wkt <- names(wkt_first_gt)
testthat::expect_true("shape_location_shape_gps_longitude" %in% cn_wkt)
testthat::expect_true("shape_location_shape_gps_latitude" %in% cn_wkt)
testthat::expect_true("shape_location_shape_gps_altitude" %in% cn_wkt)
testthat::expect_true(
  is.numeric(wkt_first_gt$shape_location_shape_gps_longitude)
)
testthat::expect_true(
  is.numeric(wkt_first_gt$shape_location_shape_gps_latitude)
)
testthat::expect_true(
  is.numeric(wkt_first_gt$shape_location_shape_gps_altitude)
)

## End(Not run)

```

split_geotrace

Annotate a dataframe containing a geotrace column with lon, lat, alt of the geotrace's first point.

Description**[Stable]****Usage**

```
split_geotrace(
  data,
  colname,
  wkt = FALSE,
  odkc_version = get_default_odkc_version()
)
```

Arguments

<code>data</code>	(dataframe) A dataframe with a geotrace column.
<code>colname</code>	(chr) The name of the geotrace column. This column will be retained.
<code>wkt</code>	Whether geofields are GeoJSON (if FALSE) or WKT strings (if TRUE), default: FALSE.
<code>odkc_version</code>	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code> . Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: <code>get_default_odkc_version</code> or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

Details

This function is used by `handle_ru_geopoints` on all geopoint fields as per `form_schema`.

The format of the geotrace (GeoJSON, WKT, ODK Linestring) is determined via parameters `wkt` and `odkc_version`, rather than inferred from the class of the column. ODK Linestrings are character vectors without a leading "LINESTRING (", WKT are character vectors with a leading "LINESTRING (", and GeoJSON are list columns.

Value

The given dataframe with the geotrace column `colname`, plus three new columns, `colname_longitude`, `colname_latitude`, `colname_altitude`. The three new columns are prefixed with the original `colname` to avoid naming conflicts with any other geotrace columns.

See Also

Other utilities: `attachment_get()`, `attachment_link()`, `attachment_url()`, `drop_null_coords()`, `form_schema_parse()`, `get_one_attachment()`, `get_one_submission()`, `get_one_submission_att_list()`, `get_one_submission_audit()`, `handle_ru_attachments()`, `handle_ru_datetimes()`, `handle_ru_geopoints()`, `handle_ru_geoshapes()`, `handle_ru_geotraces()`, `isodt_to_local()`, `odata_submission_rectangle()`, `predict_ruodk_name()`, `prepend_uuid()`, `ru_http_request()`, `split_geopoint()`, `split_geoshape()`, `strip_uuid()`, `tidyeval`, `unnest_all()`

Examples

```
## Not run:
library(magrittr)
data("geo_fs")
data("geo_wkt_raw")
data("geo_gj_raw")

# Find variable names of geotrace
geo_fields <- geo_fs %>%
  dplyr::filter(type == "geotrace") %>%
  magrittr::extract2("ruodk_name")
geo_fields[1] # First geotrace in data: path_location_path_gps

# Rectangle but don't parse submission data (GeoJSON and WKT)
geo_gj_rt <- geo_gj_raw %>%
  odata_submission_rectangle(form_schema = geo_fs)
geo_wkt_rt <- geo_wkt_raw %>%
  odata_submission_rectangle(form_schema = geo_fs)

# Data with first geotrace split
gj_first_gt <- split_geotrace(geo_gj_rt, geo_fields[1], wkt = FALSE)
testthat::expect_true(
  "path_location_path_gps_longitude" %in% names(gj_first_gt)
)
testthat::expect_true(
  "path_location_path_gps_latitude" %in% names(gj_first_gt)
)
testthat::expect_true(
  "path_location_path_gps_altitude" %in% names(gj_first_gt)
)
testthat::expect_true(
  is.numeric(gj_first_gt$path_location_path_gps_longitude)
)
testthat::expect_true(
  is.numeric(gj_first_gt$path_location_path_gps_latitude)
)
testthat::expect_true(
  is.numeric(gj_first_gt$path_location_path_gps_altitude)
)

wkt_first_gt <- split_geotrace(geo_wkt_rt, geo_fields[1], wkt = TRUE)
testthat::expect_true(
  "path_location_path_gps_longitude" %in% names(wkt_first_gt)
)
testthat::expect_true(
  "path_location_path_gps_latitude" %in% names(wkt_first_gt)
)
testthat::expect_true(
  "path_location_path_gps_altitude" %in% names(wkt_first_gt)
)
testthat::expect_true(
  is.numeric(wkt_first_gt$path_location_path_gps_longitude)
```

```

)
testthat::expect_true(
  is.numeric(wkt_first_gt$path_location_path_gps_latitude)
)
testthat::expect_true(
  is.numeric(wkt_first_gt$path_location_path_gps_altitude)
)

## End(Not run)

```

submission_audit_get *Get submission audits for a list of submission instance IDs.*

Description

[Experimental]

Usage

```

submission_audit_get(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

iid	A list of submission instance IDs, e.g. from <code>submission_list\$instance_id</code> .
pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code> . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: <code>get_default_fid</code> . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Uses `get_one_submission_audit` on a list of submission instance IDs (`iid`) as returned from `submission_list$instance_id`. By giving the list of `iid` to download explicitly, that list can be modified using information not accessible to ruODK, e.g. `iid` can be restricted to "only not already downloaded submissions".

To get the combined submission audit logs for one form as one single, concatenated `audit.csv` file, use `submission_export`.

Value

A nested list of submission audit logs.

See Also

<https://docs.getodk.org/central-api-submission-management/#retrieving-submission-xml>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attach`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_de`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# Step 1: Setup ruODK with OData Service URL (has url, pid, fid)
ruODK::ru_setup(svc = "...")

# Step 2: List all submissions of form
sl <- submission_list()

# Step 3: Get submission audit logs
sa <- submission_audit_get(sl$instance_id)

## End(Not run)
```

submission_changes	<i>Show changes between versions of one Submission.</i>
--------------------	---

Description

[Experimental]

Usage

```
submission_changes(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

This returns the changes, or edits, between different versions of a Submission. These changes are returned in an object that is indexed by the instanceID that uniquely identifies each version. Between two Submissions, there is an array of objects representing how each field changed. Each change object contains the old and new values, as well as the path of that changed node in the Submission XML. These changes reflect the updated instanceID and deprecatedID fields as well as the edited value.

Value

A list indexed by version instanceID. Each entry holds the field changes with old and new values and the path of the changed node. Names are kept exactly as returned by Central so versions stay addressable by instanceID.

See Also

<https://docs.getodk.org/central-api-submission-management/#getting-changes-between-versions>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_attachment_upload()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

d <- submission_changes(sl$instance_id[[1]])

d |> listviewer::jsonedit()

## End(Not run)
```

submission_comment_create

Post a comment to one Submission.

Description**[Experimental]****Usage**

```

submission_comment_create(
    iid,
    body,
    pid = get_default_pid(),
    fid = get_default_fid(),
    url = get_default_url(),
    un = get_default_un(),
    pw = get_default_pw(),
    retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
body	(character) The text of the comment.
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

The only accepted data is body, which contains the body of the comment to be made. It is not possible to edit or delete a comment once it has been made. This endpoint requires ODK Central v1.2 or later.

Value

A list with the new comment's body and the commenting Actor's actor_id.

See Also

<https://docs.getodk.org/central-api-submission-management/#posting-comments>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_list()`, `submission_draft_attachment_download()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_delete()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

c <- submission_comment_create(sl$instance_id[[1]], "Looks good.")

c$body
# > "Looks good."

## End(Not run)
```

submission_comment_list

List all comments of one Submission.

Description

[Experimental]

Usage

```
submission_comment_list(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
```



```

un = get_default_un(),
pw = get_default_pw(),
retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Comments have only a body comment text and an Actor that made the comment. It is not possible to get a specific comment's details, or to edit or delete a comment once it has been made. This endpoint requires ODK Central v1.2 or later.

Value

A tibble with one row per comment and the columns body (the comment text) and actor_id (the ID of the commenting Actor).

See Also

<https://docs.getodk.org/central-api-submission-management/#listing-comments>

Other submission-management: [attachment_delete\(\)](#), [attachment_list\(\)](#), [attachment_upload\(\)](#), [encryption_key_list\(\)](#), [submission_audit_get\(\)](#), [submission_changes\(\)](#), [submission_comment_create\(\)](#),

```

submission_create(), submission_csv(), submission_delete(), submission_detail(), submission_draft_attach
submission_draft_attachment_download(), submission_draft_attachment_list(), submission_draft_attachment
submission_draft_create(), submission_draft_export(), submission_draft_get(), submission_draft_keys(),
submission_draft_list(), submission_edit(), submission_export(), submission_geodata(),
submission_geojson(), submission_get(), submission_list(), submission_restore(), submission_review(),
submission_submitters(), submission_update(), submission_version_attachment_delete(),
submission_version_attachment_download(), submission_version_attachment_list(), submission_version_de
submission_version_geojson(), submission_version_xml(), submission_versions()

```

Examples

```

## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

cl <- submission_comment_list(sl$instance_id[[1]])

cl |> knitr::kable()

## End(Not run)

```

submission_create	<i>Create a Submission.</i>
-------------------	-----------------------------

Description

[Experimental]

Usage

```

submission_create(
  pid = get_default_pid(),
  fid = get_default_fid(),
  xml = NULL,
  device_id = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code> . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
-----	---

fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: <code>get_default_fid</code> . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
xml	(character) The Submission XML as a single string.
device_id	(character) Optionally record a particular deviceID associated with this Submission. It is recorded along with the data, but Central does nothing more with it. Default: NULL (not sent).
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

To create a Submission by REST rather than over the OpenRosa interface, POST the Submission XML to this endpoint. Unlike the OpenRosa Form Submission API, this interface does not accept Submission attachments upon Submission creation. Instead, the server determines which attachments are expected based on the Submission XML, and you can use `attachment_upload()` to add the attachments afterwards.

If the XML is unparseable or there is some other input problem with your data, Central answers 400. If a Submission already exists with the given instanceID, Central answers 409. The `<meta>` block holds a unique instanceID for each Submission.

Value

A list with the new Submission's metadata as per the ODK Central API docs, identical in shape to `submission_detail()`. Top level list elements are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-submission-management/#creating-a-submission>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`,

```

submission_draft_get(), submission_draft_keys(), submission_draft_list(), submission_edit(),
submission_export(), submission_geodata(), submission_geojson(), submission_get(),
submission_list(), submission_restore(), submission_review(), submission_submitters(),
submission_update(), submission_version_attachment_delete(), submission_version_attachment_download(),
submission_version_attachment_list(), submission_version_detail(), submission_version_geojson(),
submission_version_xml(), submission_versions()

```

Examples

```

## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

xml <- paste0(
  '<data id="simple" version="1">',
  "<meta><instanceID>uuid:85cb9aff-005e-4edd-9739-dc9c1a829c44</instanceID></meta>", # nolint
  "<name>Jo</name>",
  "</data>"
)

s <- submission_create(fid = "simple", xml = xml)

s$instance_id
# > "uuid:85cb9aff-005e-4edd-9739-dc9c1a829c44"

## End(Not run)

```

submission_csv

Export the root Submission table to CSV.

Description

[Experimental]

Usage

```

submission_csv(
  pid = get_default_pid(),
  fid = get_default_fid(),
  filter = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: <code>get_default_pid</code> . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: <code>get_default_fid</code> . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
filter	(character) An OData-style \$filter query to filter the Submissions by. Only certain fields are available to reference. Default: NULL (not sent).
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

The ZIP export endpoints provide all possible related repeat table files, as well as the media files associated with the Submissions. To export just the root table (no repeat data nor media files), call this endpoint instead, which directly gives CSV data. An OData-style \$filter query filters the Submissions. This endpoint requires ODK Central v1.1 or later.

Value

A tibble with the root Submission table, one row per Submission.

See Also

<https://docs.getodk.org/central-api-submission-management/#exporting-root-data-to-plain-csv>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`,

```
submission_list(), submission_restore(), submission_review(), submission_submitters(),
submission_update(), submission_version_attachment_delete(), submission_version_attachment_download(),
submission_version_attachment_list(), submission_version_detail(), submission_version_geojson(),
submission_version_xml(), submission_versions()
```

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

csv <- submission_csv()

csv |> knitr::kable()

## End(Not run)
```

submission_delete	Delete a Submission.
-------------------	----------------------

Description

[Experimental]

Usage

```
submission_delete(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

- iid The instance_id, a UUID, as returned by [submission_list](#).
- pid The numeric ID of the project, e.g.: 2.
Default: [get_default_pid](#).
Set default pid through `ru_setup(pid="...")`.
See `vignette("Setup", package = "ruODK")`.
- fid The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147".
Default: [get_default_fid](#).
Set default fid through `ru_setup(fid="...")`.
See `vignette("Setup", package = "ruODK")`.

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

When a Submission is deleted, there is a period of time (30 days) during which it can be restored with `submission_restore()`. After this time, all of its resources and attachments are automatically purged. Deletion is soft. The Submission stays restorable for 30 days.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-submission-management/#deleting-a-submission>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

submission_delete(sl$instance_id[[1]])
# > $success
```

```
# > [1] TRUE

## End(Not run)
```

submission_detail	Show metadata for one submission.
-------------------	-----------------------------------

Description

[Stable]

Usage

```
submission_detail(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Value

A nested list of submission metadata.

See Also

<https://docs.getodk.org/central-api-submission-management/#getting-submission-metadata>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

sub <- submission_detail(sl$instance_id[[1]])

# The details for one submission return exactly one row
nrow(sub)
# > 1

# The columns are metadata about the submission and the submitter
names(sub)
# > "instance_id" "submitter_id" "submitter_type" "submitter_display_name"
# > "submitter_created_at" "device_id" "created_at"

## End(Not run)
```

submission_draft_attachment_delete

Clear an attachment of a Draft Submission.

Description

[Experimental]

Usage

```
submission_draft_attachment_delete(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  filename,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
filename	(character) The name of the expected file slot, as given by <code>submission_draft_attachment_list()</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

A DELETE to a Draft Submission attachment's endpoint clears the uploaded bytes while the expected file slot remains listed.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-submission-management/#draft-submissions>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

sl <- submission_draft_list(fid = fid)
iid <- sl$instance_id[[1]]

submission_draft_attachment_delete(iid, fid = fid, filename = "photo.jpg")
# > $success
# > [1] TRUE

## End(Not run)
```

submission_draft_attachment_download

Download one attachment of a Draft Submission.

Description

[Experimental]

Usage

```
submission_draft_attachment_download(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  filename,
```

```

dest = NULL,
url = get_default_url(),
un = get_default_un(),
pw = get_default_pw(),
retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
filename	(character) The name of the file to download, as given by <code>submission_draft_attachment_list()</code> .
dest	(character) The local file path to save the attachment to. Default: NULL, which saves to a temporary file with the same extension.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

If the server holds a copy of the expected file, append its filename to the Draft Submission attachments URL to download only that file.

Value

The local file path the attachment was saved to.

See Also

<https://docs.getodk.org/central-api-submission-management/#draft-submissions>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

sl <- submission_draft_list(fid = fid)
iid <- sl$instance_id[[1]]

al <- submission_draft_attachment_list(iid, fid = fid)

path <- submission_draft_attachment_download(iid, fid = fid, filename = al$name[[1]]) # nolint

file.info(path)$size

## End(Not run)
```

`submission_draft_attachment_list`

List expected attachments of one Draft Submission.

Description

[Experimental]

Usage

```
submission_draft_attachment_list(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
```

```

url = get_default_url(),
un = get_default_un(),
pw = get_default_pw(),
retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

When a Draft Submission is created, its XML data determines which file attachments it references. This lists the expected attachments with a boolean flag telling whether the server holds a copy of each file or not.

Value

A tibble with one row per expected attachment and the columns name and exists.

See Also

<https://docs.getodk.org/central-api-submission-management/#draft-submissions>

Other submission-management: [attachment_delete\(\)](#), [attachment_list\(\)](#), [attachment_upload\(\)](#), [encryption_key_list\(\)](#), [submission_audit_get\(\)](#), [submission_changes\(\)](#), [submission_comment_create\(\)](#),

```

submission_comment_list(), submission_create(), submission_csv(), submission_delete(),
submission_detail(), submission_draft_attachment_delete(), submission_draft_attachment_download(),
submission_draft_attachment_upload(), submission_draft_create(), submission_draft_export(),
submission_draft_get(), submission_draft_keys(), submission_draft_list(), submission_edit(),
submission_export(), submission_geodata(), submission_geojson(), submission_get(),
submission_list(), submission_restore(), submission_review(), submission_submitters(),
submission_update(), submission_version_attachment_delete(), submission_version_attachment_download(),
submission_version_attachment_list(), submission_version_detail(), submission_version_geojson(),
submission_version_xml(), submission_versions()

```

Examples

```

## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

sl <- submission_draft_list(fid = fid)

al <- submission_draft_attachment_list(sl$instance_id[[1]], fid = fid)

al |> knitr::kable()

## End(Not run)

```

submission_draft_attachment_upload

Upload an attachment of a Draft Submission.

Description

[Experimental]

Usage

```

submission_draft_attachment_upload(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  filename,
  file,
  content_type = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
filename	(character) The name of the expected file slot, as given by <code>submission_draft_attachment_list()</code> .
file	(character) Path to a local file whose bytes are uploaded to the expected file slot.
content_type	(character) The MIME type sent as Content-Type header. Default: NULL, which sends "application/octet-stream".
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

To upload a binary to an expected file slot of a Draft Submission, POST the binary to its endpoint. Supply a content_type if you have one.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-submission-management/#draft-submissions>

Other submission-management: [attachment_delete\(\)](#), [attachment_list\(\)](#), [attachment_upload\(\)](#), [encryption_key_list\(\)](#), [submission_audit_get\(\)](#), [submission_changes\(\)](#), [submission_comment_create\(\)](#), [submission_comment_list\(\)](#), [submission_create\(\)](#), [submission_csv\(\)](#), [submission_delete\(\)](#), [submission_detail\(\)](#), [submission_draft_attachment_delete\(\)](#), [submission_draft_attachment_download\(\)](#), [submission_draft_attachment_list\(\)](#), [submission_draft_create\(\)](#), [submission_draft_export\(\)](#),


```

submission_draft_get(), submission_draft_keys(), submission_draft_list(), submission_edit(),
submission_export(), submission_geodata(), submission_geojson(), submission_get(),
submission_list(), submission_restore(), submission_review(), submission_submitters(),
submission_update(), submission_version_attachment_delete(), submission_version_attachment_download(),
submission_version_attachment_list(), submission_version_detail(), submission_version_geojson(),
submission_version_xml(), submission_versions()

```

Examples

```

## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

sl <- submission_draft_list(fid = fid)
iid <- sl$instance_id[[1]]

submission_draft_attachment_upload(
  iid,
  fid = fid,
  filename = "photo.jpg",
  file = "/path/to/photo.jpg"
)

## End(Not run)

```

submission_draft_create

Create a Submission on a Draft Form.

Description

[Experimental]

Usage

```

submission_draft_create(
  pid = get_default_pid(),
  fid = get_default_fid(),
  xml = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
xml	(character) The Submission XML as a single string.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

To create a Submission on a Draft Form by REST, POST the Submission XML to this endpoint. Draft Submissions test a Draft Form definition before publication. If a Submission already exists with the given instanceID, Central answers 409.

Value

A list with the new Draft Submission's metadata as per the ODK Central API docs. Top level list elements are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-submission-management/#draft-submissions>

Other submission-management: [attachment_delete\(\)](#), [attachment_list\(\)](#), [attachment_upload\(\)](#), [encryption_key_list\(\)](#), [submission_audit_get\(\)](#), [submission_changes\(\)](#), [submission_comment_create\(\)](#), [submission_comment_list\(\)](#), [submission_create\(\)](#), [submission_csv\(\)](#), [submission_delete\(\)](#), [submission_detail\(\)](#), [submission_draft_attachment_delete\(\)](#), [submission_draft_attachment_download\(\)](#), [submission_draft_attachment_list\(\)](#), [submission_draft_attachment_upload\(\)](#), [submission_draft_export\(\)](#), [submission_draft_get\(\)](#), [submission_draft_keys\(\)](#), [submission_draft_list\(\)](#), [submission_edit\(\)](#), [submission_export\(\)](#), [submission_geodata\(\)](#), [submission_geojson\(\)](#), [submission_get\(\)](#), [submission_list\(\)](#), [submission_restore\(\)](#), [submission_review\(\)](#), [submission_submitters\(\)](#),

```

submission_update(), submission_version_attachment_delete(), submission_version_attachment_download(),
submission_version_attachment_list(), submission_version_detail(), submission_version_geojson(),
submission_version_xml(), submission_versions()

```

Examples

```

## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

s <- submission_draft_create(
  fid = fl$fid[[1]],
  xml = "<data id=\"simple\"><meta><instanceID>uuid:1</instanceID></meta></data>" # nolint
)

s$instance_id

## End(Not run)

```

submission_draft_export

Export Draft Submissions to CSV.

Description

[Experimental]

Usage

```

submission_draft_export(
  pid = get_default_pid(),
  fid = get_default_fid(),
  dest = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
-----	--

fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
dest	(character) The local file path to save the ZIP export to. Default: NULL, which saves to a temporary .zip file.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This downloads the ZIP export of a Draft Form's Submissions: the Submission data with related repeat table files and media files.

Value

The local file path the ZIP export was saved to.

See Also

<https://docs.getodk.org/central-api-submission-management/#draft-submissions>

Other submission-management: [attachment_delete\(\)](#), [attachment_list\(\)](#), [attachment_upload\(\)](#), [encryption_key_list\(\)](#), [submission_audit_get\(\)](#), [submission_changes\(\)](#), [submission_comment_create\(\)](#), [submission_comment_list\(\)](#), [submission_create\(\)](#), [submission_csv\(\)](#), [submission_delete\(\)](#), [submission_detail\(\)](#), [submission_draft_attachment_delete\(\)](#), [submission_draft_attachment_download\(\)](#), [submission_draft_attachment_list\(\)](#), [submission_draft_attachment_upload\(\)](#), [submission_draft_create\(\)](#), [submission_draft_get\(\)](#), [submission_draft_keys\(\)](#), [submission_draft_list\(\)](#), [submission_edit\(\)](#), [submission_export\(\)](#), [submission_geodata\(\)](#), [submission_geojson\(\)](#), [submission_get\(\)](#), [submission_list\(\)](#), [submission_restore\(\)](#), [submission_review\(\)](#), [submission_submitters\(\)](#), [submission_update\(\)](#), [submission_version_attachment_delete\(\)](#), [submission_version_attachment_download\(\)](#), [submission_version_attachment_list\(\)](#), [submission_version_detail\(\)](#), [submission_version_geojson\(\)](#), [submission_version_xml\(\)](#), [submission_versions\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

zip <- submission_draft_export(fid = fl$fid[[1]])

file.info(zip)$size

## End(Not run)
```

submission_draft_get *Download one Draft Submission.*

Description

[Experimental]

Usage

```
submission_draft_get(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This returns the XML of one Submission made to a Draft Form as a nested list. Repeating subgroups stay nested, since their presence and length depend on the Submission data.

Value

A nested list of Draft Submission data.

See Also

<https://docs.getodk.org/central-api-submission-management/#draft-submissions>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()
fid <- fl$fid[[1]]

sl <- submission_draft_list(fid = fid)

sub <- submission_draft_get(sl$instance_id[[1]], fid = fid)

names(sub)

## End(Not run)
```

submission_draft_keys *List encryption keys of Draft Submissions.*

Description

[Experimental]

Usage

```
submission_draft_keys(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .

tz A timezone to convert dates and times to.
 Read vignette("setup", package = "ruODK") to learn how ruODK's timezone can be set globally or per function.

Details

This lists the managed and self-supplied encryption keys of Submissions made to a Draft Form.

Value

A tibble with one row per encryption key and all key metadata as columns as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-submission-management/#draft-submissions>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

kl <- submission_draft_keys(fid = fl$fid[[1]])

kl |> knitr::kable()

## End(Not run)
```

`submission_draft_list` *List all Submissions of a Draft Form.*

Description

[Experimental]

Usage

```
submission_draft_list(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.

Details

This lists every Submission made to the Draft Form, every time. Draft Submissions test a Draft Form definition before publication.

Value

A tibble with one row per Draft Submission and all Submission metadata as columns as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-submission-management/#listing-all-submissions-on-a-draft-form>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

fl <- form_list()

sl <- submission_draft_list(fid = fl$fid[[1]])

sl |> knitr::kable()

## End(Not run)
```

submission_edit

Edit one Submission field and leave a comment.

Description

[Experimental]

Usage

```
submission_edit(
  iid,
  field,
  value,
  comment = NULL,
```

```

    pid = get_default_pid(),
    fid = get_default_fid(),
    url = get_default_url(),
    un = get_default_un(),
    pw = get_default_pw(),
    retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
field	(character) The XForms field to edit, e.g. "name" or "/data/name".
value	The new field value, coerced to a string. It must match the field type from <code>form_schema()</code> : integers for "int", numbers for "decimal", anything for "string".
comment	(character) A comment about the edit, posted with <code>submission_comment_create()</code> . Default: NULL (no comment posted).
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This convenience wrapper edits one field of a Submission and, optionally, leaves a comment about the edit. It downloads the current Submission XML, sets the field to the new value, rotates the deprecatedID metadata node to the current instanceID, assigns a new instanceID, uploads the result with `submission_update()`, and posts the comment with `submission_comment_create()`. The new value is checked against the field type from `form_schema()`.

Value

A list with the update result from `submission_update()` and, if a comment was given, the comment result from `submission_comment_create()`.

See Also

<https://docs.getodk.org/central-api-submission-management/#updating-submission-data>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()
iid <- sl$instance_id[[1]]

se <- submission_edit(
  iid = iid,
  field = "name",
  value = "Jo updated",
  comment = "Fixed the name."
)

se$update$current_version$instanceId

## End(Not run)
```

submission_export

Export all form submissions including repeats and attachments to CSV.

Description

[Maturing]

Usage

```

submission_export(
  local_dir = here::here(),
  overwrite = TRUE,
  media = TRUE,
  repeats = TRUE,
  deleted_fields = FALSE,
  split_select_multiples = FALSE,
  group_paths = TRUE,
  filter = NULL,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  pp = get_default_pp(),
  retries = get_retries(),
  odkc_version = get_default_odkc_version(),
  verbose = get_ru_verbose()
)

```

Arguments

<code>local_dir</code>	The local folder to save the downloaded files to, default: <code>here::here</code> .
<code>overwrite</code>	Whether to overwrite previously downloaded zip files, default: <code>FALSE</code>
<code>media</code>	Whether to include media attachments, default: <code>TRUE</code> . This feature only has effect on ODK Central v1.1 and higher. Setting this feature to <code>FALSE</code> with an <code>odkc_version < 1.1</code> and will display a verbose noop message, but still return all media attachments.
<code>repeats</code>	Whether to include repeat data (if <code>TRUE</code>), or whether to return the root table only (<code>FALSE</code>). Default: <code>TRUE</code> . Requesting <code>repeats=FALSE</code> will also omit any media, and override the parameter <code>media</code> . Setting this feature to <code>FALSE</code> with an <code>odkc_version < 1.1</code> and will display a verbose noop message, but still include all repeat data.
<code>deleted_fields</code>	Whether to restore all fields previously deleted from this form for this export (<code>TRUE</code>). All known fields and data for those fields will be merged and exported. default: <code>FALSE</code>
<code>split_select_multiples</code>	Whether to split select multiple answers into columns (<code>TRUE</code>). If <code>TRUE</code> , a boolean column is created for every known select multiple option in the export. The option name is in the field header, and a 0 or a 1 is present in each cell indicating whether that option was checked for that row. Default: <code>FALSE</code> .
<code>group_paths</code>	Whether to keep group path prefixes in field header names (<code>TRUE</code> , e.g. <code>meta-instanceID</code>) or remove them (<code>FALSE</code> , e.g. <code>instanceID</code>). Default: <code>TRUE</code> .
<code>filter</code>	(str) An OData-style <code>\$filter</code> query to filter the Submissions in the export. Only a subset of the <code>\$filter</code> features is available. Default: <code>NULL</code> (no filtering, all Submissions exported).

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pp	The passphrase for an encrypted form. Default: NULL. Passphrases can be stored e.g. as environment variables. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
odkc_version	The ODK Central version as a semantic version string (year.minor.patch), e.g. "2023.5.1". The version is shown on ODK Central's version page <code>/version.txt</code>. Discard the "v". ruODK uses this parameter to adjust for breaking changes in ODK Central. Default: get_default_odkc_version or "2023.5.1" if unset. Set default <code>get_default_odkc_version</code> through <code>ru_setup(odkc_version="2023.5.1")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

This function exports all the Submission data associated with a Form as one zip file containing one or more CSV files, as well as all multimedia attachments associated with the included Submissions.

For an incremental download of a subset of submissions, use [submission_list](#) or [odata_submission_get](#) with filter queries.

Contents:

The inclusion of subtables (from repeating form groups) can be toggled through `repeats`, whereas the inclusion of media attachments can be toggled through `media`. The splitting of select multiple answers into one boolean column per option can be toggled through `split_select_multiples`. The group path prefixes of field header names can be toggled through `group_paths`. The exported rows can be restricted through `filter`.

Download location:

The file will be downloaded to the project root unless specified otherwise (via `local_dir`). Subsequently, the zip file can be extracted. Attachment filenames (e.g. "12345.jpg") should be prepended with `media` (resulting in e.g. `media/12345.jpg`) in order to represent the relative path to the actual attachment file (as extracted from the zip file).

Encryption:

ODK Central supports two modes of encryption - learn about them [here](#). ruODK supports project managed encryption, however the support is limited to exactly one encryption key. The supplied passphrase will be used against the first returned encryption key. Remaining encryption keys are ignored by ruODK.

If an incorrect passphrase is given, the request is terminated immediately. It has been reported that multiple requests with incorrect passphrases can crash ODK Central.

Value

The absolute path to the exported ZIP file named after the form ID. The exported ZIP file will have the extension `.zip` unless only the root table was requested (with `repeats=FALSE`), in which case the exported file will have the extension `.csv`. In contrast to ODK Central, which exports to `submissions.csv(.zip)`, the exported ZIP file is named after the form to avoid accidentally overwriting the ZIP export from another form.

See Also

<https://docs.getodk.org/central-api-submission-management/#exporting-form-submissions-to-csv>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")
```

```

se <- submission_export()

# Unzip and inspect the loot
t <- tempdir()
f <- unzip(se, exdir = t)
fs::dir_ls(t)
fid <- get_test_fid()
sub <- fs::path(t, glue::glue("{fid}.csv")) %>% readr::read_csv()
sub %>% knitr::kable(.)

## End(Not run)

```

submission_geodata	<i>Get Submissions geodata as GeoJSON.</i>
--------------------	--

Description

[Experimental]

Usage

```

submission_geodata(
  pid = get_default_pid(),
  fid = get_default_fid(),
  filter = NULL,
  limit = NULL,
  fieldpath = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See vignette("Setup", package = "ruODK").
filter	(character) An OData-style \$filter query to filter the Submissions by. Default: NULL (not sent).

limit	(numeric) The maximum number of Submissions to extract geoinformation from. Default: NULL (not sent).
fieldpath	(character) The field to extract geoinformation from. At most one field can be given. Default: NULL (the first geofield outside any repeat group).
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This endpoint returns Submissions' geodata, one Form field per Submission, in GeoJSON format. If no particular field path is supplied, the first non-repeat geopoint, geotrace or geoshape field is used. No particular ordering is implied. An OData-style `$filter` query filters the Submissions.

Value

A list with the GeoJSON FeatureCollection as returned by Central: type and features with geometry and properties per Submission.

See Also

<https://docs.getodk.org/central-api-submission-management/#retrieving-submissions-geodata>

Other submission-management: [attachment_delete\(\)](#), [attachment_list\(\)](#), [attachment_upload\(\)](#), [encryption_key_list\(\)](#), [submission_audit_get\(\)](#), [submission_changes\(\)](#), [submission_comment_create\(\)](#), [submission_comment_list\(\)](#), [submission_create\(\)](#), [submission_csv\(\)](#), [submission_delete\(\)](#), [submission_detail\(\)](#), [submission_draft_attachment_delete\(\)](#), [submission_draft_attachment_download\(\)](#), [submission_draft_attachment_list\(\)](#), [submission_draft_attachment_upload\(\)](#), [submission_draft_create\(\)](#), [submission_draft_export\(\)](#), [submission_draft_get\(\)](#), [submission_draft_keys\(\)](#), [submission_draft_list\(\)](#), [submission_edit\(\)](#), [submission_export\(\)](#), [submission_geojson\(\)](#), [submission_get\(\)](#), [submission_list\(\)](#), [submission_restore\(\)](#), [submission_review\(\)](#), [submission_submitters\(\)](#), [submission_update\(\)](#), [submission_version_attachment_delete\(\)](#), [submission_version_attachment_download\(\)](#), [submission_version_attachment_list\(\)](#), [submission_version_detail\(\)](#), [submission_version_geojson\(\)](#), [submission_version_xml\(\)](#), [submission_versions\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

g <- submission_geodata()

g$type
# > "FeatureCollection"

## End(Not run)
```

submission_geojson	<i>Get the GeoJSON of one Submission.</i>
--------------------	---

Description**[Experimental]****Usage**

```
submission_geojson(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This endpoint returns the geodata of a single Submission in GeoJSON format.

Value

A list with the GeoJSON FeatureCollection as returned by Central: type and features with geometry and properties.

See Also

<https://docs.getodk.org/central-api-submission-management/#retrieving-geojson-of-a-single-submission>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "....svc", un = "me@email.com", pw = "...")

sl <- submission_list()

g <- submission_geojson(sl$instance_id[[1]])

g$type
```

```
# > "FeatureCollection"

## End(Not run)
```

submission_get	<i>Get submissions for a list of submission instance IDs.</i>
----------------	---

Description

Uses [get_one_submission](#) on a list of submission instance IDs (`iid`) as returned from `submission_list$instance_id`. By giving the list of `iid` to download explicitly, that list can be modified using information not accessible to ruODK, e.g. `iid` can be restricted to "only not already downloaded submissions".

Usage

```
submission_get(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

<code>iid</code>	A list of submission instance IDs, e.g. from <code>submission_list\$instance_id</code> .
<code>pid</code>	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default <code>pid</code> through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
<code>fid</code>	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default <code>fid</code> through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
<code>url</code>	The ODK Central base URL without trailing slash. Default: get_default_url . Set default <code>url</code> through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
<code>un</code>	The ODK Central username (an email address). Default: get_default_un . Set default <code>un</code> through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
<code>pw</code>	The ODK Central password. Default: get_default_pw . Set default <code>pw</code> through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").

retries The number of attempts to retrieve a web resource.
 This parameter is given to `RETRY`(times = retries).
 Default: 3.

Details

Forms with submission audit enabled will also receive the submission audit as `audit.csv`. This will overwrite all previous `audit.csv` files. To get the combined submission audit logs as one single, concatenated `audit.csv` file, use `submission_export`. Note that the submission audit is different from the server audit. The submission audit logs the actions of the enumerator in ODK Collect, whereas the form audit logs the actions of the project manager in ODK Central.

Value

A nested list of submission data.

See Also

<https://docs.getodk.org/central-api-submission-management/#retrieving-submission-xml>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# Step 1: Setup ruODK with OData Service URL (has url, pid, fid)
ruODK::ru_setup(svc = "...")

# Step 2: List all submissions of form
sl <- submission_list()

# Step 3: Get submissions
subs <- submission_get(sl$instance_id)

## End(Not run)
```

submission_list	<i>List all submissions of one form.</i>
-----------------	--

Description**[Stable]****Usage**

```
submission_list(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un. Set default un through <code>ru_setup(un="...")</code>. See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw. Set default pw through <code>ru_setup(pw="...")</code>. See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY(times = retries). Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.

tz A timezone to convert dates and times to.
 Read `vignette("setup", package = "ruODK")` to learn how ruODK's timezone can be set globally or per function.

Value

A tibble containing some high-level details of the form submissions. One row per submission, columns are submission attributes:

- * `instance_id`: uuid, string. The unique ID for each submission.
- * `submitter_id`: user ID, integer.
- * `created_at`: time of submission upload, dttm
- * `updated_at`: time of submission update on server, dttm or NA

See Also

<https://docs.getodk.org/central-api-submission-management/#listing-all-submissions-on-a-form>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# Set default credentials, see vignette("setup")
ruODK::ru_setup(
  svc = ...,
  un = "me@email.com",
  pw = "..."
)

sl <- submission_list()
sl %>% knitr::kable(.)

fl <- form_list()

# submission_list returns a tibble
class(sl)
# > c("tbl_df", "tbl", "data.frame")

# Submission attributes are the tibble's columns
names(sl)
```

```
# > "instance_id" "submitter_id" "device_id" "created_at" "updated_at"

# Number of submissions (rows) is same as advertised in form_list
form_list_nsub <- fl %>%
  filter(fid == get_test_fid()) %>%
  magrittr::extract2("submissions") %>%
  as.numeric()
nrow(sl) == form_list_nsub
# > TRUE

## End(Not run)
```

submission_restore	<i>Restore a deleted Submission.</i>
--------------------	--------------------------------------

Description

[Experimental]

Usage

```
submission_restore(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Restores a Submission deleted with `submission_delete()` within its 30-day grace period. After this time, all of its resources and attachments are automatically purged and the Submission can no longer be restored.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-submission-management/#restoring-a-deleted-submission>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

submission_restore(sl$instance_id[[1]])
# > $success
# > [1] TRUE

## End(Not run)
```

submission_review	<i>Review a Submission.</i>
-------------------	-----------------------------

Description**[Experimental]****Usage**

```
submission_review(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  review_state = "",
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
review_state	(character) The new review state of the Submission. One of "approved", "hasIssues" or "rejected".
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

Currently, the only updatable metadata on a Submission is its reviewState. To update the Submission data itself, see `submission_update()`.

Starting with Version 2022.3, changing the reviewState of a Submission to approved can create an Entity in a Dataset if the corresponding Form maps Dataset Properties to Form Fields.

This function records a review decision on a Submission. Clearing the review state back to unreviewed (received) is not supported by this function.

Value

A list with the reviewed Submission's metadata as per the ODK Central API docs. Top level list elements are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-submission-management/#updating-submission-metadata>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

s <- submission_review(sl$instance_id[[1]], review_state = "approved")

s$review_state
# > "approved"

## End(Not run)
```

submission_submitters *List all submitting Actors of one Form.*

Description

[Experimental]

Usage

```
submission_submitters(
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code> .

tz A timezone to convert dates and times to.
 Read `vignette("setup", package = "ruODK")` to learn how ruODK's timezone can be set globally or per function.

Details

This endpoint provides a listing of all known submitting Actors to a given Form. Each Actor that has submitted to the given Form is returned once.

Value

A tibble with one row per submitting Actor and all Actor metadata as columns as per the ODK Central API docs. Column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-submission-management/#listing-submitters>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

st <- submission_submitters()

st |> knitr::kable()

## End(Not run)
```

submission_update	<i>Update Submission data.</i>
-------------------	--------------------------------

Description

[Experimental]

Usage

```
submission_update(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  xml = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

<code>iid</code>	The instance_id, a UUID, as returned by submission_list .
<code>pid</code>	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>fid</code>	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>xml</code>	(character) The complete new Submission XML as a single string, including the deprecatedID metadata node.
<code>url</code>	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>un</code>	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>pw</code>	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>retries</code>	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

You can use this endpoint to submit updates to an existing Submission. The instanceID submitted with the initial version of the Submission is used permanently to reference that Submission logically, which is to say the initial Submission and all its subsequent versions. Each subsequent version also provides its own instanceID.

To perform an update, provide in the Submission XML an additional deprecatedID metadata node with the instanceID of the particular and current Submission version you are replacing. If the deprecatedID you give is anything other than the identifier of the current version of the Submission at the time the server receives it, Central answers 409. deprecatedID must contain the current instanceID. submission_get() returns the current instanceID. The new version must have a new instanceID.

The XML data you send replaces the existing data entirely. All of the data must be present in the updated XML.

When you create a new Submission version, any uploaded media files attached to the current version that match expected attachment names in the new version are automatically copied over to the new version.

This endpoint requires ODK Central v1.2 or later.

Value

A list with the updated Submission's metadata as per the ODK Central API docs. Top level list elements are renamed from ODK's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-submission-management/#updating-submission-data>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()
iid <- sl$instance_id[[1]]

xml <- paste0(
  '<data id="simple" version="1">',
  '<meta>',
  '<instanceID>uid:85cb9aff-005e-4edd-9739-dc9c1a829c45</instanceID>',
  '<deprecatedID>', iid, '</deprecatedID>',
  '</meta>',
  '<name>Jo updated</name>',
```

```

    "</data>"
  )

  s <- submission_update(iid = iid, xml = xml)

  s$current_version$instanceId
  # > "uuid:85cb9aff-005e-4edd-9739-dc9c1a829c45"

  ## End(Not run)

```

submission_version_attachment_delete

Clear one attachment of a Submission version.

Description

[Experimental]

Usage

```

submission_version_attachment_delete(
  iid,
  vid,
  filename,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
vid	(character) The instanceID of the particular version of this Submission, e.g. from <code>submission_versions()</code> .
filename	(character) The name of the file to clear, as given by <code>submission_version_attachment_list()</code> .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

A DELETE to a version attachment's endpoint clears the uploaded bytes of that version while the expected file slot remains listed.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-submission-management/#clearing-a-submission-version-s-attachment>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()
iid <- sl$instance_id[[1]]

sv <- submission_versions(iid)
vid <- sv$instance_id[[1]]
```

```

submission_version_attachment_delete(iid, vid = vid, filename = "photo.jpg")
# > $success
# > [1] TRUE

## End(Not run)

```

submission_version_attachment_download

Download one attachment of a Submission version.

Description

[Experimental]

Usage

```

submission_version_attachment_download(
  iid,
  vid,
  filename,
  pid = get_default_pid(),
  fid = get_default_fid(),
  dest = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
vid	(character) The instanceID of the particular version of this Submission, e.g. from submission_versions() .
filename	(character) The name of the file to download, as given by submission_version_attachment_list() .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See vignette("Setup", package = "ruODK").

<code>dest</code>	(character) The local file path to save the attachment to. Default: <code>NULL</code> , which saves to a temporary file with the same extension.
<code>url</code>	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>un</code>	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>pw</code>	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
<code>retries</code>	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

If the server holds a copy of the expected file, append its filename to the version attachments URL to download only that file.

Value

The local file path the attachment was saved to.

See Also

<https://docs.getodk.org/central-api-submission-management/#downloading-a-version-s-attachment>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

s1 <- submission_list()
iid <- s1$instance_id[[1]]
```

```

sv <- submission_versions(iid)
vid <- sv$instance_id[[1]]

al <- submission_version_attachment_list(iid, vid = vid)

path <- submission_version_attachment_download(iid, vid = vid, filename = al$name[[1]]) # nolint

file.info(path)$size

## End(Not run)

```

submission_version_attachment_list

List expected attachments of one Submission version.

Description

[Experimental]

Usage

```

submission_version_attachment_list(
  iid,
  vid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
vid	(character) The instanceID of the particular version of this Submission, e.g. from <code>submission_versions()</code> .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This lists the expected Submission attachments for the given version, along with a boolean flag telling whether the server holds a copy of each expected file or not.

Value

A tibble with one row per expected attachment and the columns name and exists.

See Also

<https://docs.getodk.org/central-api-submission-management/#listing-version-expected-attachments>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()
iid <- sl$instance_id[[1]]

sv <- submission_versions(iid)
```

```

al <- submission_version_attachment_list(iid, vid = sv$instance_id[[1]])

al |> knitr::kable()

## End(Not run)

```

submission_version_detail

Show details of one Submission version.

Description

[Experimental]

Usage

```

submission_version_detail(
  iid,
  vid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
vid	(character) The instanceID of the particular version of this Submission, e.g. from <code>submission_versions()</code> .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid. Set default pid through <code>ru_setup(pid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid. Set default fid through <code>ru_setup(fid="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
url	The ODK Central base URL without trailing slash. Default: get_default_url. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Returns metadata about a particular version of the Submission. As with the normal Submission endpoint, this route returns only metadata in JSON.

Value

A tibble with one row holding the Submission version's metadata as columns as per the ODK Central API docs. Column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-submission-management/#getting-version-details>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_geojson()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()
iid <- sl$instance_id[[1]]

sv <- submission_versions(iid)

vd <- submission_version_detail(iid, vid = sv$instance_id[[1]])

vd |> knitr::kable()

## End(Not run)
```

```
submission_version_geojson
```

Get the GeoJSON of one Submission version.

Description

[Experimental]

Usage

```
submission_version_geojson(  
  iid,  
  vid,  
  pid = get_default_pid(),  
  fid = get_default_fid(),  
  url = get_default_url(),  
  un = get_default_un(),  
  pw = get_default_pw(),  
  retries = get_retries()  
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
vid	(character) The instanceID of the particular version of this Submission, e.g. from <code>submission_versions()</code> .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

retries The number of attempts to retrieve a web resource.
 This parameter is given to `RETRY`(times = retries).
 Default: 3.

Details

This endpoint returns the geodata of a single Submission version in GeoJSON format.

Value

A list with the GeoJSON FeatureCollection as returned by Central: type and features with geometry and properties.

See Also

<https://docs.getodk.org/central-api-submission-management/#getting-version-geojson>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_xml()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()
iid <- sl$instance_id[[1]]

sv <- submission_versions(iid)

g <- submission_version_geojson(iid, vid = sv$instance_id[[1]])

g$type
# > "FeatureCollection"

## End(Not run)
```

```
submission_version_xml
```

Show the XML of one Submission version.

Description

[Experimental]

Usage

```
submission_version_xml(
  iid,
  vid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
vid	(character) The instanceID of the particular version of this Submission, e.g. from <code>submission_versions()</code> .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .

retries The number of attempts to retrieve a web resource.
 This parameter is given to `RETRY(times = retries)`.
 Default: 3.

Details

Returns the XML of a particular version of the Submission.

Value

A nested list of the Submission version's XML data.

See Also

<https://docs.getodk.org/central-api-submission-management/#getting-version-xml>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_versions()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()
iid <- sl$instance_id[[1]]

sv <- submission_versions(iid)

vx <- submission_version_xml(iid, vid = sv$instance_id[[1]])

listviewer::jsonedit(vx)

## End(Not run)
```

submission_versions *List all versions of one Submission.*

Description

[Experimental]

Usage

```
submission_versions(
  iid,
  pid = get_default_pid(),
  fid = get_default_fid(),
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz()
)
```

Arguments

iid	The instance_id, a UUID, as returned by submission_list .
pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
fid	The alphanumeric form ID, e.g. "build_Spotlighting-0-8_1559885147". Default: get_default_fid . Set default fid through <code>ru_setup(fid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

orders (vector of character) Orders of datetime elements for lubridate.
Default: `c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")`.

tz A timezone to convert dates and times to.
Read `vignette("setup", package = "ruODK")` to learn how ruODK's timezone can be set globally or per function.

Details

This returns all Submission metadata for every version of this Submission, in descending creation order.

Value

A tibble with one row per Submission version and all version metadata as columns as per the ODK Central API docs. Column names are renamed from ODK Central's camelCase to snake_case.

See Also

<https://docs.getodk.org/central-api-submission-management/#listing-versions>

Other submission-management: `attachment_delete()`, `attachment_list()`, `attachment_upload()`, `encryption_key_list()`, `submission_audit_get()`, `submission_changes()`, `submission_comment_create()`, `submission_comment_list()`, `submission_create()`, `submission_csv()`, `submission_delete()`, `submission_detail()`, `submission_draft_attachment_delete()`, `submission_draft_attachment_download()`, `submission_draft_attachment_list()`, `submission_draft_attachment_upload()`, `submission_draft_create()`, `submission_draft_export()`, `submission_draft_get()`, `submission_draft_keys()`, `submission_draft_list()`, `submission_edit()`, `submission_export()`, `submission_geodata()`, `submission_geojson()`, `submission_get()`, `submission_list()`, `submission_restore()`, `submission_review()`, `submission_submitters()`, `submission_update()`, `submission_version_attachment_delete()`, `submission_version_attachment_download()`, `submission_version_attachment_list()`, `submission_version_detail()`, `submission_version_geojson()`, `submission_version_xml()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

sl <- submission_list()

sv <- submission_versions(sl$instance_id[[1]])

sv |> knitr::kable()

## End(Not run)
```

user_create

*Create a new User.***Description****[Experimental]****Usage**

```

user_create(
  email,
  password = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)

```

Arguments

email	(character) The email address of the User account to create.
password	(character) If provided, the User account is created with this password. Default: NULL (the User sets the password later).
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

All that is required to create a new User is an email address. That email address will receive a message instructing the new User on how to claim the new account and set a password. Optionally, a password may also be supplied as a part of this request. If it is, the account is immediately usable with the given credentials. Users are not able to do anything upon creation besides log in and change their own profile information. To allow Users to perform useful actions, assign them one or more Roles.

Value

A tibble with one row holding the new User's metadata as columns.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#creating-a-new-user>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

u <- user_create(email = "new.user@example.com")

u |> knitr::kable()

## End(Not run)
```

user_delete

Delete a User.

Description

[Experimental]

Usage

```
user_delete(
  actor_id,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

actor_id (numeric) The integer ID of the User.

url The ODK Central base URL without trailing slash.
Default: `get_default_url()`.
Set default url through `ru_setup(url="...")`.
See vignette("Setup", package = "ruODK").

un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Upon User deletion, the account is removed and the User is logged out of all existing sessions. The User record remains on file within the database, so that when for example information about the creator of a Form or Submission is requested, basic details are still available on file. A new User account may be created with the same email address as any deleted account.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#deleting-a-user>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

ul <- user_list()

user_delete(actor_id = ul$id[[1]])
# > $success
# > [1] TRUE

## End(Not run)
```

user_detail	Show details of one User.
-------------	---------------------------

Description

[Experimental]

Usage

```
user_detail(
  actor_id = "current",
  extended = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

actor_id	The integer ID of the User, or "current" for the authenticated User. Default: "current".
extended	(lgl) If TRUE, request extended metadata with the X-Extended-Metadata header. Only used with actor_id = "current". Default: FALSE.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

Supply the integer ID to get the User with that ID, or the text current for the currently authenticated User. With current, extended metadata can be requested: the verbs the authenticated Actor can perform server-wide, and the User's preferences.

Value

A tibble with one row holding the User's metadata as columns as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#getting-user-details>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

me <- user_detail()

me$display_name

## End(Not run)
```

user_list	List all users.
-----------	-----------------

Description

[Maturing]

Usage

```
user_list(
  qry = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries(),
  orders = get_default_orders(),
  tz = get_default_tz(),
  verbose = get_ru_verbose()
)
```

Arguments

qry	A query string to filter users by. The query string is not case sensitive and can contain special characters, such as @. The query string must be at least 5 alphabetic characters long to return good enough matches. E.g. janet will match a user with display name Janette Doe.
-----	---

	E.g., @dbca.wa will match users with an email from dbca.wa.gov.au, whereas @dbca.w or @dbca will return no matches. Default: NULL.
url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code>. Set default url through <code>ru_setup(url="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
un	The ODK Central username (an email address). Default: <code>get_default_un</code>. Set default un through <code>ru_setup(un="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
pw	The ODK Central password. Default: <code>get_default_pw</code>. Set default pw through <code>ru_setup(pw="...")</code>. See <code>vignette("Setup", package = "ruODK")</code>.
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code>. Default: 3.
orders	(vector of character) Orders of datetime elements for lubridate. Default: <code>c("YmdHMS", "YmdHMSz", "Ymd HMS", "Ymd HMSz", "Ymd", "ymd")</code>.
tz	A timezone to convert dates and times to. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's timezone can be set globally or per function.
verbose	Whether to display debug messages or not. Read <code>vignette("setup", package = "ruODK")</code> to learn how ruODK's verbosity can be set globally or per function.

Details

Currently, there are no paging or filtering options, so listing Users will get you every User in the system, every time. Optionally, a `q` query string parameter may be provided to filter the returned users by any given string. The search is performed via a trigram similarity index over both the Email and Display Name fields, and results are ordered by match score, best matches first. If a `q` parameter is given, and it exactly matches an email address that exists in the system, that user's details will always be returned, even for actors who cannot `user.list`. The request must still authenticate as a valid Actor. This allows non-Administrators to choose a user for an action (e.g. grant rights) without allowing full search.

Actors who cannot `user.list` will always receive `[]` with a 200 OK response. ruODK does not (yet) warn if this is the case, and you (the requesting Actor) have no permission to `user.list`.

Value

A tibble with user details as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#listing-all-users>
<https://www.postgresql.org/docs/9.6/pgtrgm.html>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

# All users
ul <- user_list()

# Search users
# Given a user with display name "Janette Doe" and email "@org.com.au"
user_list(qry = "jan") # no results, query string too short
user_list(qry = "jane") # no results, query string too short
user_list(qry = "janet") # returns Janette
user_list(qry = "@org") # no results, query string too short
user_list(qry = "@org.c") # no results, query string too short
user_list(qry = "@org.co") # returns all users matching "@org.co"

# Actor not allowed to user.list
user_list() # If this is empty, you might not have permissions to list users

## End(Not run)
```

`user_preference_project_delete`

Delete a project preference of the authenticated User.

Description

[Experimental]

Usage

```
user_preference_project_delete(
  pid = get_default_pid(),
  name,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
name	(character) The name of the preference.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This removes a per-project preference previously stored with `user_preference_project_set()`.

Value

A list with a single element `success` (TRUE). Central answers a successful delete with HTTP 200 and a bare body, which this function reports as success.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#deleting-a-project-preference>

Other user-management: [app_user_create\(\)](#), [app_user_delete\(\)](#), [app_user_list\(\)](#), [assignment_actors\(\)](#), [assignment_grant\(\)](#), [assignment_list\(\)](#), [assignment_revoke\(\)](#), [form_assignment_list\(\)](#), [project_assignment_list\(\)](#), [role_detail\(\)](#), [role_list\(\)](#), [user_create\(\)](#), [user_delete\(\)](#), [user_detail\(\)](#), [user_list\(\)](#), [user_preference_project_set\(\)](#), [user_preference_site_delete\(\)](#), [user_preference_site_set\(\)](#), [user_reset_password\(\)](#), [user_update\(\)](#), [user_update_password\(\)](#)

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

user_preference_project_delete(name = "formTrashCollapsed")
# > $success
# > [1] TRUE

## End(Not run)
```

user_preference_project_set

Set a project preference of the authenticated User.

Description

[Experimental]

Usage

```
user_preference_project_set(
    pid = get_default_pid(),
    name,
    value,
    url = get_default_url(),
    un = get_default_un(),
    pw = get_default_pw(),
    retries = get_retries()
)
```

Arguments

pid	The numeric ID of the project, e.g.: 2. Default: get_default_pid . Set default pid through <code>ru_setup(pid="...")</code> . See vignette("Setup", package = "ruODK").
name	(character) The name of the preference.
value	The preference value to store.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See vignette("Setup", package = "ruODK").
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See vignette("Setup", package = "ruODK").
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See vignette("Setup", package = "ruODK").
retries	The number of attempts to retrieve a web resource. This parameter is given to RETRY (times = retries). Default: 3.

Details

The Central frontend uses this API to save various user preferences, such as the sorting order of certain listings. The preference value may be of any JSON-serializable type.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#setting-a-project-preference>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

user_preference_project_set(name = "formTrashCollapsed", value = TRUE)
# > $success
# > [1] TRUE

## End(Not run)
```

user_preference_site_delete

Delete a sitewide preference of the authenticated User.

Description

[Experimental]

Usage

```
user_preference_site_delete(
  name,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

name (character) The name of the preference.

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

This removes a sitewide preference previously stored with `user_preference_site_set()`.

Value

A list with a single element `success` (TRUE). Central answers a successful delete with HTTP 200 and a bare body, which this function reports as success.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#deleting-a-sitewide-preference>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

user_preference_site_delete(name = "projectSortMode")
# > $success
# > [1] TRUE

## End(Not run)
```

user_preference_site_set

Set a sitewide preference of the authenticated User.

Description

[Experimental]

Usage

```
user_preference_site_set(
    name,
    value,
    url = get_default_url(),
    un = get_default_un(),
    pw = get_default_pw(),
    retries = get_retries()
)
```

Arguments

name	(character) The name of the preference.
value	The preference value to store.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

The Central frontend uses this API to save various user preferences, such as the sorting order of certain listings. The preference value may be of any JSON-serializable type.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#setting-a-sitewide-preference>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_reset_password()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "....svc", un = "me@email.com", pw = "...")

user_preference_site_set(name = "projectSortMode", value = "latest")
# > $success
# > [1] TRUE

## End(Not run)
```

user_reset_password	<i>Initiate a User password reset.</i>
---------------------	--

Description

[Experimental]

Usage

```
user_reset_password(
  email,
  invalidate = FALSE,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

email	(character) The email address of the User account whose password is to be reset.
invalidate	(lgl) If TRUE, the User's current password is immediately invalidated, regardless of whether the reset is completed. This requires an authenticated User with permission to do this. Default: FALSE.

url	The ODK Central base URL without trailing slash. Default: <code>get_default_url</code> . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: <code>get_default_un</code> . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: <code>get_default_pw</code> . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

Anybody can initiate a reset of any User's password. An email is sent with instructions on how to complete the password reset; it contains a token that is required to complete the process. If the email address provided does not match any User in the system, that address is still sent an email informing them of the attempt and that no account was found.

Value

A list with a single element success (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#initating-a-password-reset>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_update()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "...svc", un = "me@email.com", pw = "...")

user_reset_password(email = "my.email.address@getodk.org")
# > $success
# > [1] TRUE

## End(Not run)
```

user_update

Modify a User.

Description

[Experimental]

Usage

```
user_update(
  actor_id,
  display_name = NULL,
  email = NULL,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

actor_id	(numeric) The integer ID of the User.
display_name	(character) The friendly display name to associate with this User. Default: NULL (left unchanged).
email	(character) The email address to associate with this User. Default: NULL (left unchanged).
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

JSON data can be patched to update User details. The display name sets the friendly name the web interface uses to refer to the User. The email sets the email address associated with the account. When User details are updated, the `updatedAt` field is updated automatically.

Value

A tibble with one row holding the User's metadata as columns.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#modifying-a-user>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update_password()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "....svc", un = "me@email.com", pw = "...")

ul <- user_list()

u <- user_update(actor_id = ul$id[[1]], display_name = "New Name")

u |> knitr::kable()

## End(Not run)
```

`user_update_password` *Directly update a User password.*

Description

[Experimental]

Usage

```
user_update_password(
  actor_id,
  old_password,
  new_password,
  url = get_default_url(),
  un = get_default_un(),
  pw = get_default_pw(),
  retries = get_retries()
)
```

Arguments

actor_id	(numeric) The integer ID of the User.
old_password	(character) The User's current password.
new_password	(character) The new password the User wishes to set.
url	The ODK Central base URL without trailing slash. Default: get_default_url . Set default url through <code>ru_setup(url="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
un	The ODK Central username (an email address). Default: get_default_un . Set default un through <code>ru_setup(un="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
pw	The ODK Central password. Default: get_default_pw . Set default pw through <code>ru_setup(pw="...")</code> . See <code>vignette("Setup", package = "ruODK")</code> .
retries	The number of attempts to retrieve a web resource. This parameter is given to <code>RETRY(times = retries)</code> . Default: 3.

Details

To directly update a User password, the User's intention must be reproven by supplying the old password alongside the new password. To initiate an email-based password reset process instead, see `user_reset_password()`.

Value

A list with a single element `success` (TRUE) as per the ODK Central API docs.

See Also

<https://docs.getodk.org/central-api-accounts-and-users/#directly-updating-a-user-password>

Other user-management: `app_user_create()`, `app_user_delete()`, `app_user_list()`, `assignment_actors()`, `assignment_grant()`, `assignment_list()`, `assignment_revoke()`, `form_assignment_list()`, `project_assignment_list()`, `role_detail()`, `role_list()`, `user_create()`, `user_delete()`, `user_detail()`, `user_list()`, `user_preference_project_delete()`, `user_preference_project_set()`, `user_preference_site_delete()`, `user_preference_site_set()`, `user_reset_password()`, `user_update()`

Examples

```
## Not run:
# See vignette("setup") for setup and authentication options
# ruODK::ru_setup(svc = "....svc", un = "me@email.com", pw = "...")

u1 <- user_list()

user_update_password(
```

```
    actor_id = ul$id[[1]],  
    old_password = "old.password",  
    new_password = "new.password"  
  )  
  # > $success  
  # > [1] TRUE  
  
## End(Not run)
```

Index

* datasets

- [fq_attachments](#), 159
- [fq_data](#), 160
- [fq_data_strata](#), 161
- [fq_data_taxa](#), 162
- [fq_form_detail](#), 163
- [fq_form_list](#), 163
- [fq_form_schema](#), 164
- [fq_form_xml](#), 165
- [fq_meta](#), 165
- [fq_project_detail](#), 166
- [fq_project_list](#), 167
- [fq_raw](#), 167
- [fq_raw_strata](#), 168
- [fq_raw_taxa](#), 169
- [fq_submission_list](#), 170
- [fq_submissions](#), 170
- [fq_svc](#), 171
- [fq_zip_data](#), 172
- [fq_zip_strata](#), 172
- [fq_zip_taxa](#), 173
- [fs_v7](#), 173
- [fs_v7_raw](#), 174
- [geo_fs](#), 175
- [geo_gj](#), 175
- [geo_gj88](#), 176
- [geo_gj_raw](#), 176
- [geo_wkt](#), 177
- [geo_wkt88](#), 178
- [geo_wkt_raw](#), 178

* entity-management

- [entity_audits](#), 33
- [entity_bulk_delete](#), 36
- [entity_bulk_restore](#), 38
- [entity_changes](#), 40
- [entity_create](#), 42
- [entity_creators](#), 46
- [entity_delete](#), 48
- [entity_detail](#), 50

- [entity_geodata](#), 53
- [entity_geojson](#), 55
- [entity_list](#), 57
- [entity_restore](#), 59
- [entity_update](#), 61
- [entity_versions](#), 64
- [entitylist_create](#), 66
- [entitylist_delete](#), 69
- [entitylist_detail](#), 71
- [entitylist_download](#), 73
- [entitylist_list](#), 76
- [entitylist_property_create](#), 78
- [entitylist_property_delete](#), 80
- [entitylist_trash_download](#), 82
- [entitylist_update](#), 84
- [odata_entitylist_data_get](#), 195
- [odata_entitylist_metadata_get](#), 198
- [odata_entitylist_service_get](#), 201

* form-management

- [form_assignment_actors](#), 86
- [form_assignment_grant](#), 89
- [form_assignment_revoke](#), 92
- [form_attachment_download](#), 95
- [form_attachment_list](#), 97
- [form_create](#), 99
- [form_dataset_diff](#), 101
- [form_delete](#), 103
- [form_detail](#), 105
- [form_draft_attachment_delete](#), 106
- [form_draft_attachment_download](#), 108
- [form_draft_attachment_link](#), 110
- [form_draft_attachment_list](#), 112
- [form_draft_attachment_upload](#), 114
- [form_draft_create](#), 116
- [form_draft_dataset_diff](#), 118
- [form_draft_delete](#), 119
- [form_draft_detail](#), 121
- [form_draft_publish](#), 123

- form_draft_xlsx, 125
- form_draft_xml, 126
- form_link, 128
- form_list, 130
- form_restore, 132
- form_schema, 133
- form_schema_ext, 137
- form_update, 142
- form_version_attachment_download, 144
- form_version_attachment_list, 146
- form_version_detail, 148
- form_version_list, 150
- form_version_xlsx, 152
- form_version_xml, 154
- form_xlsx, 155
- form_xml, 157
- public_link_create, 231
- public_link_delete, 233
- public_link_detail, 235
- public_link_list, 237
- public_link_update, 238
- * included**
 - fq_attachments, 159
 - fq_data, 160
 - fq_data_strata, 161
 - fq_data_taxa, 162
 - fq_form_detail, 163
 - fq_form_list, 163
 - fq_form_schema, 164
 - fq_form_xml, 165
 - fq_meta, 165
 - fq_project_detail, 166
 - fq_project_list, 167
 - fq_raw, 167
 - fq_raw_strata, 168
 - fq_raw_taxa, 169
 - fq_submission_list, 170
 - fq_submissions, 170
 - fq_svc, 171
 - fq_zip_data, 172
 - fq_zip_strata, 172
 - fq_zip_taxa, 173
 - fs_v7, 173
 - fs_v7_raw, 174
 - geo_fs, 175
 - geo_gj, 175
 - geo_gj88, 176
 - geo_gj_raw, 176
 - geo_wkt, 177
 - geo_wkt88, 178
 - geo_wkt_raw, 178
- * messaging**
 - ru_msg_abort, 243
 - ru_msg_info, 244
 - ru_msg_noop, 245
 - ru_msg_success, 245
 - ru_msg_warn, 246
- * odata-api**
 - odata_metadata_get, 203
 - odata_service_get, 204
 - odata_submission_get, 206
- * project-management**
 - form_assignment_role_list, 94
 - project_assignment_actors, 213
 - project_assignment_grant, 215
 - project_assignment_revoke, 218
 - project_create, 220
 - project_delete, 221
 - project_detail, 223
 - project_enable_encryption, 224
 - project_list, 226
 - project_replace, 228
 - project_update, 229
- * ru_settings**
 - odata_svc_parse, 212
 - parse_odkc_version, 212
 - ru_settings, 247
 - ru_setup, 248
 - semver_gt, 251
 - semver_lt, 252
- * server-management**
 - audit_get, 27
 - config_public, 29
- * submission-management**
 - attachment_delete, 17
 - attachment_list, 23
 - attachment_upload, 25
 - encryption_key_list, 31
 - submission_audit_get, 259
 - submission_changes, 261
 - submission_comment_create, 262
 - submission_comment_list, 264
 - submission_create, 266
 - submission_csv, 268
 - submission_delete, 270

- submission_detail, [272](#)
- submission_draft_attachment_delete, [273](#)
- submission_draft_attachment_download, [275](#)
- submission_draft_attachment_list, [277](#)
- submission_draft_attachment_upload, [279](#)
- submission_draft_create, [281](#)
- submission_draft_export, [283](#)
- submission_draft_get, [285](#)
- submission_draft_keys, [287](#)
- submission_draft_list, [288](#)
- submission_edit, [290](#)
- submission_export, [292](#)
- submission_geodata, [296](#)
- submission_geojson, [298](#)
- submission_get, [300](#)
- submission_list, [302](#)
- submission_restore, [304](#)
- submission_review, [306](#)
- submission_submitters, [308](#)
- submission_update, [309](#)
- submission_version_attachment_delete, [312](#)
- submission_version_attachment_download, [314](#)
- submission_version_attachment_list, [316](#)
- submission_version_detail, [318](#)
- submission_version_geojson, [320](#)
- submission_version_xml, [322](#)
- submission_versions, [324](#)
- * user-management**
 - app_user_create, [6](#)
 - app_user_delete, [8](#)
 - app_user_list, [9](#)
 - assignment_actors, [11](#)
 - assignment_grant, [13](#)
 - assignment_list, [14](#)
 - assignment_revoke, [16](#)
 - form_assignment_list, [90](#)
 - project_assignment_list, [217](#)
 - role_detail, [240](#)
 - role_list, [242](#)
 - user_create, [326](#)
 - user_delete, [327](#)
 - user_detail, [329](#)
 - user_list, [330](#)
 - user_preference_project_delete, [332](#)
 - user_preference_project_set, [334](#)
 - user_preference_site_delete, [335](#)
 - user_preference_site_set, [337](#)
 - user_reset_password, [338](#)
 - user_update, [340](#)
 - user_update_password, [341](#)
- * utilities**
 - attachment_get, [19](#)
 - attachment_link, [21](#)
 - drop_null_coords, [30](#)
 - form_schema_parse, [141](#)
 - get_one_attachment, [179](#)
 - get_one_submission, [181](#)
 - get_one_submission_att_list, [183](#)
 - get_one_submission_audit, [185](#)
 - handle_ru_attachments, [187](#)
 - handle_ru_datetimes, [189](#)
 - handle_ru_geopoints, [190](#)
 - handle_ru_geoshapes, [192](#)
 - handle_ru_geotracers, [193](#)
 - odata_submission_rectangle, [210](#)
 - split_geopoint, [253](#)
 - split_geoshape, [254](#)
 - split_geotrace, [256](#)
- app_user_create, [6](#)
- app_user_create(), [9](#), [10](#), [12](#), [14](#), [15](#), [17](#), [91](#), [218](#), [241](#), [243](#), [327](#), [328](#), [330](#), [332](#), [333](#), [335](#), [336](#), [338](#), [339](#), [341](#), [342](#)
- app_user_delete, [8](#)
- app_user_delete(), [7](#), [10](#), [12](#), [14](#), [15](#), [17](#), [91](#), [218](#), [241](#), [243](#), [327](#), [328](#), [330](#), [332](#), [333](#), [335](#), [336](#), [338](#), [339](#), [341](#), [342](#)
- app_user_list, [9](#)
- app_user_list(), [7](#), [9](#), [12](#), [14](#), [15](#), [17](#), [91](#), [218](#), [241](#), [243](#), [327](#), [328](#), [330](#), [332](#), [333](#), [335](#), [336](#), [338](#), [339](#), [341](#), [342](#)
- assignment_actors, [11](#)
- assignment_actors(), [7](#), [9](#), [10](#), [14](#), [15](#), [17](#), [91](#), [218](#), [241](#), [243](#), [327](#), [328](#), [330](#), [332](#), [333](#), [335](#), [336](#), [338](#), [339](#), [341](#), [342](#)
- assignment_grant, [13](#)
- assignment_grant(), [7](#), [9](#), [10](#), [12](#), [15](#), [17](#), [91](#), [218](#), [241](#), [243](#), [327](#), [328](#), [330](#), [332](#)

- [333, 335, 336, 338, 339, 341, 342](#)
- [assignment_list, 14](#)
- [assignment_list\(\), 7, 9, 10, 12, 14, 17, 91, 218, 241, 243, 327, 328, 330, 332, 333, 335, 336, 338, 339, 341, 342](#)
- [assignment_revoke, 16](#)
- [assignment_revoke\(\), 7, 9, 10, 12, 14, 15, 91, 218, 241, 243, 327, 328, 330, 332, 333, 335, 336, 338, 339, 341, 342](#)
- [attachment_delete, 17](#)
- [attachment_delete\(\), 24, 26, 33, 260, 262, 264, 265, 267, 269, 271, 273, 275, 277, 278, 280, 282, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [attachment_get, 19, 180](#)
- [attachment_get\(\), 22, 30, 142, 180, 182, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257](#)
- [attachment_link, 21](#)
- [attachment_link\(\), 21, 30, 142, 180, 182, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257](#)
- [attachment_list, 23, 159](#)
- [attachment_list\(\), 18, 26, 33, 260, 262, 264, 265, 267, 269, 271, 273, 275, 277, 278, 280, 282, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [attachment_upload, 25](#)
- [attachment_upload\(\), 18, 24, 33, 260, 262, 264, 265, 267, 269, 271, 273, 275, 277, 278, 280, 282, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [attachment_url, 180](#)
- [attachment_url\(\), 21, 22, 30, 142, 180, 182, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257](#)
- [audit_get, 27](#)
- [audit_get\(\), 30](#)
- [config_public, 29](#)
- [config_public\(\), 28](#)
- [drop_null_coords, 30](#)
- [drop_null_coords\(\), 21, 22, 142, 180, 182, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257](#)
- [encryption_key_list, 31](#)
- [encryption_key_list\(\), 18, 24, 26, 260, 262, 264, 265, 267, 269, 271, 273, 275, 277, 278, 280, 282, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [entity_audits, 33](#)
- [entity_audits\(\), 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_bulk_delete, 36](#)
- [entity_bulk_delete\(\), 35, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_bulk_restore, 38](#)
- [entity_bulk_restore\(\), 35, 37, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_changes, 40](#)
- [entity_changes\(\), 35, 37, 39, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_create, 42](#)
- [entity_create\(\), 35, 37, 39, 41, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_creators, 46](#)
- [entity_creators\(\), 35, 37, 39, 41, 45, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_delete, 48](#)
- [entity_delete\(\), 35, 37, 39, 41, 45, 47, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_detail, 50](#)
- [entity_detail\(\), 35, 37, 39, 41, 45, 47, 49, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_geodata, 53](#)
- [entity_geodata\(\), 35, 37, 39, 41, 45, 47, 49, 52, 56, 58, 60, 63, 66, 68, 70, 72, 75,](#)

- [77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_geojson, 55](#)
- [entity_geojson\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 58, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_list, 57](#)
- [entity_list\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 60, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_restore, 59](#)
- [entity_restore\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 63, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_update, 61](#)
- [entity_update\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 66, 68, 70, 72, 75, 77, 79, 81, 84, 86, 198, 200, 202](#)
- [entity_versions, 64](#)
- [entity_versions\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 68, 70, 72, 75, 77, 80, 81, 84, 86, 198, 200, 202](#)
- [entitylist_create, 66](#)
- [entitylist_create\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 70, 72, 75, 77, 80, 81, 84, 86, 198, 200, 202](#)
- [entitylist_delete, 69](#)
- [entitylist_delete\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 72, 75, 77, 80, 81, 84, 86, 198, 200, 202](#)
- [entitylist_detail, 71](#)
- [entitylist_detail\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 75, 77, 80, 81, 84, 86, 198, 200, 202](#)
- [entitylist_download, 73](#)
- [entitylist_download\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 77, 80, 81, 84, 86, 198, 200, 202](#)
- [entitylist_list, 76](#)
- [entitylist_list\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 80, 81, 84, 86, 198, 200, 202](#)
- [entitylist_property_create, 78](#)
- [entitylist_property_create\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 78, 81, 84, 86, 198, 200, 202](#)
- [entitylist_property_delete, 80](#)
- [entitylist_property_delete\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 78, 80, 84, 86, 198, 200, 202](#)
- [entitylist_trash_download, 82](#)
- [entitylist_trash_download\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 78, 80, 81, 86, 198, 200, 202](#)
- [entitylist_update, 84](#)
- [entitylist_update\(\), 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 78, 80, 81, 84, 198, 200, 202](#)
- [form_assignment_actors, 86](#)
- [form_assignment_actors\(\), 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_assignment_grant, 89](#)
- [form_assignment_grant\(\), 88, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_assignment_list, 90](#)
- [form_assignment_list\(\), 7, 9, 10, 12, 14, 15, 17, 218, 241, 243, 327, 328, 330, 332, 333, 335, 336, 338, 339, 341, 342](#)
- [form_assignment_revoke, 92](#)
- [form_assignment_revoke\(\), 88, 90, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_assignment_role_list, 94](#)
- [form_assignment_role_list\(\), 215, 216, 219, 221, 222, 224, 226, 227, 229, 231](#)
- [form_attachment_download, 95](#)
- [form_attachment_download\(\), 88, 90, 93, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136,](#)

- [140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_attachment_list, 97](#)
- [form_attachment_list\(\), 88, 90, 93, 97, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_create, 99](#)
- [form_create\(\), 88, 90, 93, 97, 98, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_dataset_diff, 101](#)
- [form_dataset_diff\(\), 88, 90, 93, 97, 98, 100, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_delete, 103](#)
- [form_delete\(\), 88, 90, 93, 97, 98, 100, 102, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_detail, 105, 163](#)
- [form_detail\(\), 88, 90, 93, 97, 98, 100, 102, 104, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_draft_attachment_delete, 106](#)
- [form_draft_attachment_delete\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_draft_attachment_download, 108](#)
- [form_draft_attachment_download\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_draft_attachment_link, 110](#)
- [form_draft_attachment_link\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_draft_attachment_list, 112](#)
- [form_draft_attachment_list\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_draft_attachment_upload, 114](#)
- [form_draft_attachment_upload\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 158, 232, 234, 236, 238, 240](#)
- [form_draft_create, 116](#)
- [form_draft_create\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_draft_dataset_diff, 118](#)
- [form_draft_dataset_diff\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_draft_delete, 119](#)
- [form_draft_delete\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157,](#)

- [159, 232, 234, 236, 238, 240](#)
- [form_draft_detail, 121](#)
- [form_draft_detail\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_draft_publish, 123](#)
- [form_draft_publish\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_draft_xlsx, 125](#)
- [form_draft_xlsx\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_draft_xml, 126](#)
- [form_draft_xml\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_link, 128](#)
- [form_link\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_list, 130, 163](#)
- [form_list\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_restore, 132](#)
- [form_restore\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_schema, 133, 134, 135, 138, 139, 141, 174, 175, 211, 253, 255, 257](#)
- [form_schema\(\), 88, 90, 93, 97, 98, 100, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_schema_ext, 137, 139](#)
- [form_schema_ext\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 143, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_schema_parse, 134, 138, 141, 141, 174](#)
- [form_schema_parse\(\), 21, 22, 30, 180, 182, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257](#)
- [form_update, 142](#)
- [form_update\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 145, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_version_attachment_download, 144](#)
- [form_version_attachment_download\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 147, 149, 151, 153, 155, 157, 159, 232, 234, 236, 238, 240](#)
- [form_version_attachment_list, 146](#)
- [form_version_attachment_list\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 149, 151, 153, 155, 157, 159, 233, 234, 236, 238, 240](#)
- [form_version_detail, 148](#)
- [form_version_detail\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124,](#)

- [126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 151, 153, 155, 157, 159, 233, 234, 236, 238, 240](#)
- [form_version_list, 150](#)
- [form_version_list\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 153, 155, 157, 159, 233, 234, 236, 238, 240](#)
- [form_version_xlsx, 152](#)
- [form_version_xlsx\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 233, 234, 236, 238, 240](#)
- [form_version_xml, 154](#)
- [form_version_xml\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 159, 233, 234, 236, 238, 240](#)
- [form_xlsx, 155](#)
- [form_xlsx\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 159, 233, 234, 236, 238, 240](#)
- [form_xml, 139, 157, 165](#)
- [form_xml\(\), 88, 90, 93, 97, 98, 101, 102, 104, 106, 108, 109, 111, 113, 115, 117, 119, 121, 122, 124, 126, 128, 129, 131, 133, 136, 140, 143, 145, 147, 149, 151, 153, 155, 157, 233, 234, 236, 238, 240](#)
- [fq_attachments, 159, 160–179](#)
- [fq_data, 160, 160–179](#)
- [fq_data_strata, 160, 161, 162–179](#)
- [fq_data_taxa, 160, 161, 162, 163–179](#)
- [fq_form_detail, 160–162, 163, 164–179](#)
- [fq_form_list, 160–162, 163, 163–179](#)
- [fq_form_schema, 160–163, 164, 164–179](#)
- [fq_form_xml, 160–164, 165, 166–179](#)
- [fq_meta, 160–164, 165, 165–179](#)
- [fq_project_detail, 160–165, 166, 166–179](#)
- [fq_project_list, 160–166, 167, 168–179](#)
- [fq_raw, 160–166, 167, 167, 169–179](#)
- [fq_raw_strata, 160–167, 168, 168, 170–179](#)
- [fq_raw_taxa, 160–168, 169, 169–179](#)
- [fq_submission_list, 160–169, 170, 170–179](#)
- [fq_submissions, 160–169, 170, 170–179](#)
- [fq_svc, 160–170, 171, 171–179](#)
- [fq_zip_data, 160–171, 172, 173–179](#)
- [fq_zip_strata, 160–171, 172, 172–179](#)
- [fq_zip_taxa, 160–172, 173, 173–179](#)
- [fs_v7, 160–172, 173, 173–179](#)
- [fs_v7_raw, 160–173, 174, 174–179](#)
- [geo_fs, 160–174, 175, 176–179](#)
- [geo_gj, 160–174, 175, 175–179](#)
- [geo_gj88, 160–175, 176, 176–179](#)
- [geo_gj_raw, 160–175, 176, 176–179](#)
- [geo_wkt, 160–176, 177, 177–179](#)
- [geo_wkt88, 160–177, 178, 178](#)
- [geo_wkt_raw, 160–177, 178, 179](#)
- [get_default_fid, 18, 20, 23, 25, 32, 87, 89, 92, 96, 98, 102, 103, 105, 107, 109, 110, 112, 114, 116, 118, 120, 122, 123, 125, 127, 134, 138, 143, 144, 146, 148, 150, 152, 154, 156, 158, 181, 183, 185, 188, 203, 205, 207, 232, 233, 235, 237, 239, 248, 259, 261, 263, 265, 267, 269, 270, 272, 274, 276, 278, 280, 282, 284, 285, 287, 289, 291, 294, 296, 298, 300, 302, 304, 306, 308, 310, 312, 314, 316, 318, 320, 322, 324](#)
- [get_default_fid\(ru_settings\), 247](#)
- [get_default_odkc_version, 12, 13, 16, 34, 37, 39, 41, 44, 47, 49, 51, 54, 56, 58, 60, 62, 65, 67, 70, 72, 74, 77, 79, 81, 83, 85, 87, 95, 135, 139, 192, 194, 196, 199, 202, 208, 214, 216, 219, 225, 248, 255, 257, 294](#)
- [get_default_odkc_version\(ru_settings\), 247](#)
- [get_default_orders\(ru_settings\), 247](#)
- [get_default_pid, 7, 8, 10, 18, 20, 23, 25, 32, 34, 36, 38, 40, 43, 46, 48, 50, 53, 55, 57, 59, 61, 64, 67, 69, 71, 73, 76, 78, 80, 82, 85, 87, 89, 91, 92, 94, 96, 98, 99, 102, 103, 105, 107, 109, 110, 112, 114, 116, 118, 120, 122, 123,](#)

- [125, 127, 130, 132, 134, 138, 142, 144, 146, 148, 150, 152, 154, 156, 158, 181, 183, 185, 187, 196, 199, 201, 203, 205, 207, 214, 216, 217, 219, 222, 223, 225, 228, 230, 231, 233, 235, 237, 239, 248, 259, 261, 263, 265, 266, 269, 270, 272, 274, 276, 278, 280, 282, 283, 285, 287, 289, 291, 294, 296, 298, 300, 302, 304, 306, 308, 310, 312, 314, 316, 318, 320, 322, 324, 333, 334](#)
- [get_default_pid\(ru_settings\), 247](#)
- [get_default_pp, 248](#)
- [get_default_pp\(ru_settings\), 247](#)
- [get_default_pw, 7, 8, 10, 11, 13, 15, 16, 18, 20, 24, 26, 27, 32, 34, 37, 39, 41, 43, 47, 49, 51, 53, 55, 57, 60, 62, 65, 67, 69, 71, 74, 77, 79, 81, 83, 85, 87, 89, 91, 93, 94, 96, 98, 100, 102, 104, 105, 107, 109, 111, 112, 115, 117, 118, 120, 122, 124, 125, 127, 129, 130, 132, 134, 139, 143, 145, 147, 149, 150, 152, 154, 156, 158, 180, 182, 184, 186, 188, 196, 199, 201, 203, 205, 208, 214, 216, 217, 219, 220, 222, 223, 225, 227, 228, 230, 232, 234, 236, 237, 239, 241, 242, 248, 260, 261, 263, 265, 267, 269, 271, 272, 274, 276, 278, 280, 282, 284, 286, 287, 289, 291, 294, 297, 299, 300, 302, 305, 306, 308, 310, 313, 315, 317, 319, 320, 322, 324, 326, 328, 329, 331, 333, 334, 336, 337, 339, 340, 342](#)
- [get_default_pw\(ru_settings\), 247](#)
- [get_default_tz, 248](#)
- [get_default_tz\(ru_settings\), 247](#)
- [get_default_un, 7, 8, 10, 11, 13, 15, 16, 18, 20, 23, 26, 27, 32, 34, 37, 39, 41, 43, 47, 49, 51, 53, 55, 57, 60, 62, 65, 67, 69, 71, 73, 77, 79, 81, 83, 85, 87, 89, 91, 93, 94, 96, 98, 100, 102, 104, 105, 107, 109, 111, 112, 114, 117, 118, 120, 122, 124, 125, 127, 129, 130, 132, 134, 139, 143, 145, 147, 148, 150, 152, 154, 156, 158, 180, 182, 183, 186, 188, 196, 199, 201, 203, 205, 208, 214, 216, 217, 219, 220, 222, 223, 225, 227, 228, 230, 232, 234, 235, 237, 239, 241, 242, 248, 259, 261, 263, 265, 267, 269, 271, 272, 274, 276, 278, 280, 282, 284, 285, 287, 289, 291, 294, 297, 299, 300, 302, 304, 306, 308, 310, 313, 315, 317, 318, 320, 322, 324, 326, 327, 329, 331, 333, 334, 336, 337, 339, 340, 342](#)
- [get_default_un\(ru_settings\), 247](#)
- [get_default_url, 7, 8, 10, 11, 13, 15, 16, 18, 20, 23, 25, 27, 29, 32, 34, 36, 38, 40, 43, 47, 49, 51, 53, 55, 57, 59, 62, 65, 67, 69, 71, 73, 77, 79, 81, 83, 85, 87, 89, 91, 93, 94, 96, 98, 100, 102, 103, 105, 107, 109, 111, 112, 114, 116, 118, 120, 122, 124, 125, 127, 129, 130, 132, 134, 139, 143, 145, 147, 148, 150, 152, 154, 156, 158, 180, 181, 183, 185, 188, 196, 199, 201, 203, 205, 208, 214, 216, 217, 219, 220, 222, 223, 225, 226, 228, 230, 232, 234, 235, 237, 239, 241, 242, 248, 259, 261, 263, 265, 267, 269, 271, 272, 274, 276, 278, 280, 282, 284, 285, 287, 289, 291, 294, 297, 299, 300, 302, 304, 306, 308, 310, 313, 315, 317, 318, 320, 322, 324, 326, 327, 329, 331, 333, 334, 336, 337, 339, 340, 342](#)
- [get_default_url\(ru_settings\), 247](#)
- [get_one_attachment, 179](#)
- [get_one_attachment\(\), 21, 22, 30, 142, 182, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257](#)
- [get_one_submission, 181, 300](#)
- [get_one_submission\(\), 21, 22, 30, 142, 180, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257](#)
- [get_one_submission_att_list, 183](#)
- [get_one_submission_att_list\(\), 21, 22, 30, 142, 180, 182, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257](#)
- [get_one_submission_audit, 185, 260](#)
- [get_one_submission_audit\(\), 21, 22, 30, 142, 180, 182, 184, 188, 190, 191, 193, 194, 211, 253, 255, 257](#)
- [get_retries, 248](#)

- `get_retries(ru_settings)`, 247
- `get_ru_verbose`, 248
- `get_ru_verbose(ru_settings)`, 247
- `get_test_fid`, 248
- `get_test_fid(ru_settings)`, 247
- `get_test_fid_att`, 248
- `get_test_fid_att(ru_settings)`, 247
- `get_test_fid_gap`, 248
- `get_test_fid_gap(ru_settings)`, 247
- `get_test_fid_wkt`, 248
- `get_test_fid_wkt(ru_settings)`, 247
- `get_test_fid_zip`, 248
- `get_test_fid_zip(ru_settings)`, 247
- `get_test_odkc_version`, 248
- `get_test_odkc_version(ru_settings)`, 247
- `get_test_pid`, 248
- `get_test_pid(ru_settings)`, 247
- `get_test_pp`, 248
- `get_test_pp(ru_settings)`, 247
- `get_test_pw`, 248
- `get_test_pw(ru_settings)`, 247
- `get_test_un`, 248
- `get_test_un(ru_settings)`, 247
- `get_test_url`, 248
- `get_test_url(ru_settings)`, 247

- `handle_ru_attachments`, 20, 141, 187, 209
- `handle_ru_attachments()`, 21, 22, 30, 142, 180, 182, 184, 186, 190, 191, 193, 194, 211, 253, 255, 257
- `handle_ru_datetimes`, 141, 189, 209
- `handle_ru_datetimes()`, 21, 22, 30, 142, 180, 182, 184, 186, 188, 191, 193, 194, 211, 253, 255, 257
- `handle_ru_geopoints`, 141, 190, 209, 253, 255, 257
- `handle_ru_geopoints()`, 21, 22, 30, 142, 180, 182, 184, 186, 188, 190, 193, 194, 211, 253, 255, 257
- `handle_ru_geoshapes`, 141, 192, 209
- `handle_ru_geoshapes()`, 21, 22, 30, 142, 180, 182, 184, 186, 188, 190, 191, 194, 211, 253, 255, 257
- `handle_ru_geotraces`, 141, 193, 209
- `handle_ru_geotraces()`, 21, 22, 30, 142, 180, 182, 184, 186, 188, 190, 191, 193, 211, 253, 255, 257

- `isodt_to_local()`, 21, 22, 30, 142, 180, 182, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257

- `make_clean_names`, 135, 140

- `odata_entitylist_data_get`, 195
- `odata_entitylist_data_get()`, 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 78, 80, 81, 84, 86, 200, 202
- `odata_entitylist_metadata_get`, 198
- `odata_entitylist_metadata_get()`, 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 78, 80, 81, 84, 86, 198, 202
- `odata_entitylist_service_get`, 201
- `odata_entitylist_service_get()`, 35, 37, 39, 41, 45, 47, 49, 52, 54, 56, 58, 60, 63, 66, 68, 70, 72, 75, 78, 80, 81, 84, 86, 198, 200
- `odata_metadata_get`, 203
- `odata_metadata_get()`, 205, 209
- `odata_service_get`, 204, 206
- `odata_service_get()`, 204, 209
- `odata_submission_get`, 135, 140, 160–162, 175–179, 206, 208–210, 249, 294
- `odata_submission_get()`, 204, 205
- `odata_submission_rectangle`, 135, 140, 209, 210
- `odata_submission_rectangle()`, 21, 22, 30, 142, 180, 182, 184, 186, 188, 190, 191, 193, 194, 253, 255, 257
- `odata_svc_parse`, 212
- `odata_svc_parse()`, 213, 248, 251–253

- `parse_odkc_version`, 212
- `parse_odkc_version()`, 212, 248, 251–253
- `predict_ruodk_name()`, 21, 22, 30, 142, 180, 182, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257
- `prepend_uuid()`, 21, 22, 30, 142, 180, 182, 184, 186, 188, 190, 191, 193, 194, 211, 253, 255, 257
- `project_assignment_actors`, 213
- `project_assignment_actors()`, 95, 216, 219, 221, 222, 224, 226, 227, 229, 231
- `project_assignment_grant`, 215

- `project_assignment_grant()`, [95](#), [215](#), [219](#), [221](#), [222](#), [224](#), [226](#), [227](#), [229](#), [231](#)
- `project_assignment_list`, [217](#)
- `project_assignment_list()`, [7](#), [9](#), [10](#), [12](#), [14](#), [15](#), [17](#), [91](#), [241](#), [243](#), [327](#), [328](#), [330](#), [332](#), [333](#), [335](#), [336](#), [338](#), [339](#), [341](#), [342](#)
- `project_assignment_revoke`, [218](#)
- `project_assignment_revoke()`, [95](#), [215](#), [216](#), [221](#), [222](#), [224](#), [226](#), [227](#), [229](#), [231](#)
- `project_create`, [220](#)
- `project_create()`, [95](#), [215](#), [216](#), [219](#), [222](#), [224](#), [226](#), [227](#), [229](#), [231](#)
- `project_delete`, [221](#)
- `project_delete()`, [95](#), [215](#), [216](#), [219](#), [221](#), [224](#), [226](#), [227](#), [229](#), [231](#)
- `project_detail`, [166](#), [223](#), [223](#)
- `project_detail()`, [95](#), [215](#), [216](#), [219](#), [221](#), [222](#), [226](#), [227](#), [229](#), [231](#)
- `project_enable_encryption`, [224](#)
- `project_enable_encryption()`, [95](#), [215](#), [216](#), [219](#), [221](#), [222](#), [224](#), [227](#), [229](#), [231](#)
- `project_list`, [167](#), [226](#), [226](#)
- `project_list()`, [95](#), [215](#), [216](#), [219](#), [221](#), [222](#), [224](#), [226](#), [229](#), [231](#)
- `project_replace`, [228](#)
- `project_replace()`, [95](#), [215](#), [216](#), [219](#), [221](#), [222](#), [224](#), [226](#), [227](#), [231](#)
- `project_update`, [229](#)
- `project_update()`, [95](#), [215](#), [216](#), [219](#), [221](#), [222](#), [224](#), [226](#), [227](#), [229](#)
- `public_link_create`, [231](#)
- `public_link_create()`, [88](#), [90](#), [93](#), [97](#), [98](#), [101](#), [102](#), [104](#), [106](#), [108](#), [109](#), [111](#), [113](#), [115](#), [117](#), [119](#), [121](#), [122](#), [124](#), [126](#), [128](#), [129](#), [131](#), [133](#), [136](#), [140](#), [143](#), [145](#), [147](#), [149](#), [151](#), [153](#), [155](#), [157](#), [159](#), [234](#), [236](#), [238](#), [240](#)
- `public_link_delete`, [233](#)
- `public_link_delete()`, [88](#), [90](#), [93](#), [97](#), [98](#), [101](#), [102](#), [104](#), [106](#), [108](#), [109](#), [111](#), [113](#), [115](#), [117](#), [119](#), [121](#), [122](#), [124](#), [126](#), [128](#), [129](#), [131](#), [133](#), [136](#), [140](#), [143](#), [145](#), [147](#), [149](#), [151](#), [153](#), [155](#), [157](#), [159](#), [233](#), [236](#), [238](#), [240](#)
- `public_link_detail`, [235](#)
- `public_link_detail()`, [88](#), [90](#), [93](#), [97](#), [98](#), [101](#), [102](#), [104](#), [106](#), [108](#), [109](#), [111](#), [113](#), [115](#), [117](#), [119](#), [121](#), [122](#), [124](#), [126](#), [128](#), [129](#), [131](#), [133](#), [136](#), [140](#), [143](#), [145](#), [147](#), [149](#), [151](#), [153](#), [155](#), [157](#), [159](#), [233](#), [234](#), [238](#), [240](#)
- `public_link_list`, [237](#)
- `public_link_list()`, [88](#), [90](#), [93](#), [97](#), [98](#), [101](#), [102](#), [104](#), [106](#), [108](#), [109](#), [111](#), [113](#), [115](#), [117](#), [119](#), [121](#), [122](#), [124](#), [126](#), [128](#), [129](#), [131](#), [133](#), [136](#), [140](#), [143](#), [145](#), [147](#), [149](#), [151](#), [153](#), [155](#), [157](#), [159](#), [233](#), [234](#), [236](#), [240](#)
- `public_link_update`, [238](#)
- `public_link_update()`, [88](#), [90](#), [93](#), [97](#), [98](#), [101](#), [102](#), [104](#), [106](#), [108](#), [109](#), [111](#), [113](#), [115](#), [117](#), [119](#), [121](#), [122](#), [124](#), [126](#), [128](#), [129](#), [131](#), [133](#), [136](#), [140](#), [143](#), [145](#), [147](#), [149](#), [151](#), [153](#), [155](#), [157](#), [159](#), [233](#), [234](#), [236](#), [238](#)
- `RETRY`, [7](#), [8](#), [10](#), [12](#), [13](#), [15](#), [16](#), [18](#), [20](#), [24](#), [26](#), [27](#), [29](#), [32](#), [34](#), [37](#), [39](#), [41](#), [44](#), [47](#), [49](#), [51](#), [53](#), [55](#), [57](#), [60](#), [62](#), [65](#), [67](#), [69](#), [71](#), [74](#), [77](#), [79](#), [81](#), [83](#), [85](#), [87](#), [89](#), [91](#), [93](#), [94](#), [96](#), [98](#), [100](#), [102](#), [104](#), [105](#), [107](#), [109](#), [111](#), [113](#), [115](#), [117](#), [118](#), [120](#), [122](#), [124](#), [126](#), [127](#), [129](#), [130](#), [132](#), [135](#), [139](#), [143](#), [145](#), [147](#), [149](#), [150](#), [153](#), [154](#), [156](#), [158](#), [180](#), [182](#), [184](#), [186](#), [188](#), [196](#), [199](#), [201](#), [204](#), [205](#), [208](#), [214](#), [216](#), [217](#), [219](#), [220](#), [222](#), [223](#), [225](#), [227](#), [228](#), [230](#), [232](#), [234](#), [236](#), [237](#), [239](#), [241](#), [242](#), [249](#), [260](#), [261](#), [263](#), [265](#), [267](#), [269](#), [271](#), [272](#), [274](#), [276](#), [278](#), [280](#), [282](#), [284](#), [286](#), [287](#), [289](#), [291](#), [294](#), [297](#), [299](#), [301](#), [302](#), [305](#), [306](#), [308](#), [310](#), [313](#), [315](#), [317](#), [319](#), [321](#), [323](#), [324](#), [326](#), [328](#), [329](#), [331](#), [333](#), [334](#), [336](#), [337](#), [339](#), [340](#), [342](#)
- `role_detail`, [240](#)
- `role_detail()`, [7](#), [9](#), [10](#), [12](#), [14](#), [15](#), [17](#), [91](#), [218](#), [243](#), [327](#), [328](#), [330](#), [332](#), [333](#), [335](#), [336](#), [338](#), [339](#), [341](#), [342](#)
- `role_list`, [242](#)
- `role_list()`, [7](#), [9](#), [10](#), [12](#), [14](#), [15](#), [17](#), [91](#), [218](#), [241](#), [327](#), [328](#), [330](#), [332](#), [333](#), [335](#), [336](#), [338](#), [339](#), [341](#), [342](#)

- `ru_http_request()`, [21](#), [22](#), [30](#), [142](#), [180](#),
[182](#), [184](#), [186](#), [188](#), [190](#), [191](#), [193](#),
[194](#), [211](#), [253](#), [255](#), [257](#)
- `ru_msg_abort`, [243](#)
- `ru_msg_abort()`, [244–246](#)
- `ru_msg_info`, [244](#)
- `ru_msg_info()`, [244–246](#)
- `ru_msg_noop`, [245](#)
- `ru_msg_noop()`, [244](#), [246](#)
- `ru_msg_success`, [245](#)
- `ru_msg_success()`, [244–246](#)
- `ru_msg_warn`, [246](#)
- `ru_msg_warn()`, [244–246](#)
- `ru_settings`, [247](#), [248](#)
- `ru_settings()`, [212](#), [213](#), [251–253](#)
- `ru_setup`, [248](#), [248–250](#)
- `ru_setup()`, [212](#), [213](#), [248](#), [252](#), [253](#)
- `ruODK`, [248](#), [250](#), [251](#)
-
- `semver_gt`, [251](#)
- `semver_gt()`, [212](#), [213](#), [248](#), [251](#), [253](#)
- `semver_lt`, [252](#)
- `semver_lt()`, [212](#), [213](#), [248](#), [251](#), [252](#)
- `split_geopoint`, [191](#), [253](#)
- `split_geopoint()`, [21](#), [22](#), [30](#), [142](#), [180](#), [182](#),
[184](#), [186](#), [188](#), [190](#), [191](#), [193](#), [194](#),
[211](#), [255](#), [257](#)
- `split_geoshape`, [30](#), [193](#), [208](#), [254](#)
- `split_geoshape()`, [21](#), [22](#), [30](#), [142](#), [180](#), [182](#),
[184](#), [186](#), [188](#), [190](#), [191](#), [193](#), [194](#),
[211](#), [253](#), [257](#)
- `split_geotrace`, [30](#), [194](#), [208](#), [256](#)
- `split_geotrace()`, [21](#), [22](#), [30](#), [142](#), [180](#), [182](#),
[184](#), [186](#), [188](#), [190](#), [191](#), [193](#), [194](#),
[211](#), [253](#), [255](#)
- `strip_uuid()`, [21](#), [22](#), [30](#), [142](#), [180](#), [182](#), [184](#),
[186](#), [188](#), [190](#), [191](#), [193](#), [194](#), [211](#),
[253](#), [255](#), [257](#)
- `submission_audit_get`, [259](#)
- `submission_audit_get()`, [18](#), [24](#), [26](#), [33](#),
[262](#), [264](#), [265](#), [267](#), [269](#), [271](#), [273](#),
[275](#), [277](#), [278](#), [280](#), [282](#), [284](#), [286](#),
[288](#), [290](#), [292](#), [295](#), [297](#), [299](#), [301](#),
[303](#), [305](#), [307](#), [309](#), [311](#), [313](#), [315](#),
[317](#), [319](#), [321](#), [323](#), [325](#)
- `submission_changes`, [261](#)
- `submission_changes()`, [18](#), [24](#), [26](#), [33](#), [260](#),
[264](#), [265](#), [267](#), [269](#), [271](#), [273](#), [275](#),
[277](#), [278](#), [280](#), [282](#), [284](#), [286](#), [288](#),
[290](#), [292](#), [295](#), [297](#), [299](#), [301](#), [303](#),
[305](#), [307](#), [309](#), [311](#), [313](#), [315](#), [317](#),
[319](#), [321](#), [323](#), [325](#)
- `submission_create`, [266](#)
- `submission_create()`, [18](#), [24](#), [26](#), [33](#), [260](#),
[262](#), [264](#), [266](#), [269](#), [271](#), [273](#), [275](#),
[277](#), [279](#), [280](#), [282](#), [284](#), [286](#), [288](#),
[290](#), [292](#), [295](#), [297](#), [299](#), [301](#), [303](#),
[305](#), [307](#), [309](#), [311](#), [313](#), [315](#), [317](#),
[319](#), [321](#), [323](#), [325](#)
- `submission_csv`, [268](#)
- `submission_csv()`, [18](#), [24](#), [26](#), [33](#), [260](#), [262](#),
[264](#), [266](#), [267](#), [271](#), [273](#), [275](#), [277](#),
[279](#), [280](#), [282](#), [284](#), [286](#), [288](#), [290](#),
[292](#), [295](#), [297](#), [299](#), [301](#), [303](#), [305](#),
[307](#), [309](#), [311](#), [313](#), [315](#), [317](#), [319](#),
[321](#), [323](#), [325](#)
- `submission_delete`, [270](#)
- `submission_delete()`, [18](#), [24](#), [26](#), [33](#), [260](#),
[262](#), [264](#), [266](#), [267](#), [269](#), [273](#), [275](#),
[277](#), [279](#), [280](#), [282](#), [284](#), [286](#), [288](#),
[290](#), [292](#), [295](#), [297](#), [299](#), [301](#), [303](#),
[305](#), [307](#), [309](#), [311](#), [313](#), [315](#), [317](#),
[319](#), [321](#), [323](#), [325](#)
- `submission_detail`, [272](#)
- `submission_detail()`, [18](#), [24](#), [26](#), [33](#), [260](#),
[262](#), [264](#), [266](#), [267](#), [269](#), [271](#), [275](#),
[277](#), [279](#), [280](#), [282](#), [284](#), [286](#), [288](#),
[290](#), [292](#), [295](#), [297](#), [299](#), [301](#), [303](#),
[305](#), [307](#), [309](#), [311](#), [313](#), [315](#), [317](#),
[319](#), [321](#), [323](#), [325](#)
- `submission_draft_attachment_delete`,
[273](#)
- `submission_draft_attachment_delete()`,

- [18, 24, 26, 33, 260, 262, 264, 266, 267, 269, 271, 273, 277, 279, 280, 282, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_draft_attachment_download, 275](#)
- [submission_draft_attachment_download\(\), 18, 24, 26, 33, 260, 262, 264, 266, 267, 269, 271, 273, 275, 279, 280, 282, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_draft_attachment_list, 277](#)
- [submission_draft_attachment_list\(\), 18, 24, 26, 33, 260, 262, 264, 266, 267, 269, 271, 273, 275, 277, 280, 282, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_draft_attachment_upload, 279](#)
- [submission_draft_attachment_upload\(\), 18, 24, 26, 33, 260, 262, 264, 266, 267, 269, 271, 273, 275, 277, 279, 282, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_draft_create, 281](#)
- [submission_draft_create\(\), 19, 24, 26, 33, 260, 262, 264, 266, 267, 269, 271, 273, 275, 277, 279, 280, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_draft_export, 283](#)
- [submission_draft_export\(\), 19, 24, 26, 33, 260, 262, 264, 266, 267, 269, 271, 273, 275, 277, 279, 280, 282, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_draft_get, 285](#)
- [submission_draft_get\(\), 19, 24, 26, 33, 260, 262, 264, 266, 268, 269, 271, 273, 275, 277, 279, 281, 282, 284, 286, 288, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_draft_keys, 287](#)
- [submission_draft_keys\(\), 19, 24, 26, 33, 260, 262, 264, 266, 268, 269, 271, 273, 275, 277, 279, 281, 282, 284, 286, 290, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_draft_list, 288](#)
- [submission_draft_list\(\), 19, 24, 26, 33, 260, 262, 264, 266, 268, 269, 271, 273, 275, 277, 279, 281, 282, 284, 286, 288, 292, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_edit, 290](#)
- [submission_edit\(\), 19, 24, 26, 33, 260, 262, 264, 266, 268, 269, 271, 273, 275, 277, 279, 281, 282, 284, 286, 288, 290, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_export, 20, 22, 172, 173, 209, 292](#)
- [submission_export\(\), 19, 24, 26, 33, 260, 262, 264, 266, 268, 269, 271, 273, 275, 277, 279, 281, 282, 284, 286, 288, 290, 292, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_geodata, 296](#)
- [submission_geodata\(\), 19, 24, 26, 33, 260, 262, 264, 266, 268, 269, 271, 273, 275, 277, 279, 281, 282, 284, 286, 288, 290, 292, 295, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_geojson, 298](#)
- [submission_geojson\(\), 19, 24, 26, 33, 260, 262, 264, 266, 268, 269, 271, 273, 275, 277, 279, 281, 282, 284, 286, 288, 290, 292, 295, 297, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325](#)
- [submission_get, 171, 300](#)

- `submission_get()`, [19](#), [24](#), [26](#), [33](#), [260](#), [262](#),
[264](#), [266](#), [268](#), [269](#), [271](#), [273](#), [275](#),
[277](#), [279](#), [281](#), [282](#), [284](#), [286](#), [288](#),
[290](#), [292](#), [295](#), [297](#), [299](#), [303](#), [305](#),
[307](#), [309](#), [311](#), [313](#), [315](#), [317](#), [319](#),
[321](#), [323](#), [325](#)
- `submission_list`, [18](#), [23](#), [25](#), [170](#), [171](#), [181](#),
[183](#), [185](#), [259–261](#), [263](#), [265](#), [270](#),
[272](#), [274](#), [276](#), [278](#), [280](#), [285](#), [291](#),
[294](#), [298](#), [300](#), [302](#), [304](#), [306](#), [310](#),
[312](#), [314](#), [316](#), [318](#), [320](#), [322](#), [324](#)
- `submission_list()`, [19](#), [24](#), [26](#), [33](#), [260](#), [262](#),
[264](#), [266](#), [268](#), [270](#), [271](#), [273](#), [275](#),
[277](#), [279](#), [281](#), [282](#), [284](#), [286](#), [288](#),
[290](#), [292](#), [295](#), [297](#), [299](#), [301](#), [305](#),
[307](#), [309](#), [311](#), [313](#), [315](#), [317](#), [319](#),
[321](#), [323](#), [325](#)
- `submission_restore`, [304](#)
- `submission_restore()`, [19](#), [24](#), [26](#), [33](#), [260](#),
[262](#), [264](#), [266](#), [268](#), [270](#), [271](#), [273](#),
[275](#), [277](#), [279](#), [281](#), [282](#), [284](#), [286](#),
[288](#), [290](#), [292](#), [295](#), [297](#), [299](#), [301](#),
[303](#), [307](#), [309](#), [311](#), [313](#), [315](#), [317](#),
[319](#), [321](#), [323](#), [325](#)
- `submission_review`, [306](#)
- `submission_review()`, [19](#), [24](#), [26](#), [33](#), [260](#),
[262](#), [264](#), [266](#), [268](#), [270](#), [271](#), [273](#),
[275](#), [277](#), [279](#), [281](#), [282](#), [284](#), [286](#),
[288](#), [290](#), [292](#), [295](#), [297](#), [299](#), [301](#),
[303](#), [305](#), [309](#), [311](#), [313](#), [315](#), [317](#),
[319](#), [321](#), [323](#), [325](#)
- `submission_submitters`, [308](#)
- `submission_submitters()`, [19](#), [24](#), [26](#), [33](#),
[260](#), [262](#), [264](#), [266](#), [268](#), [270](#), [271](#),
[273](#), [275](#), [277](#), [279](#), [281](#), [282](#), [284](#),
[286](#), [288](#), [290](#), [292](#), [295](#), [297](#), [299](#),
[301](#), [303](#), [305](#), [307](#), [311](#), [313](#), [315](#),
[317](#), [319](#), [321](#), [323](#), [325](#)
- `submission_update`, [309](#)
- `submission_update()`, [19](#), [24](#), [26](#), [33](#), [260](#),
[262](#), [264](#), [266](#), [268](#), [270](#), [271](#), [273](#),
[275](#), [277](#), [279](#), [281](#), [283](#), [284](#), [286](#),
[288](#), [290](#), [292](#), [295](#), [297](#), [299](#), [301](#),
[303](#), [305](#), [307](#), [309](#), [313](#), [315](#), [317](#),
[319](#), [321](#), [323](#), [325](#)
- `submission_version_attachment_delete`,
[312](#)
- `submission_version_attachment_delete()`,
[19](#), [24](#), [26](#), [33](#), [260](#), [262](#), [264](#), [266](#), [268](#), [270](#), [271](#), [273](#),
[275](#), [277](#), [279](#), [281](#), [283](#), [284](#), [286](#), [288](#), [290](#), [292](#), [295](#), [297](#), [299](#),
[301](#), [303](#), [305](#), [307](#), [309](#), [311](#), [313](#), [315](#), [317](#), [319](#), [321](#), [323](#),
[325](#)
- `submission_version_attachment_download`,
[314](#)
- `submission_version_attachment_download()`,
[19](#), [24](#), [26](#), [33](#), [260](#), [262](#), [264](#), [266](#),
[268](#), [270](#), [271](#), [273](#), [275](#), [277](#), [279](#),
[281](#), [283](#), [284](#), [286](#), [288](#), [290](#), [292](#),
[295](#), [297](#), [299](#), [301](#), [303](#), [305](#), [307](#),
[309](#), [311](#), [313](#), [317](#), [319](#), [321](#), [323](#),
[325](#)
- `submission_version_attachment_list`,
[316](#)
- `submission_version_attachment_list()`,
[19](#), [24](#), [26](#), [33](#), [260](#), [262](#), [264](#), [266](#),
[268](#), [270](#), [271](#), [273](#), [275](#), [277](#), [279](#),
[281](#), [283](#), [284](#), [286](#), [288](#), [290](#), [292](#),
[295](#), [297](#), [299](#), [301](#), [303](#), [305](#), [307](#),
[309](#), [311](#), [313](#), [315](#), [319](#), [321](#), [323](#),
[325](#)
- `submission_version_detail`, [318](#)
- `submission_version_detail()`, [19](#), [24](#), [26](#),
[33](#), [260](#), [262](#), [264](#), [266](#), [268](#), [270](#),
[271](#), [273](#), [275](#), [277](#), [279](#), [281](#), [283](#),
[284](#), [286](#), [288](#), [290](#), [292](#), [295](#), [297](#),
[299](#), [301](#), [303](#), [305](#), [307](#), [309](#), [311](#),
[313](#), [315](#), [317](#), [321](#), [323](#), [325](#)
- `submission_version_geojson`, [320](#)
- `submission_version_geojson()`, [19](#), [24](#), [26](#),
[33](#), [260](#), [262](#), [264](#), [266](#), [268](#), [270](#),
[271](#), [273](#), [275](#), [277](#), [279](#), [281](#), [283](#),
[284](#), [286](#), [288](#), [290](#), [292](#), [295](#), [297](#),
[299](#), [301](#), [303](#), [305](#), [307](#), [309](#), [311](#),
[313](#), [315](#), [317](#), [319](#), [323](#), [325](#)
- `submission_version_xml`, [322](#)
- `submission_version_xml()`, [19](#), [24](#), [26](#), [33](#),
[260](#), [262](#), [264](#), [266](#), [268](#), [270](#), [271](#),
[273](#), [275](#), [277](#), [279](#), [281](#), [283](#), [284](#),
[286](#), [288](#), [290](#), [292](#), [295](#), [297](#), [299](#),
[301](#), [303](#), [305](#), [307](#), [309](#), [311](#), [313](#),
[315](#), [317](#), [319](#), [321](#), [325](#)
- `submission_versions`, [324](#)
- `submission_versions()`, [19](#), [24](#), [26](#), [33](#), [260](#),
[262](#), [264](#), [266](#), [268](#), [270](#), [271](#), [273](#),

- 275, 277, 279, 281, 283, 284, 286,
288, 290, 292, 295, 297, 299, 301,
303, 305, 307, 309, 311, 313, 315,
317, 319, 321, 323
- `tidyeval`, 21, 22, 30, 142, 180, 182, 184, 186,
188, 190, 191, 193, 194, 211, 253,
255, 257
- `unnest_all()`, 21, 22, 30, 142, 180, 182, 184,
186, 188, 190, 191, 193, 194, 211,
253, 255, 257
- `user_create`, 326
- `user_create()`, 7, 9, 10, 12, 14, 15, 17, 91,
218, 241, 243, 328, 330, 332, 333,
335, 336, 338, 339, 341, 342
- `user_delete`, 327
- `user_delete()`, 7, 9, 10, 12, 14, 15, 17, 91,
218, 241, 243, 327, 330, 332, 333,
335, 336, 338, 339, 341, 342
- `user_detail`, 329
- `user_detail()`, 7, 9, 10, 12, 14, 15, 17, 91,
218, 241, 243, 327, 328, 332, 333,
335, 336, 338, 339, 341, 342
- `user_list`, 330
- `user_list()`, 7, 9, 10, 12, 14, 15, 17, 91, 218,
241, 243, 327, 328, 330, 333, 335,
336, 338, 339, 341, 342
- `user_preference_project_delete`, 332
- `user_preference_project_delete()`, 7, 9,
10, 12, 14, 15, 17, 91, 218, 241, 243,
327, 328, 330, 332, 335, 336, 338,
339, 341, 342
- `user_preference_project_set`, 334
- `user_preference_project_set()`, 7, 9, 10,
12, 14, 15, 17, 91, 218, 241, 243,
327, 328, 330, 332, 333, 336, 338,
339, 341, 342
- `user_preference_site_delete`, 335
- `user_preference_site_delete()`, 7, 9, 10,
12, 14, 15, 17, 91, 218, 241, 243,
327, 328, 330, 332, 333, 335, 338,
339, 341, 342
- `user_preference_site_set`, 337
- `user_preference_site_set()`, 7, 9, 10, 12,
14, 15, 17, 91, 218, 241, 243, 327,
328, 330, 332, 333, 335, 336, 339,
341, 342
- `user_reset_password`, 338
- `user_reset_password()`, 7, 9, 10, 12, 14, 15,
17, 91, 218, 241, 243, 327, 328, 330,
332, 333, 335, 336, 338, 341, 342
- `user_update`, 340
- `user_update()`, 7, 9, 10, 12, 14, 15, 17, 91,
218, 241, 243, 327, 328, 330, 332,
333, 335, 336, 338, 339, 342
- `user_update_password`, 341
- `user_update_password()`, 7, 9, 10, 12, 14,
15, 17, 91, 218, 241, 243, 327, 328,
330, 332, 333, 335, 336, 338, 339,
341
- `warn_on_schema_drift()`, 204, 205, 209
- `yell_if_error()`, 212, 213, 248, 251–253
- `yell_if_missing()`, 212, 213, 248, 251–253