Package ‘ruODK’

September 24, 2019

Type Package

Title An R Client for the ODK Central API

Version 0.6.5

Description Utilities to access and tidy up data from ODK
Central's API. ODK Central is OpenDataKit's clearinghouse for
digitally captured data <https://docs.opendatakit.org/central-intro/>.
ODK Central's API is documented at
<https://odkcentral.docs.apiary.io/>.

License GPL-3

URL https://dbca-wa.github.io/ruODK/,
https://github.com/dbca-wa/ruODK

BugReports https://github.com/dbca-wa/ruODK/issues

Depends R (>= 3.4)

Imports dplyr (>= 0.8.3),
fs (>= 1.3.1),
glue (>= 1.3.1),
httr (>= 1.4.1),
janitor (>= 1.2.0),
lubridate (>= 1.7.4),
magrittr (>= 1.5),
purrr (>= 0.3.2),
readr (>= 1.3.1),
rlang (>= 0.4.0),
stringr (>= 1.4.0),
tibble (>= 2.1.3),
tidyr (>= 1.0.0),
xml2 (>= 1.2.2)

Suggests covr (>= 3.3.1),
DT (>= 0.8),
ggplot2 (>= 3.2.1),
knitr (>= 1.24),
leaflet (>= 2.0.2),

lifecycle (>= 0.1.0),
listviewer (>= 2.1.0),
rlist (>= 0.4.6.1),
rmarkdown (>= 1.15),
testthat (>= 2.2.1),
usethis (>= 1.5.1)

VignetteBuilder knitr

RdMacros lifecycle

Encoding UTF-8

Language en_AU

LazyData true

RoxygenNote 6.1.1

X-schema.org-applicationCategory Data Access

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

R topics documented:

`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(), verbose = FALSE)
```

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 idential 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")`.

`verbose` Whether to display debug messages or not (default)

Details

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 named by the submission's uuid (minus "uuid:" for Windows compatibility) inside the given ‘local_dir‘. E.g.:

workdir/attachments/xxxxxxxxxx/filename1.jpg

workdir/attachments/xxxxxxxxxx/filename2.jpg

workdir/attachments/xxxxxxxxxx/filename3.jpg

Value

The relative file path for the downloaded attachment(s)

See Also

`https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/'-form-attachments/downloading-a-form-attachment`

`https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/attachments/downloading-an-attachment`

Other odata-api: `attachment_url`, `get_one_attachment`, `odata_metadata_get`, `odata_service_get`, `odata_submission_get`, `odata_submission_parse`, `unnest_all`

`attachment_link` *Prefix attachment columns from CSV export with a local attachment file path.*

Description

Maturing

Usage

```
attachment_link(data_tbl, att_path = "media", att_contains = "photo")
```

Arguments

data_tbl The downloaded submissions from `ruODK::submission_export()` read into a ‘tibble‘ by `readr::read_csv()`.

att_path A local path, default: "media" (as per .csv.zip export). Selected columns of the dataframe (containing attchment filenames) are prefixed with ‘att_path‘, thus turning them into relative paths.

att_contains A shared part of attachment fieldnames, default: "photo". Columns of the dataframe are selected by the presence of ‘att_contains‘.

Value

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

See Also

Other restful-api: attachment_list, audit_get, form_detail, form_list, form_schema_parse, form_schema, form_xml, project_create, project_detail, project_list, submission_detail, submission_export, submission_get, submission_list

Examples

```
## Not run:
t <- tempdir()
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.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"))

# 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(.) %>%
  attachment_link(.) %>%
  parse_datetime(tz = "Australia/Perth")

## End(Not run)
```

attachment_list — *List all submissions of one form.*

Description

Stable

Usage

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

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: `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")`.

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 (see below).

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://odkcentral.docs.apiary.io/#reference/forms-and-submissions/attachments/listing-expected-submission-attachments

https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/'-form-attachments/listing-expected-form-attachments

Other restful-api: attachment_link, audit_get, form_detail, form_list, form_schema_parse, form_schema, form_xml, project_create, project_detail, project_list, submission_detail, submission_export, submission_get, submission_list

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.svc"
  ),
  un = "me@email.com",
  pw = "..."
)

sl <- submission_list()

al <- attachment_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)
```

attachment_url *Build the download URL for one or many submission UUIDs and filenames.*

Description

Stable

Usage

```
attachment_url(uuid, fn, pid = get_default_pid(),
  fid = get_default_fid(), url = get_default_url())
```

Arguments

`uuid`	The UUID of one form submission, or a vector of UUIDs.
`fn`	The attachment filename, as per ODK form submission, or a vector of attachment filenames.
`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")`.

Details

This is a helper function used by ‘attachment_get‘. This function is vectorised and accepts single values or vectors for uuid and fn.

Value

The inferred download URL.

See Also

Other odata-api: `attachment_get`, `get_one_attachment`, `odata_metadata_get`, `odata_service_get`, `odata_submission_get`, `odata_submission_parse`, `unnest_all` audit_get *Get server audit log entries.*

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"))
```

Arguments

`action` string. The action to filter the logs, e.g. "user.create". See https://odkcentral.docs.apiary.io/#reference/system-endpoints/server-audit-logs/ 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 `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")`.

Details

Parameters to filter the audit logs: ‘action=form.create&start=2000-01-01z&end=2000-12-31T23

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: dttm. Time of action on server.

See Also

https://odkcentral.docs.apiary.io/#reference/system-endpoints/server-audit-logs/getting-audit-log-entries

Other restful-api: attachment_link, attachment_list, form_detail, form_list, form_schema_parse, form_schema, form_xml, project_create, project_detail, project_list, submission_detail, submission_export, submission_get, submission_list

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.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)
```

form_detail *Show details for one form.*

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())
```

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 `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")`.

Value

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

See Also

`https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/'-individual-form`

Other restful-api: `attachment_link`, `attachment_list`, `audit_get`, `form_list`, `form_schema_parse`, `form_schema`, `form_xml`, `project_create`, `project_detail`, `project_list`, `submission_detail`, `submission_export`, `submission_get`, `submission_list`

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.svc"
  ),
  un = "me@email.com",
  pw = "..."
)

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

# The first form in the test project
f <- form_detail(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" "last_submission" "hash"

## End(Not run)
```

---

form_list *List all forms.*

Description

Stable

Usage

```
form_list(pid = get_default_pid(), url = get_default_url(),
  un = get_default_un(), pw = get_default_pw())
```

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").

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").

Value

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

See Also

https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/forms

Other restful-api: attachment_link, attachment_list, audit_get, form_detail, form_schema_parse, form_schema, form_xml, project_create, project_detail, project_list, submission_detail, submission_export, submission_get, submission_list

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.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")

## End(Not run)
```

form_schema *Show the schema of one form.*

Description

Stable

Usage

```
form_schema(flatten = FALSE, odata = FALSE, parse = FALSE,
  pid = get_default_pid(), fid = get_default_fid(),
  url = get_default_url(), un = get_default_un(),
  pw = get_default_pw())
```

Arguments

flatten	Whether to flatten the resulting list of lists (TRUE) or not (FALSE, default).
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. 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 'flatten=TRUE' `` `form_schema` `` will raise a warning and return the unparsed, flattened form schema.
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")`.

Value

A nested list 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 XForm field type, as in "string", "select1", "geopoint", "binary" and so on). These fields are nested in lists of tuples ‘name‘ (the XForm 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.).

See Also

https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/'-individual-form/retrieving-form-schema-json

Other restful-api: attachment_link, attachment_list, audit_get, form_detail, form_list, form_schema_parse, form_xml, project_create, project_detail, project_list, submission_detail, submission_export, submission_get, submission_list

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.svc"
  ),
  un = "me@email.com",
  pw = "..."
)

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

# With defaults
fs_nested <- form_schema(
  flatten = FALSE,
  odata = FALSE
)
listviewer::jsonedit(fs_nested)

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

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

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 XForm field group.
names(fs_nested[[1]])
# > "children" ...

# 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)
fsp <- form_schema(parse = TRUE)

# Attachments: use \code{`attachment_get`} on each of
fsp %>% dplyr::filter(type == "binary")

# dateYime: use \code{`parse_datetime`} on each of
fsp %>% dplyr::filter(type == "dateTime")

# Point location: will be split into lat/lon/alt/acc
fsp %>% dplyr::filter(type == "geopoint")

## End(Not run)
```

`form_schema_parse` *Parse a form_schema into a tibble of fields with name, type, and path.*

Description

Maturing

Usage

```
form_schema_parse(fs, verbose = FALSE)
```

Arguments

fs	The output of form_schema as nested list
verbose	Whether to display debug messages or not (default)

Details

The ‘form_schema‘ 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 can be used to inform the user which columns require `` `parse_datetime` ``, `` `attachment_get` ``, or `` `attachment_link` ``, respectively.

See Also

Other restful-api: `attachment_link`, `attachment_list`, `audit_get`, `form_detail`, `form_list`, `form_schema`, `form_xml`, `project_create`, `project_detail`, `project_list`, `submission_detail`, `submission_export`, `submission_get`, `submission_list`

Examples

```
## Not run:
# Option 1: in two steps
fs <- form_schema(flatten = FALSE) # Default, but shown for clarity
fsp <- form_schema_parse(fs)

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

fsp

# Attachments: use \code{`attachment_get`} on each of
fsp %>% dplyr::filter(type == "binary")

# dateYime: use \code{`parse_datetime`} on each of
fsp %>% dplyr::filter(type == "dateTime")

# Point location: will be split into lat/lon/alt/acc
fsp %>% dplyr::filter(type == "geopoint")

## 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(), url = get_default_url(),
  un = get_default_un(), pw = get_default_pw())
```

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 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").

Value

The form XML as a nested list.

See Also

https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/'-individual-form/retrieving-form-xml

Other restful-api: attachment_link, attachment_list, audit_get, form_detail, form_list, form_schema_parse, form_schema, project_create, project_detail, project_list, submission_detail, submission_export, submission_get, submission_list

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.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"

## End(Not run)
```

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_raw_strata`, `fq_raw_taxa`, `fq_raw`, `fq_svc`

`fq_raw` *OData submission data for an ODK Central form.*

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_meta`, `fq_raw_strata`, `fq_raw_taxa`, `fq_svc`

`fq_raw_strata` *OData submission data for a subgroup of an ODK Central form.*

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_meta`, `fq_raw_taxa`, `fq_raw`, `fq_svc`

`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_meta`, `fq_raw_strata`, `fq_raw`, `fq_svc`

`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_meta`, `fq_raw_strata`, `fq_raw_taxa`, `fq_raw`

`get_one_attachment` *Download one media attachment.*

Description

Stable

Usage

```
get_one_attachment(pth, fn, src, url = get_default_url(),
  un = get_default_un(), pw = get_default_pw(), verbose = FALSE)
```

Arguments

`pth`	A local file path to save the attachment to.
`fn`	The attachment filename, as per ODK form submission. If NA, no file will be downloaded, but NA will be returned.
`src`	The attachment’s download URL, generated by ‘attachment_url‘.
`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")`.
`verbose`	Whether to display debug messages or not (default).

Details

This is a helper function used by ‘get_attachment‘. This function is not vectorised, but mapped by ‘get_attachment‘ to a tibble of input parameters.

Value

The relative local path to the downloaded attachment or NA.

See Also

Other odata-api: `attachment_get`, `attachment_url`, `odata_metadata_get`, `odata_service_get`, `odata_submission_get`, `odata_submission_parse`, `unnest_all`

`isodt_to_local` *Parse an ISO8601 datetime string to a timezone aware datetime.*

Description

Stable

Usage

```
isodt_to_local(datetime_string, orders = c("YmdHMS", "YmdHMSz"),
  tz = "UTC")
```

Arguments

`datetime_string`
(character) An ISO8601 datetime string as produced by XForms exported from ODK Central.

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

`tz` (character) The timezone string for lubridate. Default: `"UTC"`.

Value

A lubridate PosixCT datetime in the given timezone.

See Also

Other utilities: `%>%`, `parse_datetime`, `prepend_uuid`, `strip_uuid`, `tidyeval`

`odata_metadata_get` *Retrieve metadata from an OData URL ending in .svc as list of lists.*

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())
```

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 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").

Value

A nested list containing Edmx (dataset schema definition) and .attrs (Version).

See Also

https://odkcentral.docs.apiary.io/#reference/odata-endpoints/odata-form-service/metadata-document

Other odata-api: attachment_get, attachment_url, get_one_attachment, odata_service_get, odata_submission_get, odata_submission_parse, unnest_all

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.svc"
  ),
  un = "me@email.com",
  pw = "..."
)

meta <- odata_metadata_get()
listviewer::jsonedit(meta)

## End(Not run)
```

odata_service_get *Retrieve service metadata from an OData URL ending in .svc as list of lists*

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())
```

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 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").

Value

A tibble with one row per submission data endpoint. Columns: name, kind, url.

See Also

https://odkcentral.docs.apiary.io/#reference/odata-endpoints/odata-form-service/service-document

Other odata-api: attachment_get, attachment_url, get_one_attachment, odata_metadata_get, odata_submission_get, odata_submission_parse, unnest_all

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.svc"
  ),
  un = "me@email.com",
  pw = "..."
)

svc <- odata_service_get()
svc

## End(Not run)
```

odata_submission_get *Retrieve /Submissions from an OData URL ending in .svc as list of lists.*

Description

Maturing

Usage

```
odata_submission_get(table = "Submissions", skip = NA, top = NA,
  count = FALSE, wkt = FALSE, pid = get_default_pid(),
  fid = get_default_fid(), url = get_default_url(),
  un = get_default_un(), pw = get_default_pw())
```

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 in the metadata under DataServices.Schema.[EntityType].Name

`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.

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").

Details

Coming soon: [Parse submissions and download attachments](https://github.com/dbca-wa/ruODK/issues/6)

Value

A nested list of lists. ‘value‘ contains the submissions, which can be "rectangled" using ‘tidyr::unnest_wider("element_name" ‘@odata.context‘ is the URL of the metadata. ‘@odata.count‘ is the total number of rows in the table.

See Also

https://odkcentral.docs.apiary.io/#reference/odata-endpoints/odata-form-service

https://odkcentral.docs.apiary.io/#reference/odata-endpoints/odata-form-service/data-document

Other odata-api: attachment_get, attachment_url, get_one_attachment, odata_metadata_get, odata_service_get, odata_submission_parse, unnest_all

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.svc"
  ),
  un = "me@email.com",
  pw = "..."
)

data <- odata_submissions_get(table = "Submissions")

# Skip one row, return the next 1 rows (top), include total row count
```

```
data <- odata_submissions_get(
  table = "Submissions",
  skip = 1,
  top = 1,
  count = TRUE
)

# Parse point coordinates into lat, lon, alt
data <- odata_submissions_get(
  table = "Submissions",
  wkt = FALSE
)

# Parse point coordinates into WKT like "POINT (115.8840312 -31.9961844 0)"
data <- odata_submissions_get(
  table = "Submissions",
  wkt = TRUE
)

listviewer::jsonedit(data)

# Next: parse this nested list in to a tidy tibble with `parse_submissions`

## End(Not run)
```

odata_submission_parse

Parse the output of ‘odata_submission_get‘ into a tidy tibble and unnest all levels.

Description

Maturing

Usage

```
odata_submission_parse(data, names_repair = "universal",
  verbose = FALSE)
```

Arguments

`data`	A nested list of lists as given by ‘ruODK::get_submissions‘.
`names_repair`	The argument ‘names_repair‘ for ‘tidyr::unnest_wider‘, default: "universal".
`verbose`	Whether to print verbose messages, default: FALSE.

Details

Coming soon: [Better column names](https://github.com/dbca-wa/ruODK/issues/7)

Value

The submissions as unnested tibble

See Also

Other odata-api: attachment_get, attachment_url, get_one_attachment, odata_metadata_get, odata_service_get, odata_submission_get, unnest_all

Examples

```
## Not run:
# Using canned data
data_parsed <- parse_submissions(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 *Retrieve URL, project ID, and form ID from an ODK Central OData service URL.*

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://sandbox.central.opendatakit.org/v1/projects/14/forms/build_Flo Quadrat-0-2_1558575936.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: ru_settings, ru_setup, yell_if_error, yell_if_missing

parse_datetime — *Parse all columns containing a certain word in its name to localised datetime.*

Description

Stable

Usage

```
parse_datetime(df, col_contains = "time", orders = c("YmdHMS",
  "YmdHMSz"), tz = "Australia/Perth")
```

Arguments

df	A dataframe or tibble.
col_contains	A character string indicating a date/time column. This can be a part of the column name or the whole column name. Default: "time" will match all column names containing "time".
orders	(vector of character) Orders of datetime elements for lubridate. Default: `c("YmdHMS","YmdHMSz")`.
tz	(character) The timezone string for lubridate. Default: `"UTC"`.

Details

This function wraps a `` `dplyr::mutate_at()` `` operation on all datetime columns containing a telltale ‘col_contains‘ string like "time" (default), "datetime" or "date". The operator using this function will have to know the column names (or commonalities like ‘"..._time"‘).

Value

The dataframe with matching columns mutated to lubridate datetimes.

See Also

Other utilities: %>%, isodt_to_local, prepend_uuid, strip_uuid, tidyeval

Examples

```
## Not run:
tsi <- "data/build_Turtle-Sighting-0-1_1559790020.csv" %>%
  readr::read_csv(na = c("", "NA", "na")) %>%
  janitor::clean_names() %>%
  link_attachments() %>%
  parse_datetime(dt_contains = "Date", tz = "Australia/Perth") %>%
  parse_datetime(tz = "Australia/Perth")

## End(Not run)
```

prepend_uuid *Prepend a leading "uuid:" to any string, e.g. an md5 hash.*

Description

Stable

Usage

```
prepend_uuid(md5hash)
```

Arguments

md5hash	A string, e.g. an md5 hash.

Value

The string with a prepended "uuid:"

See Also

Other utilities: %>%, isodt_to_local, parse_datetime, strip_uuid, tidyeval

project_create *Create a new project.*

Description

Experimental

Usage

```
project_create(name, url = get_default_url(), un = get_default_un(),
  pw = get_default_pw())
```

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 `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")`.

Value

A tibble with one row per project and all project metadata as columns as per ODK Central API docs.

See Also

https://odkcentral.docs.apiary.io/#reference/project-management/projects/creating-a-project

Other restful-api: attachment_link, attachment_list, audit_get, form_detail, form_list, form_schema_parse, form_schema, form_xml, project_detail, project_list, submission_detail, submission_export, submission_get, submission_list

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.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_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())
```

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")`.
`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")`.

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://odkcentral.docs.apiary.io/#reference/project-management/projects/getting-project-details`

Other restful-api: `attachment_link`, `attachment_list`, `audit_get`, `form_detail`, `form_list`, `form_schema_parse`, `form_schema`, `form_xml`, `project_create`, `project_list`, `submission_detail`, `submission_export`, `submission_get`, `submission_list`

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.svc"
  ),
  un = "me@email.com",
  pw = "..."
)

pd <- project_detail()

pd %>%
  dplyr::select(-"verbs") %>%
  knitr::kable(.)

## 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())
```

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: `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")`.

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://odkcentral.docs.apiary.io/#reference/project-management/projects/listing-projects

Other restful-api: attachment_link, attachment_list, audit_get, form_detail, form_list, form_schema_parse, form_schema, form_xml, project_create, project_detail, submission_detail, submission_export, submission_get, submission_list

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.svc"
  ),
  un = "me@email.com",
  pw = "..."
)

pl <- project_list()

knitr::kable(p)

# project_list returns a tibble
class(p)
# > "tbl_df" "tbl" "data.frame"

# columns are project metadata
names(p)
# > "id" "name" "forms" "app_users" "last_submission"
# > "created_at" "updated_at" "archived"

## End(Not run)
```

ru_settings *Get or set* ruODK *settings.*

Description

Stable

Usage

```
ru_settings()

get_default_pid()

get_default_fid()

get_default_url()

get_default_un()

get_default_pw()

get_test_url()

get_test_un()

get_test_pw()

get_test_pid()

get_test_fid()

get_test_fid_zip()

get_test_fid_att()
```

Value

`ru_settings` prints your default ODK Central project ID, form ID, url, username, and password, and corresponding optional test server settings. `ru_setup` sets your production and test settings, while `get_(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_test_pid`, `get_test_fid`, `get_test_fid_zip`, `get_test_fid_att`, `get_test_url`, `get_test_un`, `get_test_pw`.

Other ru_settings: `odata_svc_parse`, `ru_setup`, `yell_if_error`, `yell_if_missing`

Examples

```
ru_settings()
```

ru_setup *Configure default* ruODK *settings.*

Description

Stable

Usage

```
ru_setup(svc = NULL, pid = NULL, fid = NULL, url = NULL,
  un = NULL, pw = NULL, test_svc = NULL, test_pid = NULL,
  test_fid = NULL, test_fid_zip = NULL, test_fid_att = NULL,
  test_url = NULL, test_un = NULL, test_pw = 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.opendatakit.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).

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_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`.

Details

`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`, `ru_settings`, `yell_if_error`, `yell_if_missing`

Examples

```
# \code{ruODK} users only need default settings to their ODK Central instance:
ru_setup(url = "https://my-odkc.com", un = "me@email.com", pw = "...")

# \code{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 = "...",
  test_url = "https://sandbox.central.opendatakit.org",
  test_un = "me@email.com",
  test_pw = "...",
  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"
)
```

`strip_uuid` *Strip the leading "uuid:" from a UUID hash.*

Description

Stable

Usage

```
strip_uuid(uuid)
```

Arguments

uuid	A string which may contain any number of "uuid:"

Value

The string with every occurrence of "uuid:" deleted.

See Also

Other utilities: %>%, isodt_to_local, parse_datetime, prepend_uuid, tidyeval

---

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())
```

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: 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").

Value

A nested list of submission metadata.

See Also

https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/submissions/getting-submission-details

Other restful-api: attachment_link, attachment_list, audit_get, form_detail, form_list, form_schema_parse, form_schema, form_xml, project_create, project_detail, project_list, submission_export, submission_get, submission_list

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.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
names(sub)
# > "instance_id" "submitter_id" "submitter" "created_at" "updated_at"

## End(Not run)
```

submission_export *Export all form submissions including repeats and attachments to CSV.*

Description

To export all the Submission data associated with a Form, just add .csv.zip to the end of the listing URL. The response will be a zip file containing one or more CSV files, as well as all multimedia attachments associated with the included Submissions.

Usage

```
submission_export(local_dir = here::here(), overwrite = TRUE,
  pid = get_default_pid(), fid = get_default_fid(),
  url = get_default_url(), un = get_default_un(),
  pw = get_default_pw(), verbose = FALSE)
```

Arguments

`local_dir`	The local folder to save the downloaded files to, default: 'here::here()'.
`overwrite`	Whether to overwrite previously downloaded zip files, default: FALSE
`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")`.
`verbose`	Whether to display debug messages or not (default)

Details

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).

This function downloads all submissions and attachments in one go. For incremental download of a subset of submissions, use 'submission_list', choose the submissions of interest (e.g. by submission date), and use their uuids to download them one by one via 'submission_get'. Download attachments as listed for each submission ('attachment_list').

Stable

Value

The absolute path to the zip file named "'fid'.zip" containing submissions as CSV, plus separate CSVs for any repeating groups, plus any attachments in a subfolder 'media'.

See Also

https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/submissions/exporting-form-submissions-to-csv

Other restful-api: attachment_link, attachment_list, audit_get, form_detail, form_list, form_schema_parse, form_schema, form_xml, project_create, project_detail, project_list, submission_detail, submission_get, submission_list

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.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_get *Download one submission.*

Description

Stable

Usage

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

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: `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")`.

Value

A nested list of submission data.

See Also

https://odkcentral.docs.apiary.io/#reference/forms-and-submissions/submissions/retrieving-submission-xml

Other restful-api: `attachment_link`, `attachment_list`, `audit_get`, `form_detail`, `form_list`, `form_schema_parse`, `form_schema`, `form_xml`, `project_create`, `project_detail`, `project_list`, `submission_detail`, `submission_export`, `submission_list`

Examples

```
## Not run:
# Set default credentials, see vignette "setup"
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.svc"
  ),
  un = "me@email.com",
  pw = "..."
)

# With explicit credentials, see tests
sl <- submission_list()

sub <- submission_get(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)
```

submission_list *List all submissions of one form.*

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())
```

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 `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")`.

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://odkcentral.docs.apiary.io/#reference/forms-and-submissions/submissions/listing-all-submissions-on-a-form

Other restful-api: attachment_link, attachment_list, audit_get, form_detail, form_list, form_schema_parse, form_schema, form_xml, project_create, project_detail, project_list, submission_detail, submission_export, submission_get

Examples

```
## Not run:
# Set default credentials, see vignette("setup")
ruODK::ru_setup(
  svc = paste0(
    "https://sandbox.central.opendatakit.org/v1/projects/14/",
    "forms/build_Flora-Quadrat-0-2_1558575936.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)
```

`unnest_all` *Recursively unnest_wide all list columns in a tibble.*

Description

Stable

Usage

```
unnest_all(nested_tbl, names_repair = "universal", verbose = FALSE)
```

Arguments

`nested_tbl`	A nested tibble
`names_repair`	The argument ‘names_repair‘ for ‘tibble::unnest_wider‘, default: "universal".
`verbose`	Whether to print verbose messages, default: FALSE.

Value

The unnested tibble in wide format

See Also

Other odata-api: `attachment_get`, `attachment_url`, `get_one_attachment`, `odata_metadata_get`, `odata_service_get`, `odata_submission_get`, `odata_submission_parse`

`yell_if_error` *Warn about failed web requests and give helpful troubleshooting tips.*

Description

Stable

Usage

```
yell_if_error(response, url, un, pw, pid = NULL, fid = NULL)
```

Arguments

`response`	A httr response object
`url`	A URL (character)
`un`	A username (character)
`pw`	A password (character)
`pid`	A project ID (numeric, optional)
`fid`	A form ID (character, optional)

Details

A wrapper around `httr::stop_for_status()` with a helpful error message. Examples: see tests for `ruODK::project_list()`. This function is used internally but may be useful for debugging and `ruODK` development.

Value

The response object

See Also

Other ru_settings: `odata_svc_parse`, `ru_settings`, `ru_setup`, `yell_if_missing`

`yell_if_missing` *Abort on missing ODK Central credentials (url, username, password).*

Description

Stable

Usage

```
yell_if_missing(url, un, pw, pid = NULL, fid = NULL)
```

Arguments

`url`	A URL (character)
`un`	A username (character)
`pw`	A password (character)
`pid`	A project ID (numeric, optional)
`fid`	A form ID (character, optional)

Details

This is a helper function to pat down `ruODK` functions for missing credentials and stop with a loud but informative yell.

See Also

Other ru_settings: `odata_svc_parse`, `ru_settings`, `ru_setup`, `yell_if_error`

Examples

```
testthat::expect_error(yell_if_missing("", "username", "password"))
testthat::expect_error(yell_if_missing("url", "", "password"))
testthat::expect_error(yell_if_missing("url", "username", ""))
testthat::expect_error(yell_if_missing(NULL, "", ""))
testthat::expect_error(yell_if_missing("", "", ""))
testthat::expect_error(yell_if_missing("", "", "", ""))
testthat::expect_error(yell_if_missing("", "", "", "", ""))
```

Index