Package ‘JirAgileR’

June 8, 2021

Title JIRA REST API Wrapper for R
Version 0.0.2.1
Description Allows to interact with the 'JIRA SERVER REST API' to analyze the retrieved data in R. For further information about the API visit <https://docs.atlassian.com/software/jira/docs/api/REST/8.9.1/>.
License MIT + file LICENSE
BugReports https://github.com/matbmeijer/JirAgileR/issues
Imports httr, jsonlite, magrittr
Suggests covr, testthat (>= 2.1.0), knitr, rmarkdown, dplyr
RoxygenNote 7.1.1
Encoding UTF-8
Language en-US
NeedsCompilation no
Author Matthias Brenninkmeijer [aut, cre]
Maintainer Matthias Brenninkmeijer <matthias@brennotten.net>
Repository CRAN
Date/Publication 2021-06-08 08:30:02 UTC

R topics documented:

adapt_list ............................................................... 2
basic_issues_info ..................................................... 3
basic_jql_fields ...................................................... 3
choose_field_function ............................................... 4
conc ................................................................. 4
get_jira_credentials .................................................. 5
get_jira_dashboards .................................................. 5
get_jira_groups ....................................................... 6
adapt_list

Adapt the path of class `url`

Description

Adapt the path of class `url` to consider the old path when modifying.

Usage

```r
adapt_list(old_path, path)
```

Arguments

- `old_path`  Passed path in parameter domain
- `path`  Path of API endpoint

Value

Returns a vector of the resulting path.

Warning

Internal function
**basic_issues_info**

**Extract the basic key information of the issues**

---

**Description**

Internal function to extract the basic key information as a data.frame.

**Usage**

```r
basic_issues_info(x)
```

**Arguments**

- `x` JIRA issue list item.

**Value**

Returns data.frame with basic field information.

**Warning**

Internal function

---

**basic_jql_fields**

**Returns default JQL fields used**

---

**Description**

Internal function used to define the default JQL fields used for the `get_jira_issues()` function.

**Usage**

```r
basic_jql_fields()
```

**Value**

Returns a character vector with the default JQL fields.

**Warning**

Internal function
choose_field_function  Function to choose for the right field parser function

Description

Internal function to choose/switch to the correct function to parse each field for each issue

Usage

choose_field_function(x, type)

Arguments

  x              The fields nested data to flatten.
  type           The fields' name.

Value

Returns a parsed, cleaned data.frame with all the fields.

Warning

Internal function

Author(s)

Matthias Brenninkmeijer https://github.com/matbmeijer

conc  Concatenates multiple strings

Description

Internal function with an opinionated default behavior to concatenate character values.

Usage

conc(x, y = "", decr = FALSE, unique = TRUE)

Arguments

  x                        A single character vector to concatenate together.
  y                        By default a , string used to define the character to collapse the x parameter.
  decr                    Optional logical parameter that defines the sorting order, by default set to FALSE, which results in an alphabetical order.
  unique                  Optional logical parameter to concatenate only unique values, by default set to TRUE
get_jira_credentials

Value

Returns a single character string.

Warning

Internal function

Description

Retrieves a data.frame with the JIRA credentials previously saved into the environment under the JIRAGILER_PAT variable through the save_jira_credentials() function.

Usage

get_jira_credentials()

Value

Returns a data.frame with the saved JIRA credentials

Examples

## Not run:
save_jira_credentials(domain="https://bugreports.qt.io",
    username='__INSERT_YOUR_USERNAME_HERE__',
    password='__INSERT_YOUR_PASSWORD_HERE__')

get_jira_credentials()

## End(Not run)

get_jira_dashboards

Retrieves all dashboards a data.frame

Description

Calls JIRA's latest REST API, optionally with basic authentication, to get all dashboards

Usage

get_jira_dashboards(
    domain = NULL,
    username = NULL,
    password = NULL,
    maxResults = 20L,
    verbose = FALSE
)
get_jira_groups

Arguments

domain
Custom JIRA domain URL as for example https://bugreports.qt.io. Can be passed as a parameter or can be previously defined through the save_jira_credentials() function.

username
Username used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.

password
Password used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.

maxResults
Max results authorized to obtain for each API call. By default JIRA sets this value to 20 issues.

verbose
Explicitly informs the user of the JIRA API request process.

Value

Returns a flattened, formatted data.frame with the dashboards in the domain.

See Also

For more information about Atlassians JIRA API visit the following link: JIRA API Documentation.

Examples

get_jira_dashboards(domain = "https://bugreports.qt.io")

get_jira_groups

Get JIRA server groups data.frame

Description

Makes a request to JIRA’s latest REST API to retrieve all the groups in JIRA.

Usage

get_jira_groups(
  domain = NULL,
  username = NULL,
  password = NULL,
  verbose = FALSE,
  maxResults = 1000
)
get_jira_issues

Arguments

- **domain**: Custom JIRA domain URL as for example `https://bugreports.qt.io`. Can be passed as a parameter or can be previously defined through the `save_jira_credentials()` function.
- **username**: Username used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.
- **password**: Password used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.
- **verbose**: Explicitly informs the user of the JIRA API request process. Optional parameter.
- **maxResults**: Number of maximum groups to return. Set by default to `1000`.

Value

Returns a data.frame with all the JIRA groups

Warning

The function works with the latest JIRA REST API and to work you need to have an internet connection. Calling the function too many times might block your access, you will receive a 403 error code. To unblock your access you will have to access interactively through your browser, signing out and signing in again, and might even have to enter a CAPTCHA at `https://jira.yourdomain.com/secure/Dashboard.jspa`. This only happens if the API is called upon multiple times in a short period of time.

See Also

For more information about Atlassian's JIRA API go to JIRA API Documentation

Examples

```r
## Not run:
get_jira_groups("https://bugreports.qt.io")

## End(Not run)
```

get_jira_issues  Retrieves all issues of a JIRA query as a data.frame

Description

Calls JIRA’s latest REST API, optionally with basic authentication, to get all issues of a JIRA query (JQL). Allows to specify which fields to obtain.
Usage

```r
get_jira_issues(
  domain = NULL,
  username = NULL,
  password = NULL,
  jql_query,
  fields = basic_jql_fields(),
  maxResults = 50,
  verbose = FALSE,
  as.data.frame = TRUE
)
```

Arguments

- **domain**: Custom JIRA domain URL as for example https://bugreports.qt.io. Can be passed as a parameter or can be previously defined through the `save_jira_credentials()` function.
- **username**: Username used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.
- **password**: Password used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.
- **jql_query**: JIRA's decoded JQL query. By definition, it works with:
  - Fields
  - Operators
  - Keywords
  - Functions
To learn how to create a query visit this ATLASSIAN site or the following cheat-sheet.
- **fields**: Optional argument to define the specific JIRA fields to obtain. If no value is entered, by default the following fields are passed:
  - status
  - priority
  - created
  - reporter
  - summary
  - description
  - assignee
  - updated
  - issuuetype
  - fixVersions
To obtain a list of all supported fields use the following function: `supported_jql_fields()`.
**get_jira_permissions**

- **maxResults**
  Max results authorized to obtain for each API call. By default JIRA sets this value to 50 issues.

- **verbose**
  Explicitly informs the user of the JIRA API request process.

- **as.data.frame**
  Defines if the function returns a flattened data.frame or the raw JIRA response.

**Value**

Returns a flattened, formatted data.frame with the issues according to the JQL query.

**Warning**

If the comment field is used as a fields parameter input, each issue and its attributes are repeated the number of comments the issue has. The function works with the latest JIRA REST API and to work you need to have a internet connection. Calling the function too many times might block your access, you will receive a 403 error code. To unblock your access you will have to access interactively through your browser, signing out and signing in again, and might even have to enter a CAPTCHA at https://jira.yourdomain.com/secure/Dashboard.jspa. This only happens if the API is called upon multiple times in a short period of time.

**See Also**

For more information about Atlassian’s JIRA API visit the following link: [JIRA API Documentation](#).

**Examples**

```r
get_jira_issues(domain = "https://bugreports.qt.io",
                 jql_query = "project="QTWB"")
```

---

**get_jira_permissions**  
*Get all the JIRA server permissions as a data.frame*

**Description**

Makes a request to JIRA’s latest REST API to retrieve the users’ permissions.

**Usage**

```r
get_jira_permissions(
  domain = NULL,
  username = NULL,
  password = NULL,
  verbose = FALSE
)
```
get_jira_projects

Arguments

- **domain**: Custom JIRA domain URL as for example https://bugreports.qt.io. Can be passed as a parameter or can be previously defined through the save_jira_credentials() function.
- **username**: Username used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.
- **password**: Password used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.
- **verbose**: Explicitly informs the user of the JIRA API request process.

Value

Returns a data.frame with all the JIRA user permissions.

Warning

The function works with the latest JIRA REST API and to work you need to have an internet connection. Calling the function too many times might block your access, you will receive a 403 error code. To unblock your access you will have to access interactively through your browser, signing out and signing in again, and might even have to enter a CAPTCHA at https://jira.yourdomain.com/secure/Dashboard.jspa. This only happens if the API is called upon multiple times in a short period of time.

See Also

For more information about Atlassians JIRA API go to JIRA API Documentation

Examples

```r
## Not run:
get_jira_permissions("https://bugreports.qt.io")

## End(Not run)
```

get_jira_projects Retrives all projects as a data.frame

Description

Makes a request to JIRA’s latest REST API to retrieve all projects and their basic project information (Name, Key, Id, Description, etc.).
get_jira_projects

Usage

get_jira_projects(
  domain = NULL,
  username = NULL,
  password = NULL,
  expand = NULL,
  verbose = FALSE
)

Arguments

domain Custom JIRA domain URL as for example https://bugreports.qt.io. Can be passed as a parameter or can be previously defined through the save_jira_credentials() function.

username Username used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.

password Password used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.

expand Specific JIRA fields the user wants to obtain for a specific field. Optional parameter.

verbose Explicitly informs the user of the JIRA API request process.

Value

Returns a data.frame with a list of projects for which the user has the BROWSE, ADMINISTER or PROJECT_ADMIN project permission.

Warning

The function works with the latest JIRA REST API and to work you need to have an internet connection. Calling the function too many times might block your access, you will receive a 403 error code. To unblock your access you will have to access interactively through your browser, signing out and signing in again, and might even have to enter a CAPTCHA at https://jira.yourdomain.com/secure/Dashboard.jspa. This only happens if the API is called upon multiple times in a short period of time.

See Also

For more information about Atlassians JIRA API go to JIRA API Documentation

Examples

```r
## Not run:
get_jira_projects("https://bugreports.qt.io")

## End(Not run)
```
get_jira_server_info  Get the JIRA server information as a data.frame

Description

Makes a request to JIRA's latest REST API to retrieve all the necessary information regarding the JIRA server version.

Usage

get_jira_server_info(
    domain = NULL,
    username = NULL,
    password = NULL,
    verbose = FALSE
)

Arguments

domain  Custom JIRA domain URL as for example https://bugreports.qt.io. Can be passed as a parameter or can be previously defined through the save_jira_credentials() function.

username  Username used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.

password  Password used to authenticate the access to the JIRA domain. If both username and password are not passed no authentication is made and only public domains can be accessed. Optional parameter.

verbose  Explicitly informs the user of the JIRA API request process.

Value

Returns a data.frame with all the JIRA server information

Warning

The function works with the latest JIRA REST API and to work you need to have an internet connection. Calling the function too many times might block your access, you will receive a 403 error code. To unblock your access you will have to access interactively through your browser, signing out and signing in again, and might even have to enter a CAPTCHA at https://jira.yourdomain.com/secure/Dashboard.jspa. This only happens if the API is called upon multiple times in a short period of time.

See Also

For more information about Atlassian's JIRA API go to JIRA API Documentation
parse_issue

Examples

```r
## Not run:
get_jira_server_info("https://bugreports.qt.io")
```

## End(Not run)

---

**parse_issue**  
*Extract the extensive fields of a single issue*

**Description**

Internal function to transform the nested more extensive JIRA issue fields into a flattened `data.frame`.

**Usage**

```r
parse_issue(issue, JirAgileR_id)
```

**Arguments**

- `issue`  
  A JIRA issue with all its extended fields

- `JirAgileR_id`  
  JirAgileR ID to assign to

**Value**

Returns `data.frame` with all the extended field information.

**Warning**

Internal function

---

**remove_jira_credentials**  
*Removes previously saved JIRA credentials*

**Description**

Removes the JIRA credentials, that have been previously saved into the environment under the `JIRAGILER_PAT` variable through the `save_jira_credentials()` function.

**Usage**

```r
remove_jira_credentials(verbos = FALSE)
```

**Arguments**

- `verbose`  
  Optional parameter to remove previously saved parameters
**save_jira_credentials**

**Value**

Returns a data.frame with the saved JIRA credentials

**Examples**

```r
## Not run:
save_jira_credentials(domain="https://bugreports.qt.io")
remove_jira_credentials()

## End(Not run)
```

**Description**

Saves the domain and/or username and password in the users’ environment. It has the advantage that it is not necessary to explicitly publish the credentials in the users code. Just do it one time and you are set. To update any of the parameters just save again and it will overwrite the older credential.

**Usage**

```r
save_jira_credentials(
  domain = NULL,
  username = NULL,
  password = NULL,
  verbose = FALSE
)
```

**Arguments**

- **domain**
  
  The users’ JIRA server domain to retrieve information from. An example would be https://bugreports.qt.io. It will be saved in the environment as JIRAGILER_DOMAIN.

- **username**
  
  The users’ username to authenticate to the domain. It will be saved in the environment as JIRAGILER_USERNAME.

- **password**
  
  The users’ password to authenticate to the domain. It will be saved in the environment as JIRAGILER_PASSWORD. If verbose is set to TRUE, it will message asterisks.

- **verbose**

  Optional parameter to inform the user when the users’ credentials have been saved.

**Value**

Saves the credentials in the users environment - it does not return any object.
Examples

```r
## Not run:
save_jira_credentials(domain="https://bugreports.qt.io",
    username="__INSERT_YOUR_USERNAME_HERE__",
    password="__INSERT_YOUR_PASSWORD_HERE__")

## End(Not run)
```

---

**supported_jql_fields**  
*Returns the supported JQL fields*

**Description**

Function shows all the supported JQL fields that are available to choose for the `get_jira_issues()` function.

**Usage**

```r
supported_jql_fields()
```

**Value**

Returns a character vector of all the supported JQL fields.

---

**to_date**  
*Transforms JIRA date character to POSIXlt format*

**Description**

Internal function to parse the date from JIRA character vectors.

**Usage**

```r
to_date(x)
```

**Arguments**

- `x`  
  Character vector to transform into a POSIXlt.

**Value**

Returns a POSIXlt object vector.

**Warning**

Internal function
unnest_df

Unnest a nested data.frame

Description
Unnests/flattens a nested data.frame

Usage
unnest_df(x)

Arguments
x A nested data.frame object

Value
Returns a flattened data.frame.

Warning
Internal function
Index

adapt_list, 2
basic_issues_info, 3
basic_jql_fields, 3
choose_field_function, 4
conc, 4
get_jira_credentials, 5
get_jira_dashboards, 5
get_jira_groups, 6
get_jira_issues, 7
get_jira_permissions, 9
get_jira_projects, 10
get_jira_server_info, 12
parse_issue, 13
remove_jira_credentials, 13
save_jira_credentials, 14
supported_jql_fields, 15
to_date, 15
unnest_df, 16