

Package ‘distilleR’

July 28, 2026

Type Package

Title A Wrap Around the 'DistillerSR' APIs

Version 1.1.0

Maintainer Luca Belmonte <luca.belmonte@efsa.europa.eu>

Description Interface to 'DistillerSR' APIs. See
<<https://apidocs.evidencepartners.com/>> for more details.

License EUPL-1.2

URL <https://openefsa.github.io/distilleR/>

BugReports <https://github.com/openefsa/distilleR/issues>

Depends R (>= 4.1.0)

Imports cli (>= 3.6.5), checkmate (>= 2.3.1), glue (>= 1.7.0), httr2
(>= 1.2.1), jsonlite (>= 1.8.7), readr (>= 2.1.5), readxl (>= 1.4.3), tibble (>= 3.3.0), purrr (>= 1.0.2)

Suggests devtools (>= 2.4.5), knitr (>= 1.0), rmarkdown (>= 2.0),
roxygen2 (>= 7.2.1), testthat (>= 3.0.0), usethis (>= 2.2.3),
covr (>= 3.6.4), openxlsx (>= 4.2.8), rlang (>= 1.1.4)

Encoding UTF-8

RoxygenNote 7.3.3

Config/testthat/edition 3

VignetteBuilder knitr

Config/Needs/website pkgdown

Repository CRAN

NeedsCompilation no

Author Lorenzo Copelli [aut] (ORCID: <<https://orcid.org/0009-0002-4305-065X>>),
Fulvio Barizzzone [aut] (ORCID: <<https://orcid.org/0009-0006-3035-520X>>),
Dayana Stephanie Buzle [aut] (ORCID:
<<https://orcid.org/0009-0003-2990-7431>>),
Rafael Vieira [aut] (ORCID: <<https://orcid.org/0009-0009-0289-5438>>),
Luca Belmonte [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-7977-9170>>)

Date/Publication 2026-07-28 10:10:07 UTC

Contents

getAsyncReportResult	2
getAsyncReportStatus	3
getAuthenticationToken	5
getProjects	6
getReport	7
getReportAsync	8
getReports	10
Index	12

getAsyncReportResult	<i>Get the result of an asynchronous job to retrieve a Distiller report.</i>
----------------------	--

Description

This function gets the result of a successful Distiller asynchronous job to retrieve a saved report associated with a given project ID. It requires a valid asynchronous job token. The result is a dataframe containing metadata about the saved report.

Usage

```
getAsyncReportResult(  
  jobToken,  
  format = c("excel", "csv"),  
  distillerAsyncInstanceUrl = Sys.getenv("DISTILLER_ASYNC_INSTANCE_URL"),  
  timeout = 1800  
)
```

Arguments

- jobToken character (string). The token associated to the successful asynchronous job.
- format character (string). The desired format for the document. It can be either excel or csv.
- distillerAsyncInstanceUrl
 character (string, optional). The asynchronous Distiller instance URL.
 By default: Sys.getenv("DISTILLER_ASYNC_INSTANCE_URL").
- timeout integer (optional). The maximum number of seconds to wait for the response.
 By default: 1800 seconds (30 minutes).

Value

A data frame containing the Distiller report as designed within DistillerSR.

See Also

[getAuthenticationToken](#)
[getProjects](#)
[getReports](#)
[getReportAsync](#)
[getAsyncReportStatus](#)

Examples

```
## Not run:
distillerToken_ <- getAuthenticationToken()

projects_ <- getProjects(distillerToken = distillerToken_)

reports_ <- getReports(
  projectId = projects_$id[1],
  distillerToken = distillerToken_)

job_ <- getReportAsync(
  projectId = projects_$id[1],
  reportID = reports_$id[7],
  format = "csv",
  distillerToken = distillerToken_)

jobToken_ <- job_$token

jobResult_ <- getAsyncReportResult(jobToken = jobToken_, format = "csv")

## End(Not run)
```

getAsyncReportStatus *Get the status of an asynchronous job to retrieve a Distiller report.*

Description

This function gets the status of a successfully submitted Distiller asynchronous job to retrieve a saved report associated with a given project ID. It requires a valid asynchronous job token. The result is a dataframe containing metadata about the job status.

Usage

```
getAsyncReportStatus(
  jobToken,
  distillerAsyncInstanceUrl = Sys.getenv("DISTILLER_ASYNC_INSTANCE_URL"),
  timeout = 1800
)
```

Arguments

`jobToken` character (string). The token associated to the submitted asynchronous job.

`distillerAsyncInstanceUrl` character (string, optional). The asynchronous Distiller instance URL.
By default: `Sys.getenv("DISTILLER_ASYNC_INSTANCE_URL")`.

`timeout` integer (optional). The maximum number of seconds to wait for the response.
By default: 1800 seconds (30 minutes).

Details

After receiving the response and parsing it to JSON, a sanitation step is applied on the data to transform NULL values to NA characters, in order for tibble's `as_tibble()` to handle them properly.

Value

A data frame containing metadata about the job status.

See Also

[getAuthenticationToken](#)
[getProjects](#)
[getReports](#)
[getReportAsync](#)
[getAsyncReportResult](#)

Examples

```
## Not run:
distillerToken_ <- getAuthenticationToken()

projects_ <- getProjects(distillerToken = distillerToken_)

reports_ <- getReports(
  projectId = projects_$id[1],
  distillerToken = distillerToken_)

job_ <- getReportAsync(
  projectId = projects_$id[1],
  reportID = reports_$id[7],
  format = "csv",
  distillerToken = distillerToken_)

jobToken_ <- job_$token

jobStatus_ <- getAsyncReportStatus(jobToken = jobToken_)

## End(Not run)
```

`getAuthenticationToken`*Authenticate to a DistillerSR session.*

Description

Authenticates a user to a DistillerSR instance using a personal access key. The function returns a valid authentication token that can be used to access protected DistillerSR API endpoints.

Usage

```
getAuthenticationToken(  
  distillerInstanceUrl = Sys.getenv("DISTILLER_INSTANCE_URL"),  
  distillerKey = Sys.getenv("DISTILLER_API_KEY"),  
  timeout = 1800  
)
```

Arguments

<code>distillerInstanceUrl</code>	character (string, optional). The base URL of the DistillerSR instance. By default: <code>Sys.getenv("DISTILLER_INSTANCE_URL")</code> .
<code>distillerKey</code>	character (string, optional). The personal access key generated in DistillerSR. By default: <code>Sys.getenv("DISTILLER_API_KEY")</code> .
<code>timeout</code>	integer (optional). The maximum number of seconds to wait for the authentication response. By default: 1800 seconds (30 minutes).

Details

By default, the personal access key and the instance URL are read from the environment variables `DISTILLER_API_KEY` and `DISTILLER_INSTANCE_URL`.

Value

A string containing a valid DistillerSR authentication token.

Examples

```
## Not run:  
# If 'DISTILLER_INSTANCE_URL' and 'DISTILLER_API_KEY' are defined in your  
# environment (e.g. .Renvirom).  
distillerToken_ <- getAuthenticationToken()  
  
# If 'distillerInstanceUrl' and 'distillerKey' are to be specified manually.  
distillerToken_ <- getAuthenticationToken(  
  distillerInstanceUrl = "https://url.to.distiller.instance",
```

```

    distillerKey = "YOUR_API_KEY")

## End(Not run)

```

getProjects	<i>Get the list of the Distiller projects associated to the authenticated user.</i>
-------------	---

Description

This function queries the DistillerSR API to retrieve the list of projects accessible to the authenticated user. It requires an authentication token and a valid API instance URL. The result is a dataframe listing available projects.

Usage

```

getProjects(
  distillerInstanceId = Sys.getenv("DISTILLER_INSTANCE_URL"),
  distillerToken,
  timeout = 1800
)

```

Arguments

distillerInstanceId	character (string, optional). The distiller instance URL. By default: Sys.getenv("DISTILLER_INSTANCE_URL").
distillerToken	character (string). The token the user gets once authenticated.
timeout	integer (optional). The maximum number of seconds to wait for the response. By default: 1800 seconds (30 minutes).

Value

A tibble with four columns:

- id: The project ID.
- name: The name of the project.
- de_project_id.
- is_hidden.

See Also

[getAuthenticationToken](#)

Examples

```
## Not run:
distillerToken_ <- getAuthenticationToken()

projects_ <- getProjects(distillerToken = distillerToken_)

## End(Not run)
```

getReport

*Get a Distiller report associated to a project of the authenticated user.***Description**

This function queries the DistillerSR API to retrieve a saved report associated with a given project ID. It requires user authentication and a valid API endpoint URL. The result is a dataframe containing metadata about the saved report.

Usage

```
getReport(
  projectId,
  reportId,
  format = c("excel", "csv"),
  distillerInstanceUrl = Sys.getenv("DISTILLER_INSTANCE_URL"),
  distillerToken,
  timeout = 1800,
  attempts = 1,
  retryEach = 600,
  verbose = TRUE
)
```

Arguments

projectId	integer. The ID of the project as provided by DistillerSR.
reportId	integer. The ID of the report as provided by DistillerSR.
format	character (string). The desired format for the document. It can be either excel or csv.
distillerInstanceUrl	character (string, optional). The distiller instance URL. By default: Sys.getenv("DISTILLER_INSTANCE_URL").
distillerToken	character (string). The token the user gets once authenticated.
timeout	integer (optional). The maximum number of seconds to wait for the response. By default: 1800 seconds (30 minutes).
attempts	integer (optional). The maximum number of attempts. By default: 1 attempt.

retryEach	integer (optional). The delay between attempts. By default: 600 seconds (10 minutes).
verbose	logical (optional). A flag to specify whether to make the function verbose or not. By default: TRUE.

Value

A data frame containing the Distiller report as designed within DistillerSR.

See Also

[getAuthenticationToken](#)
[getProjects](#)
[getReports](#)

Examples

```
## Not run:
distillerToken_ <- getAuthenticationToken()

projects_ <- getProjects(distillerToken = distillerToken_)

reports_ <- getReports(
  projectId = projects_$id[1],
  distillerToken = distillerToken_)

report_ <- getReport(
  projectId = projects_$id[1],
  reportID = reports_$id[7],
  format = "csv",
  distillerToken = distillerToken_)

## End(Not run)
```

getReportAsync

Submit an asynchronous job to retrieve a Distiller report.

Description

This function submits an asynchronous job to DistillerSR to retrieve a saved report associated with a given project ID. It requires user authentication and a valid asynchronous API endpoint URL. The result is a dataframe containing metadata about the submitted job.

Usage

```
getReportAsync(  
  projectId,  
  reportId,  
  distillerInstanceUrl = Sys.getenv("DISTILLER_INSTANCE_URL"),  
  distillerAsyncInstanceUrl = Sys.getenv("DISTILLER_ASYNC_INSTANCE_URL"),  
  distillerToken,  
  timeout = 1800  
)
```

Arguments

projectId	integer. The ID of the project as provided by DistillerSR.
reportId	integer. The ID of the report as provided by DistillerSR.
distillerInstanceUrl	character (string, optional). The Distiller instance URL. By default: Sys.getenv("DISTILLER_INSTANCE_URL").
distillerAsyncInstanceUrl	character (string, optional). The asynchronous Distiller instance URL. By default: Sys.getenv("DISTILLER_ASYNC_INSTANCE_URL").
distillerToken	character (string). The token the user gets once authenticated.
timeout	integer (optional). The maximum number of seconds to wait for the response. By default: 1800 seconds (30 minutes).

Value

A data frame containing metadata about the submitted job.

See Also

[getAuthenticationToken](#)
[getProjects](#)
[getReports](#)
[getAsyncReportStatus](#)
[getAsyncReportResult](#)

Examples

```
## Not run:  
distillerToken_ <- getAuthenticationToken()  
  
projects_ <- getProjects(distillerToken = distillerToken_)  
  
reports_ <- getReports(  
  projectId = projects_$id[1],  
  distillerToken = distillerToken_)
```

```

job_ <- getReportAsync(
  projectId = projects_$id[1],
  reportID = reports_$id[7],
  format = "csv",
  distillerToken = distillerToken_)

## End(Not run)

```

getReports	<i>Get the list of the Distiller reports associated to a project of the authenticated user.</i>
------------	---

Description

This function queries the DistillerSR API to retrieve the list of saved reports associated with a given project ID. It requires user authentication and a valid API endpoint URL. The result is a dataframe containing metadata about each saved report.

Usage

```

getReports(
  projectId,
  distillerInstanceUrl = Sys.getenv("DISTILLER_INSTANCE_URL"),
  distillerToken,
  timeout = 1800
)

```

Arguments

projectId	integer. The ID of the project as provided by DistillerSR.
distillerInstanceUrl	character (string, optional). The distiller instance URL. By default: Sys.getenv("DISTILLER_INSTANCE_URL").
distillerToken	character (string). The token the user gets once authenticated.
timeout	integer (optional). The maximum number of seconds to wait for the response. By default: 1800 seconds (30 minutes).

Value

A tibble with four columns:

- id: The ID of the saved report.
- name: The name of the report.
- date: The creation date of the report.
- view: The format of the report (e.g., html, csv, excel).

See Also[getAuthenticationToken](#)[getProjects](#)**Examples**

```
## Not run:
distillerToken_ <- getAuthenticationToken()

projects_ <- getProjects(distillerToken = distillerToken_)

reports_ <- getReports(
  projectId = projects_$id[1],
  distillerToken = distillerToken_)

## End(Not run)
```

Index

`getAsyncReportResult`, [2](#), [4](#), [9](#)
`getAsyncReportStatus`, [3](#), [3](#), [9](#)
`getAuthenticationToken`, [3](#), [4](#), [5](#), [6](#), [8](#), [9](#), [11](#)
`getProjects`, [3](#), [4](#), [6](#), [8](#), [9](#), [11](#)
`getReport`, [7](#)
`getReportAsync`, [3](#), [4](#), [8](#)
`getReports`, [3](#), [4](#), [8](#), [9](#), [10](#)