

# Package ‘ojsr’

October 14, 2022

**Type** Package

**Title** Crawler and Data Scraper for Open Journal System ('OJS')

**Version** 0.1.2

**Description** Crawler for 'OJS' pages and scraper for meta-data from articles.

You can crawl 'OJS' archives, issues, articles, galleys, and search results.

You can scrape articles metadata from their head tag in html,

or from Open Archives Initiative ('OAI') records.

Most of these functions rely on 'OJS' routing conventions

(<<https://docs.pkp.sfu.ca/dev/documentation/en/architecture-routes>>).

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Imports** dplyr (>= 0.8.3), magrittr, rvest, tidyr (>= 1.0), urltools,  
xml2, purrr, rlang

**Suggests** knitr, rmarkdown, testthat, tidyverse

**VignetteBuilder** knitr

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Gaston Becerra [aut, cre] (<<https://orcid.org/0000-0001-9432-8848>>)

**Maintainer** Gaston Becerra <gaston.becerra@gmail.com>

**Repository** CRAN

**Date/Publication** 2020-09-23 23:50:03 UTC

## R topics documented:

get_articles_from_issue . . . . .	2
get_articles_from_search . . . . .	3
get_galleys_from_article . . . . .	3
get_html_meta_from_article . . . . .	4
get_issues_from_archive . . . . .	5
get_oai_meta_from_article . . . . .	6

ojsr . . . . .	6
parse_base_url . . . . .	7
parse_oai_url . . . . .	7
<b>Index</b>	<b>9</b>

---

get\_articles\_from\_issue

*Scraping articles URLs from the ToC of OJS issues*

---

## Description

Takes a vector of OJS (issue) URLs and scrapes the links to articles from the issues table of content

## Usage

```
get_articles_from_issue(input_url, verbose = FALSE)
```

## Arguments

input_url	Character vector.
verbose	Logical.

## Value

A long-format dataframe with the url you provided (input\_url) and the articles url scrapped (output\_url)

## Examples

```
issues <- c(
  'https://revistas.ucn.cl/index.php/saludysociedad/issue/view/65',
  'https://publicaciones sociales.uba.ar/index.php/psicologiasocial/issue/view/31'
)
articles <- ojsr::get_articles_from_issue(input_url = issues, verbose = TRUE)
```

---

`get_articles_from_search`*Scraping OJS search results for a given criteria to retrieve articles' URL*

---

**Description**

takes a vector of OJS URLs and a string for search criteria to compose search result URLs, (including pagination) then it scrapes them to retrieve the articles' URLs.

**Usage**

```
get_articles_from_search(input_url, search_criteria, verbose = FALSE)
```

**Arguments**

<code>input_url</code>	Character vector.
<code>search_criteria</code>	Character string
<code>verbose</code>	Logical.

**Value**

A dataframe with the urls of the articles linked from the OJS issue page.

**Examples**

```
journals <- c(
  'https://revistapsicologia.uchile.cl/index.php/RDP/',
  'https://publicaciones.sociales.uba.ar/index.php/psicologiasocial/'
)
criteria <- "actitudes"
search_result_pages <- ojsr::get_articles_from_search(input_url = journals,
  search_criteria = criteria, verbose = TRUE)
```

---

`get_galleys_from_article`*Scraping galleys URLs from OJS articles*

---

**Description**

Takes a vector of OJS URLs and scrapes all the galleys URLs from the article view

**Usage**

```
get_galleys_from_article(input_url, verbose = FALSE)
```

**Arguments**

input_url	Character vector.
verbose	Logical.

**Value**

A long-format dataframe with the url you provided (input\_url), the articles url scrapped (output\_url), the format of the galley (format), and the url that forces download of the galley (download\_url)

**Examples**

```
articles <- c(
  'https://revistapsicologia.uchile.cl/index.php/RDP/article/view/55657',
  'https://dspace.palermo.edu/ojs/index.php/psicodebate/article/view/516/311'
)
galleys <- ojsr::get_galleys_from_article(input_url = articles, verbose = TRUE)
```

---

```
get_html_meta_from_article
```

*Scraping metadata from the OJS articles HTML*

---

**Description**

Takes a vector of OJS URLs and scrapes all metadata written in HTML from the article view

**Usage**

```
get_html_meta_from_article(input_url, verbose = FALSE)
```

**Arguments**

input_url	Character vector.
verbose	Logical.

**Value**

A long-format dataframe with the url you provided (input\_url), the name of the metadata (meta\_data\_name), the content of the metadata (meta\_data\_content), the standard in which the content is annotated (meta\_data\_scheme), and the language in which the metadata was entered (meta\_data\_xmllang)

## Examples

```
articles <- c(
  'https://publicaciones.sociales.uba.ar/index.php/psicologiasocial/article/view/2137', # article
  'https://dspace.palermo.edu/ojs/index.php/psicodebate/article/view/516/311' # xml galley
)
metadata <- ojsr::get_html_meta_from_article(articles, verbose = TRUE)
```

---

get\_issues\_from\_archive

*Scraping issues' URLs from the OJS issues archive*

---

## Description

Takes a vector of OJS URLs and scrapes the issues URLs from the issue archive.

## Usage

```
get_issues_from_archive(input_url, verbose = FALSE)
```

## Arguments

input_url	Character vector.
verbose	Logical.

## Value

A long-format dataframe with the url you provided (input\_url) and the url of issues found (output\_url)

## Examples

```
journals <- c(
  'https://dspace.palermo.edu/ojs/index.php/psicodebate/issue/archive',
  'https://publicaciones.sociales.uba.ar/index.php/psicologiasocial/article/view/2903'
)
issues <- ojsr::get_issues_from_archive(input_url = journals, verbose = TRUE)
```

---

`get_oai_meta_from_article`*Retrieving OAI records for OJS articles*

---

**Description**

This functions access OAI records (within OJS) for any article for which you provided an URL.

**Usage**

```
get_oai_meta_from_article(input_url, verbose = FALSE)
```

**Arguments**

<code>input_url</code>	Character vector.
<code>verbose</code>	Logical.

**Details**

Several limitations are in place. Please refer to vignette.

**Value**

A long-format dataframe with the url you provided (`input_url`), the name of the metadata (`meta_data_name`), and the content of the metadata (`meta_data_content`).

**Examples**

```
articles <- c(
  'https://publicaciones.sociales.uba.ar/index.php/psicologiasocial/article/view/2137', # article
  'https://dspace.palermo.edu/ojs/index.php/psicodebate/article/view/516/311' # xml galley
)
metadata_oai <- ojsr::get_oai_meta_from_article(input_url = articles, verbose = TRUE)
```

---

`ojsr`*ojsr: A package for scraping OJS*

---

**Description**

This package allows you to crawl OJS archives, issues, articles, galleys, and search results, and retrieve metadata from articles. This is useful if OJS Rest API is not available (as in installations prior to v3.1). It also includes function to parse OJS specific URL conventions.

---

parse_base_url	<i>Parses urls against OJS routing conventions and retrieves the base url</i>
----------------	---

---

**Description**

Takes a vector of urls and parses them according to OJS routing conventions, then retrieves OJS base url.

**Usage**

```
parse_base_url(input_url)
```

**Arguments**

input\_url      Character vector.

**Value**

A vector of the same length of your input.

**Examples**

```
mix_links <- c(
  'https://dspace.palermo.edu/ojs/index.php/psicodebate/issue/archive',
  'https://publicaciones.sociales.uba.ar/index.php/psicologiasocial/article/view/2903'
)
base_url <- ojsr::parse_base_url(input_url = mix_links)
```

---

parse_oai_url	<i>Parses urls against OJS routing conventions and retrieves the OAI url</i>
---------------	--

---

**Description**

Takes a vector of urls and parses them according to OJS routing conventions, then retrieves OAI entry url.

**Usage**

```
parse_oai_url(input_url)
```

**Arguments**

input\_url      Character vector.

**Value**

A vector of the same length of your input.

**Examples**

```
mix_links <- c(
  'https://dspace.palermo.edu/ojs/index.php/psicodebate/issue/archive',
  'https://publicaciones.sociales.uba.ar/index.php/psicologiasocial/article/view/2903'
)
oai_url <- ojsr::parse_oai_url(input_url = mix_links)
```



# Index

[get\\_articles\\_from\\_issue](#), 2  
[get\\_articles\\_from\\_search](#), 3  
[get\\_galleys\\_from\\_article](#), 3  
[get\\_html\\_meta\\_from\\_article](#), 4  
[get\\_issues\\_from\\_archive](#), 5  
[get\\_oai\\_meta\\_from\\_article](#), 6

[ojsr](#), 6

[parse\\_base\\_url](#), 7  
[parse\\_oai\\_url](#), 7