Package: spatialCatalogueViewer (via r-universe)

June 2, 2026

Type Package

Title A 'Shiny' Tool to Create Interactive Catalogues for Geospatial Data

Version 0.2.1

Date 2026-04-03

Author Sebastien Plutniak [aut, cre] (<https://orcid.org/0000-0002-6674-3806>)

Maintainer Sebastien Plutniak <sebastien.plutniak@posteo.net>

Description Seamlessly create interactive online catalogues for geospatial data. Items can be mapped as points or areas and retrieved using either a map or a dynamic table with search form and optional column filters.

License GPL-3

Encoding UTF-8

Imports leaflet, leaflet.extras, shiny, shinythemes, DT,

Depends R (>= 4.1.0)

URL https://github.com/sebastien-plutniak/spatialCatalogueViewer

BugReports https://github.com/sebastien-plutniak/spatialCatalogueViewer/issues

Config/pak/sysreqs libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev make libicu-dev libpng-dev libuv1-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev zlib1g-dev

Repository https://sebastien-plutniak.r-universe.dev

Date/Publication 2026-04-03 15:10:53 UTC

RemoteUrl https://github.com/sebastien-plutniak/spatialcatalogueviewer

RemoteRef HEAD

RemoteSha 646fcd800a9a73af6710a9ff10c5cf9e3a750980

Contents

`spatialCatalogueViewer` *spatialCatalogueViewer*

Description

Launch the spatialCatalogueViewer application.

Usage

```
spatialCatalogueViewer(data = NULL,
  text.title = NULL, text.left = NULL, text.top = NULL, text.bottom = NULL,
  map.provider = "Stadia.StamenTerrainBackground",
  map.set.lon = 0, map.set.lat = 0,
  map.legend.variable = NULL, map.legend.labels = NULL,
  map.legend.colors = NULL, map.height = 500,
  map.area.fill.color = "grey", map.area.fill.opacity = 0.5,
  map.show.areas = "never", map.min.zoom = NULL,
  table.hide.columns = NULL, table.filter = "none", table.pageLength = NULL,
  data.download.button = TRUE,
  tabs.contents = NULL,
  theme = "cosmo", css = NULL, js = NULL)
```

Arguments

`data`	dataframe. The data. See details for mandatory columns.
`text.title`	character. HTML text to use for the title.
`text.left`	character. HTML text to use in the columns left to the map.
`text.top`	character. HTML text to use above the map.
`text.bottom`	character. HTML text to use below the map.
`map.provider`	character. Name of the bap provider, see https://leaflet-extras.github.io/leaflet-providers/preview/ for possible values.
`map.set.lon`	integer. Initial longitude to center the map, a value in [-90, 90].
`map.set.lat`	integer. Initial latitude to center the map, a value in [-90, 90].
`map.legend.variable`	character. The name of the column to use for the legend. It also defines the legend title.
`map.legend.labels`	character. A vector with names for the legend's items. Must have the same length than 'map.legend.colors'.
`map.legend.colors`	character. A vector of color names for the legend's item. Must have the same length than 'map.legend.labels'. See colors() for a list of possible values.

`map.height` integer. Height in px of the map.

`map.area.fill.color`
character. A color name to color the areas. See colors() for a list of possible values.

`map.area.fill.opacity`
numerical. Opacity of the areas, a value between 0 and 1.

`map.show.areas` character or logical. One of TRUE, FALSE, 'always', 'never'. If TRUE or FALSE, a checkbox is displayed under the map.

`map.min.zoom` integer. Minimal zoom for the map, between 1 and 12.

`table.hide.columns`
chatacter. Name(s) of the column(s) to hide in the interactive table. Note that these columns can be searched using the 'Search' form anyways.

`table.filter` character. Whether and where to use column filters: 'none': no filters, 'bottom' / 'top': put column filters at the bottom or top of the table.

`table.pageLength`
integer. Number of rows displayed on each page of the table.

`data.download.button`
logical. Dispay or not a button to download the dataset.

`tabs.contents` character. A named list with the contents of the optional additional tabs. List elements' names are used as tab labels.

`theme` character. Name of the shinytheme to use.

`css` character. CSS code.

`js` character. Javascript code.

Details

This function launches the 'spatialCatalogueViewer' application.

The dataframe for `data` has mandatory and optional columns:

Mandatory:

- resource.name: Name of the resource (HTML contents is possible).
- coordinates: Either 'lon' and 'lat' columns or 4 columns named 'bbox.lon1', 'bbox.lat1', 'bbox.lon2', 'bbox.lat2' (where lon1 and lat1 refers to the north-west corner of the area and lon2 and lat2 to the south-east corner).

Optional:

- resource.popup: HTML contents for the map pop-up.

Value

Launch the 'spatialCatalogueViewer' Shiny application.

Author(s)

Sebastien Plutniak <sebastien.plutniak at posteo.net>

See Also

DT leaflet leaflet.extra shiny shinytheme

Examples

```
if(interactive()){
df <- data.frame(lat = c(43.6033, 49.89441, 43.2974),
                 lon = c(1.4444,  2.29649, 5.3703),
                 resource.name = c("Toulouse", "Amiens", "Marseille"),
                 population = c(511684, 135429,  877215),
                 region = c("Occitanie", "Picardie", "Provence")
)
# launch the app:
spatialCatalogueViewer(data = df,
               map.set.lon = 2.37,  map.set.lat =  47.29, map.min.zoom = 5)
}
```

Index