

Package ‘flexsiteboard’

September 7, 2022

Type Package

Title Breaks Single Page Applications from 'flexdashboard' in Multiple Files

Version 0.0.6

Description A drop-in replacement for 'flexdashboard' 'Rmd' documents, which implements an after-knit-hook to split the generated single page application in one document per main section to reduce rendering load in the web browser displaying the document. Put all 'JavaScript' stuff needed in all sections before the first headline featuring navigation menu attributes. This package is experimental and maybe replaced by a solution inside 'flexdashboard'.

License BSD_2_clause + file LICENSE

Encoding UTF-8

Imports flexdashboard, htmltools, htmlwidgets, jsonlite, knitr, rmarkdown, stats, yaml

SystemRequirements pandoc (>= 2.4) - <http://pandoc.org>

RoxygenNote 7.2.1

VignetteBuilder knitr

Suggests withr, testthat (>= 3.0.0), spelling

Config/testthat/edition 3

Language en-US

NeedsCompilation no

Author University Medicine Greifswald [cph],
Stephan Struckmann [aut, cre]

Maintainer Stephan Struckmann <stephan.struckmann@uni-greifswald.de>

Repository CRAN

Date/Publication 2022-09-07 18:40:06 UTC

R topics documented:

flex_site_board	2
pandoc_convert	2
pandoc_input_formats	3
pandoc_output_formats	4
pandoc_read	4
pandoc_write	5

Index	6
--------------	----------

flex_site_board	<i>Flexible Dashboards Split in Sub-Pages</i>
-----------------	---

Description

Flexible Dashboards Split in Sub-Pages

Usage

```
flex_site_board(...)
```

Arguments

... see flexdashbaord
Hint: Needs all global JavaScript code before the first header.

Value

a new format for rmarkdown

pandoc_convert	<i>Convert a Pandoc File</i>
----------------	------------------------------

Description

Convert a Pandoc File

Usage

```
pandoc_convert(
  file,
  text = NULL,
  from = pandoc_input_formats(),
  to = pandoc_output_formats()
)
```

Arguments

file	file to read
text	string to use instead of file if set
from	input format
to	output format

Value

the converted text

Examples

```
if (rmarkdown::pandoc_available()) {  
  x <- pandoc_convert(  
    text = "\\section{Test}", from = "latex", to = "markdown")  
  stopifnot(identical(x, "Test\n====") || identical(x, "# Test"))  
}
```

pandoc_input_formats *Give all Input Formats Supported by Pandoc*

Description

Give all Input Formats Supported by Pandoc

Usage

```
pandoc_input_formats(use_cache = TRUE)
```

Arguments

use_cache	don't call Pandoc again
-----------	-------------------------

Value

all supported Pandoc input formats

pandoc_output_formats *Give all Output Formats Supported by Pandoc*

Description

Give all Output Formats Supported by Pandoc

Usage

```
pandoc_output_formats(use_cache = TRUE)
```

Arguments

use_cache don't call Pandoc again

Value

all supported Pandoc output formats

pandoc_read *Read a Pandoc File from some Supported Format*

Description

Read a Pandoc File from some Supported Format

Usage

```
pandoc_read(file, text = NULL, from = pandoc_input_formats())
```

Arguments

file file to read
text string to use instead of file if set
from format of the file

Value

the Pandoc object

pandoc_write	<i>Write a Pandoc File</i>
--------------	----------------------------

Description

Write a Pandoc File

Usage

```
pandoc_write(pandoc, to = pandoc_output_formats())
```

Arguments

pandoc	internal Pandoc representation
to	output format (supported by Pandoc)

Value

the converted text

Index

`flex_site_board`, [2](#)

`pandoc_convert`, [2](#)

`pandoc_input_formats`, [3](#)

`pandoc_output_formats`, [4](#)

`pandoc_read`, [4](#)

`pandoc_write`, [5](#)