

Package ‘resizableSplitLayout’

May 9, 2026

Title Resizable Split Layout Module for 'shiny'

Version 0.1.1

Description A 'shiny' module to facilitate page layouts with resizable panes for page content based on 'split.js' 'JavaScript' library (<<https://split.js.org>>).

License GPL (>= 3)

Imports shiny, shinyjs

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.3.3

NeedsCompilation no

Author Michael Schaffer [aut, cre] (ORCID:
<<https://orcid.org/0000-0003-1381-0275>>)

Maintainer Michael Schaffer <mschaff@gmail.com>

Repository CRAN

Date/Publication 2025-12-09 07:50:02 UTC

Contents

resizableSplitDemo	2
resizableSplitUI	2
Index	4

resizableSplitDemo *Run Example App for resizableSplitLayout*

Description

This function launches a simple Shiny application demonstrating how to use the `resizableSplitLayout` module.

Usage

```
resizableSplitDemo()
```

Value

No return value, called for side effects (launches a Shiny app)

Examples

```
# Run this function to see the module in action
if (interactive()) {
  resizableSplitDemo()
}
```

resizableSplitUI *resizableSplitUI*

Description

resizableSplitUI

Usage

```
resizableSplitUI(
  id,
  sidebarpane,
  maincontenttop,
  maincontentbottom,
  sidebar.percentage = 15,
  maincontenttop.percentage = 85,
  return.only.taglist = FALSE
)
```

Arguments

<code>id</code>	Module identifier
<code>sidebarpane</code>	Content for page sidebar
<code>maincontenttop</code>	Content for page main top pane
<code>maincontentbottom</code>	Content for page main bottom pane
<code>sidebar.percentage</code>	Width for sidebar as percentage
<code>maincontenttop.percentage</code>	Height for page main top pane as percentage
<code>return.only.taglist</code>	logical to return taglist instead of a bootstrapPage

Value

A Shiny UI object. If `return.only.taglist = TRUE`, returns a `shiny.tag.list` object containing the resizable split layout elements. If `return.only.taglist = FALSE` (default), returns a Shiny bootstrap page (from `shiny::bootstrapPage`) containing the resizable layout.

Examples

```
if (interactive()) {  
  
  ui <- resizableSplitUI("splitModule",  
    sidebarpane = shiny::div("This is the sidebar"),  
    maincontenttop = shiny::div("This is the maincontenttop"),  
    maincontentbottom = shiny::div("This is the maincontentbottom")  
  )  
  
  server <- function(input, output, session) { }  
  
  shinyApp(ui, server)  
}
```

Index

`resizableSplitDemo`, [2](#)
`resizableSplitUI`, [2](#)