

Package: designer (via r-universe)

August 30, 2024

Title 'Shiny' UI Prototype Builder

Version 0.3.0

Description A 'shiny' application that enables the user to create a prototype UI, being able to drag and drop UI components before being able to save or download the equivalent R code.

License MIT + file LICENSE

URL <https://ashbaldry.github.io/designer/>,
<https://ashbaldry.shinyapps.io/designer/>

BugReports <https://github.com/ashbaldry/designer/issues>

Imports bs4Dash, bslib, cicerone, config (>= 0.3.1), fontawesome, golem (>= 0.3.1), htmltools, jsonlite, rappdirs, shinipsum, shiny (>= 1.7.1), shinscreenshot

Suggests chromote, knitr, lintr, rmarkdown, shinytest2, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Language en-GB

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Repository <https://ashbaldry.r-universe.dev>

RemoteUrl <https://github.com/ashbaldry/designer>

RemoteRef HEAD

RemoteSha 4d9c6b6a6b5abe388d5161dc8656e3956a7459bf

Contents

component	2
compSettingTag	2
designApp	4
sidebarItem	5

Index**6**

component	<i>Component Settings Shell</i>
-----------	---------------------------------

Description

A container for the specified component input

Usage

```
component(id, ...)
```

```
componentTab(id)
```

Arguments

id	The ID of the component input
...	Shiny tags to include inside the component

Details

The tab component contains a selection of specific inputs related to adding a new tab, as the events to create it in the UI are different to the other components

Value

A shiny.tag of the component settings

compSettingTag	<i>Bootstrap Component Inputs</i>
----------------	-----------------------------------

Description

A way to be able to adjust components so that can more easily visualise how the shiny application will look.

Usage

```
compSettingTag(id, choices = NULL)
```

```
compSettingType(id, choices)
```

```
compSettingPlot(id)
```

```
compSettingValue(id)
```

```
compSettingLabel(id, label = "Label", optional = FALSE)
compSettingID(id)
compSettingIcon(id)
compSettingColour(id, status = FALSE)
compSettingBackground(id)
compSettingFill(id, label = "Fill Whole Box")
compSettingText(id, value = NULL)
compSettingTextArea(id)
compSettingChoices(id)
compSettingRange(id)
compSettingInline(id)
compSettingDownload(id)
compSettingWidth(id)
compSettingHeight(id)
compSettingWidthNum(id, value = 3L, min = 1L)
compSettingOffset(id)
```

Arguments

<code>id</code>	Namespace to include the component
<code>choices</code>	A vector of potential choices to include in the component
<code>label</code>	Label of the input
<code>optional</code>	Logical, is the input optional?
<code>status</code>	Logical, are only status colours allowed, default is FALSE
<code>value</code>	Value given to the component input
<code>min</code>	Minimum value given to the component input

Value

A shiny.tag.list of settings specific to the selected component

See Also[component](#)

`designApp`

Run the Shiny Application

Description

Runs the designer Shiny application.

For more information about how the application works, either run the "Help" guide in-app, or run `vignette("designer")`.

Usage

```
designApp(  
  onStart = NULL,  
  options = list(),  
  enableBookmarking = "url",  
  uiPattern = "/",  
  ...  
)
```

Arguments

- | | |
|--------------------------------|--|
| <code>onStart</code> | A function that will be called before the app is actually run. This is only needed for <code>shinyAppObj</code> , since in the <code>shinyAppDir</code> case, a global <code>.R</code> file can be used for this purpose. |
| <code>options</code> | Named options that should be passed to the <code>runApp</code> call (these can be any of the following: "port", "launch.browser", "host", "quiet", "display.mode" and "test.mode"). You can also specify width and height parameters which provide a hint to the embedding environment about the ideal height/width for the app. |
| <code>enableBookmarking</code> | Can be one of "url", "server", or "disable". The default value, <code>NULL</code> , will respect the setting from any previous calls to enableBookmarking() . See enableBookmarking() for more information on bookmarking your app. |
| <code>uiPattern</code> | A regular expression that will be applied to each GET request to determine whether the ui should be used to handle the request. Note that the entire request path must match the regular expression in order for the match to be considered successful. |
| <code>...</code> | arguments to pass to <code>golem_opts</code> . See get_golem_options for more details. |

Value

This function does not return a value; interrupt R to stop the application (usually by pressing `Ctrl+C` or `Esc`).

Examples

```
designApp()
```

sidebarItem

Component Accordion Item

Description

An item to add to the sidebar that opens up the settings for the selected component

Usage

```
sidebarItem(id, name, element, parent_id, ..., notes = NULL, active = FALSE)
```

Arguments

id	HTML ID to namespace on
name	Label to show on the closed accordion
element	Character string to let JS know what component has been chosen
parent_id	HTML ID of the accordion
...	Option inputs to add when expanding the accordion item
notes	A list of optional notes to include at the bottom of the settings
active	Logical, should the accordion item be open on start? Default set to FALSE

Value

A shiny.tag element containing the component accordion item with all input settings

Index

component, [2](#), [4](#)
componentTab (component), [2](#)
compSettingBackground (compSettingTag),
[2](#)
compSettingChoices (compSettingTag), [2](#)
compSettingColour (compSettingTag), [2](#)
compSettingDownload (compSettingTag), [2](#)
compSettingFill (compSettingTag), [2](#)
compSettingHeight (compSettingTag), [2](#)
compSettingIcon (compSettingTag), [2](#)
compSettingID (compSettingTag), [2](#)
compSettingInline (compSettingTag), [2](#)
compSettingLabel (compSettingTag), [2](#)
compSettingOffset (compSettingTag), [2](#)
compSettingPlot (compSettingTag), [2](#)
compSettingRange (compSettingTag), [2](#)
compSettingTag, [2](#)
compSettingText (compSettingTag), [2](#)
compSettingTextArea (compSettingTag), [2](#)
compSettingType (compSettingTag), [2](#)
compSettingValue (compSettingTag), [2](#)
compSettingWidth (compSettingTag), [2](#)
compSettingWidthNum (compSettingTag), [2](#)

designApp, [4](#)

enableBookmarking(), [4](#)

get_golem_options, [4](#)

sidebarItem, [5](#)