Package 'shinyscreenshot'

August 21, 2023

Title Capture Screenshots of Entire Pages or Parts of Pages in 'Shiny'

Version 0.2.1

Description Capture screenshots in 'Shiny' applications. Screenshots can either be of the entire viewable page, or a specific section of the page. The captured image is automatically downloaded as a PNG image, or it can also be saved on the server. Powered by the 'html2canvas' JavaScript library.

URL <https://github.com/daattali/shinyscreenshot>,

<https://daattali.com/shiny/shinyscreenshot-demo/>

BugReports <https://github.com/daattali/shinyscreenshot/issues>

Depends R (>= 3.1.0)

Imports base64enc, htmltools (>= 0.3.5), jsonlite, shiny (>= 1.0.0), uuid

Suggests timevis

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

NeedsCompilation no

Author Dean Attali [aut, cre] (<<https://orcid.org/0000-0002-5645-3493>>),
Niklas von Hertzen [aut] (html2canvas library),
Eli Grey [aut] (FileSaver library)

Maintainer Dean Attali <daattali@gmail.com>

Repository CRAN

Date/Publication 2023-08-21 08:20:06 UTC

R topics documented:

---

runExample   *Run shinyscreenshot example*

Description

Launch an example Shiny app that shows how easy it is to take screenshots with shinyscreenshot.

The demo app is also available online to experiment with.

Usage

```
runExample()
```

---

screenshot   *Capture a screenshot of a shiny app*

Description

Screenshots can be either of the entire viewable page (default), or of a specific section of the page. The captured image is automatically downloaded as a PNG image.

This function gets called from the server portion of a Shiny app, unlike screenshotButton() which is similar but gets called from the UI.

Usage

```
screenshot(
  selector = "body",
  filename = "shinyscreenshot",
  id = "",
  scale = 1,
  timer = 0,
  download = TRUE,
  server_dir = NULL
)
```

Arguments

selector	CSS selector for the element that should be captured. If multiple elements match the selector, only the first one is captured. Default is to capture the entire page.
filename	Name of the file to be saved. A PNG extension will automatically be added.
id	As an alternative to selector, an ID of the element that should be captured can be provided. If id is provided, then selector is ignored. When used in a module, the id **does not** need to be namespaced (namespacing is automatic).

scale	The scale of the image. Default is 1, which means the dimensions of the image will be exactly the dimensions in the browser. For example, a value of 2 will result in an image that's twice the height and width (and a larger file size).
timer	Number of seconds to wait before taking the screenshot. Default is 0, which takes a screenshot immediately.
download	If TRUE (default), download the screenshot image to the user's computer. If FALSE, the image isn't downloaded to the user, and a server_dir should be specified.
server_dir	Directory on the server where the screenshot image should be saved. See 'Saving to the server' section below.

Saving to the server

By default, the image is downloaded to the user's computer and is not stored on the server running the Shiny app. If a server_dir is provided, then the image is stored to this directory on the server. Note that only the directory should be specified, not the file name.

If saving the image is successful, input$shinyscreenshot will contain the full path to the image. If not, input$shinyscreenshot will contain an empty string ("").

The directory must exist and be writeable. If NULL, the image is not saved to the server. If a relative path is provided, it is relative to the Shiny app's working directory. For example, server_dir="." will save the image in the same directory that the Shiny app is in.

See Also

[screenshotButton()](screenshotButton)

Examples

```
if (interactive()) {
  library(shiny)
  library(shinyscreenshot)

  shinyApp(
    ui = fluidPage(
      h1("{shinyscreenshot} demo"),
      numericInput("num", "Number of points", 50),
      plotOutput("plot"),
      actionButton("screenshot1", "Capture entire page"),
      actionButton("screenshot2", "Capture plot")
    ),
    server = function(input, output) {
      observeEvent(input$screenshot1, {
        screenshot()
      })
      observeEvent(input$screenshot2, {
        screenshot(id = "plot")
      })
      output$plot <- renderPlot({
        plot(runif(input$num))
```

```
    })
  }
 )
}
```

---

screenshotButton   *Button that captures a screenshot of a shiny app*

Description

Create a button that, when clicked, captures a screenshot of the Shiny app. Screenshots can be either of the entire viewable page (default), or of a specific section of the page. The captured image is automatically downloaded as a PNG image.

This function gets called from the UI portion of a Shiny app, unlike screenshot() which is similar but gets called from the server.

Usage

```
screenshotButton(
  selector = "body",
  filename = "shinyscreenshot",
  id = "",
  scale = 1,
  timer = 0,
  download = TRUE,
  server_dir = NULL,
  ns = shiny::NS(NULL),
  ...
)
```

Arguments

selector	CSS selector for the element that should be captured. If multiple elements match the selector, only the first one is captured. Default is to capture the entire page.
filename	Name of the file to be saved. A PNG extension will automatically be added.
id	As an alternative to selector, an ID of the element that should be captured can be provided. If id is provided, then selector is ignored. When used in a module, the id **does** need to be namespaced, like any other UI element.
scale	The scale of the image. Default is 1, which means the dimensions of the image will be exactly the dimensions in the browser. For example, a value of 2 will result in an image that's twice the height and width (and a larger file size).
timer	Number of seconds to wait before taking the screenshot. Default is 0, which takes a screenshot immediately.
download	If TRUE (default), download the screenshot image to the user's computer. If FALSE, the image isn't downloaded to the user, and a server_dir should be specified.

server_dir	Directory on the server where the screenshot image should be saved. See 'Saving to the server' section below.
ns	The [namespace](#) object of the current module if inside a Shiny module.
...	Any other parameters that should be passed along to the [shiny::actionButton()](#).

Saving to the server

By default, the image is downloaded to the user's computer and is not stored on the server running the Shiny app. If a server_dir is provided, then the image is stored to this directory on the server. Note that only the directory should be specified, not the file name.

If saving the image is successful, input$shinyscreenshot will contain the full path to the image. If not, input$shinyscreenshot will contain an empty string ("").

The directory must exist and be writeable. If NULL, the image is not saved to the server. If a relative path is provided, it is relative to the Shiny app's working directory. For example, server_dir="." will save the image in the same directory that the Shiny app is in.

See Also

[screenshot()](#)

Examples

```
if (interactive()) {
  library(shiny)
  library(shinyscreenshot)

  shinyApp(
    ui = fluidPage(
      h1("{shinyscreenshot} demo"),
      screenshotButton(label = "Capture entire page"),
      screenshotButton(label = "Capture plot", id = "plot"), br(), br(),
      numericInput("num", "Number of points", 50),
      plotOutput("plot")
    ),
    server = function(input, output) {
      output$plot <- renderPlot({
        plot(runif(input$num))
      })
    }
  )
}
```

Index