Package 'picClip'

July 23, 2025

Type Package

Title Paste Box Input for 'Shiny'

Version 0.1.0

Author Matt Deppe

Maintainer Matt Deppe <deppemj@gmail.com>

Description Provides a 'Shiny' input widget, pasteBoxInput, that allows users to paste images directly into a 'Shiny' application. The pasted images are captured as Base64 encoded strings and can be used within the application for various purposes, such as display or further processing. This package is particularly useful for applications that require easy and quick image uploads without the need for traditional file selection dialog boxes.

License GPL-3

URL https://github.com/deppemj/picClip

Encoding UTF-8

Imports shiny, base64enc, stringr, htmltools, testthat

RoxygenNote 7.2.3

NeedsCompilation no

Repository CRAN

Date/Publication 2023-11-28 18:10:05 UTC

Contents

pasteBoxInput *Paste Box Input*

Description

Create a paste box input control for images.

Usage

```
pasteBoxInput(inputId, label, width = "100px", height = "100px")
```

Arguments

inputId	The input slot that will be used to access the value.
label	Display label for the control.
width	The width of the paste box, e.g., '100px'.
height	The height of the paste box, e.g., '100px'.

Value

A Shiny tag list that creates a UI element for pasting images.

Examples

```
if (interactive()) {
  library(shiny)
  library(base64enc)

  ui <- fluidPage(
    pasteBoxInput("testInput", "Paste Image Here", "300px", "150px"),
    uiOutput("image")
  )

  server <- function(input, output, session) {

  #This example is to show how to render the image directly back to the user
    observeEvent(input$testInput, {
      if (!is.null(input$testInput) && input$testInput != "") {
        output$image <- renderUI({
          tags$img(src = input$testInput, style = "max-width: 100%; height: auto;")
        })
      }
    })

    #This example is to show how to save the image, in this case to a temp file.
    observeEvent(input$testInput, {
      if (!is.null(input$testInput) && input$testInput != "") {
```

```
        if (grepl("^data:image", input$testInput)) {
          base64_string <- sub("", "", input$testInput)
          }
          decoded_data <- base64decode(base64_string)

          temp_file_name <- tempfile(fileext = ".png")

          writeBin(as.raw(decoded_data), temp_file_name)
          }

          })

  }
  shinyApp(ui, server)
}
```

Index