

Package ‘shinykanban’

February 3, 2025

Title Create Kanban Board in Shiny Applications

Version 0.0.1

Description Provides an interactive Kanban board widget for 'shiny' applications.

It allows users to manage tasks using a drag-and-drop interface and offers customizable styling options.

'shinykanban' is ideal for project management, task tracking, and agile workflows within 'shiny' apps.

License MIT + file LICENSE

URL <https://github.com/ugurdar/shinykanban>

BugReports <https://github.com/ugurdar/shinykanban/issues>

Encoding UTF-8

RoxygenNote 7.3.2

Imports htmlwidgets, reactR, shiny, bsicons, htmltools

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Ugur Dar [aut, cre],
Brian Pillmore [aut, cph]

Maintainer Ugur Dar <ugurdarr@gmail.com>

Repository CRAN

Date/Publication 2025-02-03 18:10:02 UTC

Contents

getSelectedCard	2
kanban	2
kanban-shiny	3
updateKanban	5
Index	6

getSelectedCard	<i>Get the Selected Card Data</i>
-----------------	-----------------------------------

Description

Retrieves the details of a card that was clicked on the Kanban board.

Usage

```
getSelectedCard(outputId, session = NULL)
```

Arguments

outputId	A character string specifying the ID of the Kanban output.
session	The Shiny session object.

Value

A list with the selected card's details as a list (listName, title, id, position, clickCount)

kanban	<i>Create a Kanban Board Widget</i>
--------	-------------------------------------

Description

This function creates an interactive Kanban board as an HTML widget.

Usage

```
kanban(
  data,
  styleOptions = list(headerBg = "#fff", headerBgHover = "#fff", headerColor = "#353535",
    headerFontSize = "1rem", listNameFontSize = "1rem", cardTitleFontSize = "1rem",
    cardTitleFontWeight = 600, cardSubTitleFontSize = "0.8rem", cardSubTitleFontWeight =
    300, addCardBgColor = "#999", deletelist = list(background-color = "#fff", color =
    "#353535", icon = bsicons::bs_icon("x"), size = "1rem"), deleteCard =
    list(background-color = "#fff", color = "#353535", icon = bsicons::bs_icon("trash"),
    size = "1rem"), addButtonText = "Add",
    cancelButtonText = "Cancel",
    addCardButtonText = "Add Card", cancelCardButtonText = "Cancel"),
  width = NULL,
  height = NULL,
  elementId = NULL
)
```

Arguments

data	A named list representing the board data.
styleOptions	A named list of style options.
width, height	Optional widget dimensions.
elementId	DOM element ID.

Value

A kanban board.

kanban-shiny	<i>Shiny bindings for Kanban Board</i>
--------------	--

Description

Output and render functions for using Kanban Board within Shiny.

Usage

```
kanbanOutput(outputId, width = "100%", height = "400px")
renderKanban(expr, env = parent.frame(), quoted = FALSE)
```

Arguments

outputId	Output variable to read the value from
width, height	A valid CSS unit (like "100%", "400px", "auto") or a number, which will be coerced to a string and have "px" appended.
expr	An expression that generates kanban board with shinykanban::kanban()
env	The parent environment for the reactive expression.
quoted	If it is TRUE, then the quote()ed value of expr will be used when expr is evaluated.

Value

kanbanOutput() returns a kanban output element that can be included in a Shiny UI.

renderKanban() returns a kanban render function that can be assigned to a Shiny output slot.

Examples

```
if(interactive()){
  library(shiny)
  library(shinykanban)
  library(bsicons)

  ui <- fluidPage(
    kanbanOutput("kanban_board")
  )

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

    kanban_data <- reactiveVal(
      list(
        "To Do" = list(
          name = "To Do",
          items = list(
            list(
              id = "task1",
              title = "Task 1",
              subtitle = "abc"
            ),
            list(
              id = "task2",
              title = "Task 2"
            )
          ),
          listPosition = 1
        ),
        "In Progress" = list(
          name = "In Progress",
          items = list(
            list(
              id = "task3",
              title = "Task 3"
            )
          ),
          listPosition = 2
        )
      ))

    output$kanban_board <- renderKanban({
      kanban(data = kanban_data())
    })

    # Get any change from kanban and update the data
    observeEvent(input$kanban_board, {
      new_list <- input$kanban_board
      new_list$`_timestamp` <- NULL
      kanban_data(new_list)
    })
  }
}
```

```
shinyApp(ui, server)
}
```

updateKanban	<i>Update the data for a Kanban input on the client.</i>
--------------	--

Description

Update the data for a Kanban input on the client.

Usage

```
updateKanban(session, inputId, data)
```

Arguments

session	The Shiny session object.
inputId	The ID of the input object.
data	The data to set.

Value

None

Index

`getSelectedCard`, [2](#)

`kanban`, [2](#)

`kanban-shiny`, [3](#)

`kanbanOutput` (`kanban-shiny`), [3](#)

`renderKanban` (`kanban-shiny`), [3](#)

`updateKanban`, [5](#)