

# Package ‘bs4Dash’

November 27, 2019

**Type** Package

**Title** A 'Bootstrap 4' Version of 'shinydashboard'

**Version** 0.5.0

**Maintainer** David Granjon <dgranjon@ymail.com>

**Description** Make 'Bootstrap 4' dashboards. Use the full power of 'AdminLTE3', a dashboard template built on top of 'Bootstrap 4' <<https://github.com/ColorlibHQ/AdminLTE>>.

**URL** <https://rinterface.github.io/bs4Dash/index.html>,  
<https://github.com/RinterFace/bs4Dash>

**BugReports** <https://github.com/RinterFace/bs4Dash/issues>

**License** GPL (>= 2) | file LICENSE

**Imports** shiny, htmltools

**Suggests** plotly, shinyWidgets, knitr, rmarkdown

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.0.0

**VignetteBuilder** knitr

**Collate** 'update-functions.R' 'render-functions.R' 'cards.R'  
'dashboardSidebar.R' 'dashboardBody.R' 'dashboardFooter.R'  
'dashboardControlbar.R' 'dashboardNavbar.R' 'dashboardPage.R'  
'aliases.R' 'bs4DashGallery.R' 'deps.R' 'grid.R'  
'useful-items.R' 'utils.R'

**NeedsCompilation** no

**Author** David Granjon [aut, cre],  
RinterFace [cph],  
Almasaeed Studio [ctb, cph] (AdminLTE3 theme for Bootstrap 4),  
Winston Chang [ctb, cph] (Utils functions from shinydashboard),  
Thomas Park [ctb, cph] (Bootstrap Sketchy theme CSS)

**Repository** CRAN

**Date/Publication** 2019-11-27 10:20:02 UTC

**R topics documented:**

attachmentBlock . . . . .	4
bs4Accordion . . . . .	5
bs4AccordionItem . . . . .	6
bs4Alert . . . . .	7
bs4Badge . . . . .	8
bs4Box . . . . .	9
bs4Callout . . . . .	11
bs4Card . . . . .	13
bs4Carousel . . . . .	17
bs4CarouselItem . . . . .	18
bs4DashBody . . . . .	19
bs4DashControlbar . . . . .	19
bs4DashFooter . . . . .	20
bs4DashGallery . . . . .	21
bs4DashNavbar . . . . .	21
bs4DashPage . . . . .	22
bs4DashSidebar . . . . .	26
bs4DropdownMenu . . . . .	27
bs4DropdownMenuItem . . . . .	29
bs4InfoBox . . . . .	30
bs4InsertTab . . . . .	32
bs4Jumbotron . . . . .	34
bs4ListGroup . . . . .	35
bs4ListGroupItem . . . . .	37
bs4Loading . . . . .	38
bs4ProgressBar . . . . .	39
bs4Quote . . . . .	40
bs4RemoveTab . . . . .	41
bs4Ribbon . . . . .	43
bs4SidebarHeader . . . . .	44
bs4SidebarMenu . . . . .	45
bs4SidebarMenuItem . . . . .	47
bs4SidebarMenuSubItem . . . . .	48
bs4SidebarUserPanel . . . . .	50
bs4SocialCard . . . . .	51
bs4Sortable . . . . .	53
bs4Stars . . . . .	54
bs4TabCard . . . . .	55
bs4TabItem . . . . .	58
bs4TabItems . . . . .	58
bs4Table . . . . .	59
bs4TableItem . . . . .	60
bs4TableItems . . . . .	61
bs4TabPanel . . . . .	61
bs4TabSetPanel . . . . .	62
bs4Timeline . . . . .	64

bs4TimelineEnd . . . . .	66
bs4TimelineItem . . . . .	66
bs4TimelineItemMedia . . . . .	67
bs4TimelineLabel . . . . .	68
bs4TimelineStart . . . . .	68
bs4UserCard . . . . .	69
bs4UserMenu . . . . .	70
bs4ValueBox . . . . .	72
bs4ValueBoxOutput . . . . .	74
cardComment . . . . .	74
cardPad . . . . .	75
cardProfile . . . . .	76
cardProfileItem . . . . .	78
cardProfileItemList . . . . .	78
column . . . . .	79
controlBarItem . . . . .	80
controlbarMenu . . . . .	80
descriptionBlock . . . . .	81
dropdownDivider . . . . .	83
dropdownItem . . . . .	83
dropdownItemList . . . . .	84
getAdminLTEColors . . . . .	84
infoBoxOutput . . . . .	85
ionicon . . . . .	85
renderbs4ValueBox . . . . .	86
renderInfoBox . . . . .	88
updatebs4Card . . . . .	88
updatebs4Controlbar . . . . .	90
updatebs4Sidebar . . . . .	91
updatebs4TabItems . . . . .	92
updatebs4TabSetPanel . . . . .	94
updateControlbarMenu . . . . .	97
userMessage . . . . .	98
userMessages . . . . .	99
userPost . . . . .	100
userPostMedia . . . . .	101
userPostTagItem . . . . .	102
userPostTagItems . . . . .	103

---

attachmentBlock	<i>AdminLTE3 attachment container</i>
-----------------	---------------------------------------

---

### Description

Create an attachment container, nice to wrap articles...

### Usage

```
attachmentBlock(..., src = NULL, title = NULL, title_url = NULL)
```

### Arguments

...	Any element.
src	Url or path to the image.
title	Attachment title.
title_url	External link.

### Author(s)

David Granjon, <dgranjon@gmail.com>

### Examples

```
if (interactive()) {  
  library(shiny)  
  library(bs4Dash)  
  shinyApp(  
    ui = bs4DashPage(  
      bs4DashNavbar(),  
      bs4DashSidebar(),  
      bs4DashControlbar(),  
      bs4DashBody(  
        bs4Card(  
          title = "Attachment example",  
          attachmentBlock(  
            src = "https://adminlte.io/themes/dev/AdminLTE/dist/img/photo1.png",  
            title = "Test",  
            title_url = "http://google.com",  
            "This is the content"  
          )  
        )  
      ),  
      title = "AttachmentBlock"  
    ),  
    server = function(input, output) { }  
  )  
}
```

---

`bs4Accordion`*Create a Bootstrap 4 accordion*

---

## Description

Beautiful accordion from AdminLTE3

## Usage

```
bs4Accordion(..., id)
```

## Arguments

<code>...</code>	Slot for <a href="#">bs4AccordionItem</a> .
<code>id</code>	Unique accordion id.

## Author(s)

David Granjon, <dgranjon@ymail.com>

## Examples

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        bs4Accordion(
          id = "accordion",
          bs4AccordionItem(
            id = "item1",
            title = "Item 1",
            status = "danger",
            "Anim pariatur cliche reprehenderit, enim
            eiusmod high life accusamus terry richardson ad
            squid. 3 wolf moon officia aute, non cupidatat
            skateboard dolor brunch. Food truck quinoa nesciunt
            laborum eiusmod. Brunch 3 wolf moon tempor, sunt
            aliqua put a bird on it squid single-origin coffee
            nulla assumenda shoreditch et. Nihil anim keffiyeh
            helvetica, craft beer labore wes anderson cred
            nesciunt sapiente ea proident. Ad vegan excepteur
```

```

    butcher vice lomo. Leggings occaecat craft beer farm-to-table,
    raw denim aesthetic synth nesciunt you probably haven't
    heard of them accusamus labore sustainable VHS"
  ),
  bs4AccordionItem(
    id = "item2",
    title = "Item 2",
    status = "warning",
    "Anim pariatur cliche reprehenderit, enim
    eiusmod high life accusamus terry richardson ad
    squid. 3 wolf moon officia aute, non cupidatat
    skateboard dolor brunch. Food truck quinoa nesciunt
    laborum eiusmod. Brunch 3 wolf moon tempor, sunt
    aliqua put a bird on it squid single-origin coffee
    nulla assumenda shoreditch et. Nihil anim keffiyeh
    helvetica, craft beer labore wes anderson cred
    nesciunt sapiente ea proident. Ad vegan excepteur
    butcher vice lomo. Leggings occaecat craft beer farm-to-table,
    raw denim aesthetic synth nesciunt you probably haven't
    heard of them accusamus labore sustainable VHS"
  )
)
)
),
server = function(input, output) {}
)
}

```

---

bs4AccordionItem

*Create a Bootstrap 4 accordion item*


---

## Description

To insert in a bs4Accordion

## Usage

```
bs4AccordionItem(..., id, title = NULL, status = NULL, width = 12)
```

## Arguments

...	Item content.
id	Item unique id.
title	Item title.
status	Item color. "primary", "success", "warning", "danger" or "info". NULL by default.
width	The width of the accordion.

**Author(s)**

David Granjon, <dgranjon@gmail.com>

---

bs4Alert

*Create a Bootstrap 4 alert*

---

**Description**

AdminLTE3 alert

**Usage**

```
bs4Alert(  
  ...,  
  title,  
  closable = TRUE,  
  width = 6,  
  elevation = NULL,  
  status = c("primary", "warning", "danger", "info", "success")  
)
```

**Arguments**

...	Alert content.
title	Alert title.
closable	Whether to allow the user to close the alert. FALSE by default.
width	Alert width. Between 1 and 12.
elevation	Alert elevation.
status	Alert status. "primary", "success", "warning", "danger" or "info".

**Author(s)**

David Granjon, <dgranjon@gmail.com>

**Examples**

```
if(interactive()){  
  library(shiny)  
  library(bs4Dash)  
  
  shiny::shinyApp(  
    ui = bs4DashPage(  
      navbar = bs4DashNavbar(),  
      sidebar = bs4DashSidebar(),  
      controlbar = bs4DashControlbar(),  
      footer = bs4DashFooter(),
```

```

    title = "test",
    body = bs4DashBody(
      title = "Alerts",
      bs4Alert(
        title = "Be Careful!",
        status = "danger",
        closable = FALSE,
        "Danger alert preview. This alert is not dismissable.
        A wonderful serenity has taken possession of my entire soul,
        like these sweet mornings of spring which
        I enjoy with my whole heart."
      ),
      bs4Alert(
        title = "Congratulation!",
        status = "success",
        closable = TRUE,
        elevation = 4,
        "Warning alert preview. This alert is dismissable.
        A wonderful serenity has taken possession of my entire soul,
        like these sweet mornings of spring which
        I enjoy with my whole heart."
      )
    ),
  ),
  server = function(input, output) {}
)
}

```

---

 bs4Badge

*Create a Bootstrap 4 dashboard badge item*


---

## Description

Beautiful badge from AdminLTE3

## Usage

```
bs4Badge(..., position = c("left", "right"), status, rounded = FALSE)
```

## Arguments

...	Badge content.
position	Badge position: "left" or "right".
status	Badge color. "primary", "danger", "info", "success", "warning", "secondary", "dark" or "light".
rounded	Whether the badge is rounded instead of square. FALSE by default.



**Author(s)**

David Granjon, <dgranjon@ymail.com>

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        bs4Badge(
          position = "right",
          status = "warning",
          "Warning"
        )
      )
    ),
    server = function(input, output) {}
  )
}
```

---

bs4Box

*AdminLTE3 simple box*

---

**Description**

Create nice and epurated box

**Usage**

```
bs4Box(..., title = NULL, width = 6, height = NULL)
```

**Arguments**

...	Body content.
title	Box title.
width	The width of the box, using the Bootstrap grid system.
height	The height of a box, in pixels or other CSS unit. By default the height scales automatically with the content.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(
        bs4SidebarMenu(
          bs4SidebarHeader("Main content"),
          bs4SidebarMenuItem(
            "Basic boxes",
            tabName = "boxes",
            icon = "desktop"
          )
        )
      ),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        bs4TabItems(
          bs4TabItem(
            tabName = "boxes",
            fluidRow(
              bs4Box(
                height = "600px",
                title = "Box 1",
                plotOutput("plot"),
                column(
                  width = 12,
                  align = "center",
                  sliderInput(
                    "obs",
                    "Number of observations:",
                    min = 0, max = 1000,
                    value = 500
                  )
                )
              )
            ),
          bs4Box(
            height = "600px",
            title = "Box 2",
            plotOutput("distPlot"),
            column(
              width = 12,
              align = "center",
```

```

        radioButtons(
          "dist",
          inline = TRUE,
          "Distribution type:",
          c("Norm" = "norm",
            "Unif" = "unif",
            "LogNorm" = "lnorm",
            "Exp" = "exp")
        )
      )
    )
  )
),
server = function(input, output) {
  output$plot <- renderPlot({
    hist(rnorm(input$obs))
  })

  output$distPlot <- renderPlot({
    dist <- switch(
      input$dist,
      norm = rnorm,
      unif = runif,
      lnorm = rlnorm,
      exp = rexp,
      rnorm
    )

    hist(dist(500))
  })
}
}

```

---

bs4Callout

*Create a Bootstrap 4 callout*


---

### Description

AdminLTE3 callout

### Usage

```

bs4Callout(
  ...,
  title,

```

```

width = 6,
elevation = NULL,
status = c("primary", "warning", "danger", "info", "success")
)

```

### Arguments

...	Callout content.
title	Callout title.
width	Callout width. Between 1 and 12.
elevation	Callout elevation.
status	Callout status. "primary", "success", "warning", "danger" or "info".

### Author(s)

David Granjon, <dgranjon@gmail.com>

### Examples

```

if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        title = "Callouts",
        bs4Callout(
          title = "I am a danger callout!",
          elevation = 4,
          status = "danger",
          "There is a problem that we need to fix.
          A wonderful serenity has taken possession of
          my entire soul, like these sweet mornings of
          spring which I enjoy with my whole heart."
        ),
        bs4Callout(
          title = "I am a danger callout!",
          status = "warning",
          "This is a yellow callout."
        )
      )
    ),
    server = function(input, output) {}
  )
}

```

---

`bs4Card`*Create a Bootstrap 4 card*

---

**Description**

Build an adminLTE3 card

**Usage**

```
bs4Card(  
    ...,  
    inputId = NULL,  
    title = NULL,  
    footer = NULL,  
    status = NULL,  
    elevation = NULL,  
    solidHeader = FALSE,  
    headerBorder = TRUE,  
    gradientColor = NULL,  
    width = 6,  
    height = NULL,  
    collapsible = TRUE,  
    collapsed = FALSE,  
    closable = TRUE,  
    maximizable = FALSE,  
    labelStatus = NULL,  
    labelText = NULL,  
    labelTooltip = NULL,  
    dropdownMenu = NULL,  
    dropdownIcon = "wrench",  
    overflow = FALSE,  
    enable_sidebar = FALSE,  
    sidebar_content = NULL,  
    sidebar_width = "25%",  
    sidebar_background = "#333a40",  
    sidebar_start_open = FALSE,  
    sidebar_icon = "cogs"  
)  
  
box(  
    ...,  
    inputId = NULL,  
    title = NULL,  
    footer = NULL,  
    status = NULL,
```

```

elevation = NULL,
solidHeader = FALSE,
headerBorder = TRUE,
gradientColor = NULL,
width = 6,
height = NULL,
collapsible = TRUE,
collapsed = FALSE,
closable = TRUE,
maximizable = FALSE,
labelStatus = NULL,
labelText = NULL,
labelTooltip = NULL,
dropdownMenu = NULL,
dropdownIcon = "wrench",
overflow = FALSE,
enable_sidebar = FALSE,
sidebar_content = NULL,
sidebar_width = "25%",
sidebar_background = "#333a40",
sidebar_start_open = FALSE,
sidebar_icon = "cogs"
)

```

### Arguments

...	Contents of the box.
inputId	Get the state of the card. Optional.
title	Optional title.
footer	Optional footer text.
status	The status of the card header. "primary", "secondary", "success", "warning", "danger", "white", "light", "dark", "transparent". NULL by default.
elevation	Card elevation.
solidHeader	Should the header be shown with a solid color background?
headerBorder	Whether to display a border between the header and body. TRUE by default
gradientColor	If NULL (the default), the background of the box will be white. Otherwise, a color string. "primary", "success", "warning" or "danger".
width	The width of the box, using the Bootstrap grid system. This is used for row-based layouts. The overall width of a region is 12, so the default width of 4 occupies 1/3 of that width. For column-based layouts, use NULL for the width; the width is set by the column that contains the box.
height	The height of a box, in pixels or other CSS unit. By default the height scales automatically with the content.
collapsible	If TRUE, display a button in the upper right that allows the user to collapse the box.

<code>collapsed</code>	If TRUE, start collapsed. This must be used with <code>collapsible=TRUE</code> .
<code>closable</code>	If TRUE, display a button in the upper right that allows the user to close the box.
<code>maximizable</code>	If TRUE, the card can be displayed in full screen mode.
<code>labelStatus</code>	status of the box label: "primary", "secondary", "success", "warning", "danger", "white", "light", "dark", "transparent".
<code>labelText</code>	Label text.
<code>labelTooltip</code>	Label tooltip displayed on hover.
<code>dropdownMenu</code>	List of items in the the boxtool dropdown menu. Use <a href="#">dropdownItemList</a> .
<code>dropdownIcon</code>	Dropdown icon. "wrench" by default.
<code>overflow</code>	Whether to enable overflow in the card body and footer. FALSE by default.
<code>enable_sidebar</code>	Whether to display the box sidebar. FALSE by default.
<code>sidebar_content</code>	Box sidebar content, if any.
<code>sidebar_width</code>	Box sidebar width in percentage. 25% by default. A character value of any width CSS understands (e.g. "100px")
<code>sidebar_background</code>	Box sidebar background color. Dark by default.
<code>sidebar_start_open</code>	Whether the box sidebar is open at start. FALSE by default.
<code>sidebar_icon</code>	Box sidebar icon.

**Author(s)**

David Granjon, <dgranjon@gmail.com>

**See Also**

Other cards: [bs4InfoBox\(\)](#), [bs4TabCard\(\)](#), [bs4ValueBox\(\)](#)

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        fluidRow(
          column(
            width = 6,
```

```

bs4Card(
  title = "Closable Box with dropdown",
  closable = TRUE,
  width = 12,
  status = "warning",
  solidHeader = FALSE,
  collapsible = TRUE,
  labelText = 1,
  labelStatus = "danger",
  labelTooltip = "Hi Bro!",
  dropdownIcon = "wrench",
  dropdownMenu = dropdownItemList(
    dropdownItem(url = "http://www.google.com", name = "Link to google"),
    dropdownItem(url = "#", name = "item 2"),
    dropdownDivider(),
    dropdownItem(url = "#", name = "item 3")
  ),
  p("Box Content")
),
column(
  width = 6,
  bs4Card(
    title = "Closable Box with gradient",
    closable = TRUE,
    width = 12,
    status = "warning",
    solidHeader = FALSE,
    gradientColor = "success",
    collapsible = TRUE,
    p("Box Content")
  )
),
fluidRow(
  bs4Card(
    title = "Closable Box with solidHeader",
    closable = TRUE,
    width = 6,
    solidHeader = TRUE,
    status = "primary",
    collapsible = TRUE,
    p("Box Content")
  ),
  bs4Card(
    title = "Maximizable Card",
    width = 6,
    status = "warning",
    closable = FALSE,
    maximizable = TRUE,
    collapsible = FALSE,
    sliderInput("obs", "Number of observations:",
      min = 0, max = 1000, value = 500
    )
  )
)

```



```
    ),
    plotOutput("distPlot")
  )
)
),
server = function(input, output) {
  output$distPlot <- renderPlot({
    hist(rnorm(input$obs))
  })
}
)
```

---

bs4Carousel

*Create a Bootstrap 4 carousel*

---

## Description

Beautiful carousel from AdminLTE3

## Usage

```
bs4Carousel(..., id, width = 12)
```

## Arguments

...	Slot for <a href="#">bs4CarouselItem</a> .
id	Unique carousel id.
width	Carousel width. Between 1 and 12.

## Author(s)

David Granjon, <dgranjon@gmail.com>

## Examples

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
```

```
body = bs4DashBody(  
  title = "Carousel",  
  bs4Carousel(  
    id = "mycarousel",  
    width = 6,  
    bs4CarouselItem(  
      active = TRUE,  
      src = "http://placeholder.it/900x500/39CCCC/ffffff&text=I+Love+Bootstrap"  
    ),  
    bs4CarouselItem(  
      active = FALSE,  
      src = "http://placeholder.it/900x500/3c8dbc/ffffff&text=I+Love+Bootstrap"  
    ),  
    bs4CarouselItem(  
      active = FALSE,  
      src = "http://placeholder.it/900x500/f39c12/ffffff&text=I+Love+Bootstrap"  
    )  
  )  
)  
)  
,  
server = function(input, output) {}  
)  
}
```

---

bs4CarouselItem

*Create a Bootstrap 4 carousel item*

---

## Description

To insert in a [bs4Carousel](#)

## Usage

```
bs4CarouselItem(active = FALSE, src = NULL)
```

## Arguments

active	Whether the item is active or not at start.
src	Item path or url.

## Author(s)

David Granjon, <dgranjon@ymail.com>

---

bs4DashBody	<i>Create a Bootstrap 4 dashboard body</i>
-------------	--

---

**Description**

Build an adminLTE3 dashboard body

**Usage**

```
bs4DashBody(...)
```

```
dashboardBody(...)
```

**Arguments**

...            Body content, slot for [bs4TabItems](#).

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

bs4DashControlbar	<i>Create a Bootstrap 4 dashboard right sidebar</i>
-------------------	---

---

**Description**

Build an adminLTE3 dashboard right sidebar

**Usage**

```
bs4DashControlbar(  
  ...,  
  inputId = NULL,  
  disable = FALSE,  
  skin = "dark",  
  title = NULL,  
  width = 250  
)
```

```
dashboardControlbar(  
  ...,  
  inputId = NULL,  
  disable = FALSE,  
  skin = "dark",  
  title = NULL,  
  width = 250  
)
```

**Arguments**

...	Any UI element.
inputId	To access the current state of the controlbar. Open is TRUE, closed is FALSE. NULL by default.
disable	If TRUE, the sidebar will be disabled.
skin	Controlbar skin. "dark" or "light".
title	Controlbar title.
width	Controlbar width. 250 px by default.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

bs4DashFooter

*Create a Bootstrap 4 dashboard footer*

---

**Description**

Build an adminLTE3 dashboard footer

**Usage**

```
bs4DashFooter(..., copyrights = NULL, right_text = NULL)
```

```
dashboardFooter(..., copyrights = NULL, right_text = NULL)
```

**Arguments**

...	Left text.
copyrights	Copyrights, if any.
right_text	Right text.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

bs4DashGallery	<i>Launch the bs4Dash Gallery</i>
----------------	-----------------------------------

---

**Description**

A gallery of all components available in bs4Dash.

**Usage**

```
bs4DashGallery(theme = "classic")
```

**Arguments**

theme            bs4Dash theme. Either "classic" or "old\_school". "classic" by default.

**Examples**

```
if (interactive()) {  
  bs4DashGallery()  
}
```

---

bs4DashNavbar	<i>Create a Bootstrap 4 dashboard navbar</i>
---------------	--

---

**Description**

Build an adminLTE3 dashboard navbar

**Usage**

```
bs4DashNavbar(  
  ...,  
  skin = "light",  
  status = NULL,  
  border = TRUE,  
  sidebarIcon = "bars",  
  compact = FALSE,  
  controlbarIcon = "th",  
  leftUi = NULL,  
  rightUi = NULL  
)  
  
dashboardHeader(  
  ...,  
  skin = "light",  
  status = NULL,  
  border = TRUE,  
  sidebarIcon = "bars",  
  compact = FALSE,  
  controlbarIcon = "th",  
  leftUi = NULL,  
  rightUi = NULL  
)
```

```

    ...,
    skin = "light",
    status = NULL,
    border = TRUE,
    sidebarIcon = "bars",
    compact = FALSE,
    controlbarIcon = "th",
    leftUi = NULL,
    rightUi = NULL
  )

```

### Arguments

...	Any UI element between left and right Ui.
skin	Navbar skin. "dark" or "light".
status	Navbar status. "primary", "danger", "warning", "success", "info", "white", "gray-light" and all other available colors. See <a href="#">getAdminLTEColors</a> .
border	Whether to separate the navbar and body by a border. TRUE by default.
sidebarIcon	Icon of the main sidebar toggle.
compact	Whether items should be compacted. FALSE by default.
controlbarIcon	Icon to toggle the controlbar (left).
leftUi	Custom left Ui content. Any Ui element.
rightUi	Custom right Ui content. Any Ui element.

### Author(s)

David Granjon, <dgranjon@ymail.com>

---

bs4DashPage

*Create a Bootstrap 4 dashboard page*

---

### Description

Build an adminLTE3 dashboard page

### Usage

```

bs4DashPage(
  navbar = NULL,
  sidebar = NULL,
  body = NULL,
  controlbar = NULL,
  footer = NULL,
  title = NULL,
  old_school = FALSE,

```

```

    sidebar_mini = TRUE,
    sidebar_collapsed = FALSE,
    controlbar_collapsed = TRUE,
    controlbar_overlay = TRUE,
    enable_preloader = FALSE,
    loading_duration = 2,
    loading_background = "#1E90FF"
)

dashboardPage(
  navbar = NULL,
  sidebar = NULL,
  body = NULL,
  controlbar = NULL,
  footer = NULL,
  title = NULL,
  old_school = FALSE,
  sidebar_mini = TRUE,
  sidebar_collapsed = FALSE,
  controlbar_collapsed = TRUE,
  controlbar_overlay = TRUE,
  enable_preloader = FALSE,
  loading_duration = 2,
  loading_background = "#1E90FF"
)

```

### Arguments

navbar	Slot for <a href="#">bs4DashNavbar</a> .
sidebar	Slot for <a href="#">bs4DashSidebar</a> .
body	Slot for <a href="#">bs4DashBody</a> .
controlbar	Slot for <a href="#">bs4DashControlbar</a> (right side).
footer	Slot for <a href="#">bs4DashFooter</a> .
title	App title.
old_school	Whether to use the wonderful sketchy design for Bootstrap 4. FALSE by default.
sidebar_mini	Whether to see sidebar icons when <a href="#">bs4DashSidebar</a> is collapsed. TRUE by default.
sidebar_collapsed	Whether the sidebar is collapsed of not at start. FALSE by default.
controlbar_collapsed	Whether the sidebar is collapsed of not at start. TRUE by default.
controlbar_overlay	Whether the sidebar covers the content when expanded. Default to TRUE.
enable_preloader	Whether to enable a page loader. FALSE by default.

loading\_duration

Loader duration in seconds. 2s by default.

loading\_background

Background color during page startup. Blue by default: [https://www.w3schools.com/cssref/css\\_colors.asp](https://www.w3schools.com/cssref/css_colors.asp).

### Author(s)

David Granjon, <dgranjon@ymail.com>

### Examples

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      enable_preloader = TRUE,
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody()
    ),
    server = function(input, output) {}
  )
}

if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = dashboardPage(
      navbar = dashboardHeader(),
      sidebar = dashboardSidebar(),
      body = dashboardBody(
        fluidRow(
          column(width = 4, plotOutput("distPlot")),
          column(width = 4, tableOutput("data")),
          column(width = 4, textOutput("result"))
        )
      ),
    controlbar = dashboardControlbar(
      skin = "light",
      tabsetPanel(
        type = "tabs",
        id = "tabsetpanel",
        tabPanel(
          title = "Tab 1",
```





---

`bs4DashSidebar`*Create a Bootstrap 4 dashboard main sidebar*

---

### Description

Build an adminLTE3 dashboard main sidebar

### Usage

```
bs4DashSidebar(  
  ...,  
  inputId = NULL,  
  disable = FALSE,  
  title = NULL,  
  skin = "dark",  
  status = "primary",  
  brandColor = NULL,  
  url = NULL,  
  src = NULL,  
  elevation = 4,  
  opacity = 0.8,  
  expand_on_hover = TRUE  
)
```

```
dashboardSidebar(  
  ...,  
  inputId = NULL,  
  disable = FALSE,  
  title = NULL,  
  skin = "dark",  
  status = "primary",  
  brandColor = NULL,  
  url = NULL,  
  src = NULL,  
  elevation = 4,  
  opacity = 0.8,  
  expand_on_hover = TRUE  
)
```

### Arguments

<code>...</code>	Slot for <a href="#">bs4SidebarMenu</a> .
<code>inputId</code>	Recover the state of the sidebar.
<code>disable</code>	If TRUE, the sidebar will be disabled.
<code>title</code>	Sidebar title.
<code>skin</code>	Sidebar skin. "dark" or "light"

status	Sidebar status. "primary", "danger", "warning", "success", "info".
brandColor	Brand color. NULL by default: "primary", "danger", "warning", "success", "info", "white" or "gray-light".
url	Sidebar brand link.
src	Sidebar brand image.
elevation	Sidebar elevation. 4 by default (until 5).
opacity	Sidebar brand opacity. From 0 to 1. 0.8 by default.
expand_on_hover	Whether to expand the sidebar on hover. TRUE by default.

**Author(s)**

David Granjon, <dgranjon@gmail.com>

---

bs4DropdownMenu

*Create a Bootstrap 4 dashboard dropdown menu*

---

**Description**

Build an adminLTE3 dashboard dropdown menu

**Usage**

```
bs4DropdownMenu(  
  ...,  
  show = FALSE,  
  labelText = NULL,  
  src = NULL,  
  status = c("primary", "warning", "danger", "info", "success"),  
  menuIcon = "bell",  
  align = "right"  
)  
  
dropdownMenu(  
  ...,  
  show = FALSE,  
  labelText = NULL,  
  src = NULL,  
  status = c("primary", "warning", "danger", "info", "success"),  
  menuIcon = "bell",  
  align = "right"  
)
```

**Arguments**

...	Slot for <a href="#">bs4DropdownMenuItem</a> .
show	Whether to start with the dropdown open. FALSE by default.
labelText	Dropdown label text.
src	Dropdown link to an external ressource.
status	Dropdown menu status. "primary", "success", "warning", "danger" or "info".
menuItem	Fontawesome icon (default = "bell")
align	Menu alignment (default = "right")

**Author(s)**

David Granjon, <dgranjon@ymail.com>

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(
        rightUi = bs4DropdownMenu(
          show = TRUE,
          labelText = "!",
          status = "danger",
          src = "http://www.google.fr",
          bs4DropdownMenuItem(
            text = "message 1",
            date = "today"
          ),
          bs4DropdownMenuItem(
            text = "message 2",
            date = "yesterday"
          )
        )
      ),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody()
    ),
    server = function(input, output) {}
  )
}
```

---

bs4DropdownMenuItem    *Create a Bootstrap 4 dashboard dropdown menu item*

---

### Description

Build an adminLTE3 dashboard dropdown menu item

### Usage

```
bs4DropdownMenuItem(  
    message,  
    from = NULL,  
    time = NULL,  
    icon = "info-circle",  
    src = NULL,  
    status = NULL,  
    type = c("notification", "message")  
)
```

```
dropdownMenuItem(  
    message,  
    from = NULL,  
    time = NULL,  
    icon = "info-circle",  
    src = NULL,  
    status = NULL,  
    type = c("notification", "message")  
)
```

### Arguments

message	Item content.
from	Item sender. Only if type is message.
time	Item date.
icon	Item icon.
src	Item image. Only if type is message.
status	Item status. Leave NULL if type is notification. See <a href="#">getAdminLTEColors</a> for valid statuses. If not NULL, Indicate the message priority.
type	Item type: notification or message.

### Author(s)

David Granjon, <dgranjon@ymail.com>

---

`bs4InfoBox`*Bootstrap 4 info box*

---

### Description

A beautiful AdminLTE3 info box.

### Usage

```
bs4InfoBox(  
    ...,  
    tabName = NULL,  
    title,  
    value = NULL,  
    icon = NULL,  
    iconElevation = 3,  
    status = NULL,  
    gradientColor = NULL,  
    width = 4,  
    elevation = NULL  
)
```

```
infoBox(  
    ...,  
    tabName = NULL,  
    title,  
    value = NULL,  
    icon = NULL,  
    iconElevation = 3,  
    status = NULL,  
    gradientColor = NULL,  
    width = 4,  
    elevation = NULL  
)
```

### Arguments

<code>...</code>	Any extra UI element.
<code>tabName</code>	Optional: <a href="#">bs4InfoBox</a> may be used to navigate between tabs.
<code>title</code>	Info box title.
<code>value</code>	The value to display in the box. Usually a number or short text.
<code>icon</code>	An icon tag, created by <a href="#">icon</a> .
<code>iconElevation</code>	Icon elevation compared to the main content (relief). 3 by default.
<code>status</code>	A color for the box. "primary", "info", "success", "warning", "danger" or NULL.

gradientColor	If NULL (the default), the background of the box will be white. Otherwise, a color string. "primary", "success", "warning" or "danger".
width	The width of the box, using the Bootstrap grid system. This is used for row-based layouts. The overall width of a region is 12, so the default width of 4 occupies 1/3 of that width. For column-based layouts, use NULL for the width; the width is set by the column that contains the box.
elevation	Infobox elevation.

**Author(s)**

David Granjon, <dgranjon@gmail.com>

**See Also**

Other cards: [bs4Card\(\)](#), [bs4TabCard\(\)](#), [bs4ValueBox\(\)](#)

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        fluidRow(
          bs4InfoBox(
            title = "Messages",
            value = 1410,
            icon = "envelope"
          ),
          bs4InfoBox(
            title = "Bookmarks",
            status = "info",
            value = 240,
            icon = "bookmark"
          ),
          bs4InfoBox(
            title = "Comments",
            gradientColor = "danger",
            value = 41410,
            icon = "comments"
          )
        )
      )
    ),
  )
},
```

```

    server = function(input, output) {}
  )
}

```

---

bs4InsertTab

*Insert a [bs4TabPanel](#) in a [bs4TabSetPanel](#)*


---

## Description

Insert a [bs4TabPanel](#) in a [bs4TabSetPanel](#)

## Usage

```

bs4InsertTab(
  inputId,
  tab,
  target,
  position = c("before", "after"),
  select = FALSE,
  session = shiny::getDefaultReactiveDomain()
)

```

## Arguments

inputId	<a href="#">bs4TabSetPanel</a> id.
tab	<a href="#">bs4TabPanel</a> to insert.
target	<a href="#">bs4TabPanel</a> after of before which the new tab will be inserted.
position	Insert before or after: c("before", "after").
select	Whether to select the newly inserted tab. FALSE by default.
session	Shiny session object.

## Examples

```

if (interactive()) {
  library(shiny)
  library(bs4Dash)

  ui <- bs4DashPage(
    sidebar_collapsed = T,
    sidebar = bs4DashSidebar(),
    bs4DashFooter(),
    body = bs4DashBody(
      actionButton("add1", "ADD tabset 1"),
      bs4TabSetPanel(
        id = "tabset1",
        side = "left",

```



```

    bs4TabPanel(
      tabName = "Tab 1",
      active = TRUE,
      p("Text 1"),
    ),
    bs4TabPanel(
      tabName = "Tab 2",
      active = FALSE,
      p("Text 2"),
    )
  ),
  actionButton("add2", "ADD tabset 2"),
  bs4TabSetPanel(
    id = "tabset2",
    side = "left",
    bs4TabPanel(
      tabName = "Tab 1",
      active = TRUE,
      p("Text 1"),
    ),
    bs4TabPanel(
      tabName = "Tab 2",
      active = FALSE,
      p("Text 2"),
    )
  )
)
)
)

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

  observeEvent(input$add1, {
    bs4InsertTab(
      inputId = "tabset1",
      bs4TabPanel(tabName = "Dynamic", "I am inserted"),
      target = "Tab 1",
      position = "after",
      select = FALSE
    )
  })

  observeEvent(input$add2, {
    bs4InsertTab(
      inputId = "tabset2",
      bs4TabPanel(tabName = "Dynamic", "I am inserted and active"),
      target = "Tab 1",
      position = "before",
      select = TRUE
    )
  })

}
shinyApp(ui, server)

```

```
}

```

---

```
bs4Jumbotron
```

```
BS4 jumbotron for AdminLTE3
```

---

## Description

Create a jumbotron

## Usage

```
bs4Jumbotron(
  ...,
  title = NULL,
  lead = NULL,
  href = NULL,
  btn_name = "More",
  status = c("primary", "warning", "danger", "info", "success")
)
```

## Arguments

...	Any content.
title	Jumbotron title.
lead	Jumbotron lead.
href	Jumbrottron external link.
btn_name	Jumbotron button name.
status	Jumbotron background color. "primary", "success", "warning", "danger" or "info".

## Author(s)

David Granjon, <dgranjon@ymail.com>

## Examples

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
```

```
body = bs4DashBody(  
  bs4Jumbotron(  
    title = "Hello, world!",  
    lead = "This is a simple hero unit, a simple jumbotron-style  
    component for calling extra attention to featured  
    content or information.",  
    "It uses utility classes for typography and spacing  
    to space content out within the larger container.",  
    status = "primary",  
    href = "http://www.google.fr"  
  )  
)  
,  
server = function(input, output) {}  
)  
}
```

---

bs4ListGroup

*BS4 list group for AdminLTE3*

---

## Description

Create a list group

## Usage

```
bs4ListGroup(..., width = 4)
```

## Arguments

...	Slot for <a href="#">bs4ListGroupItem</a> .
width	List group width. 4 by default. Between 1 and 12.

## Author(s)

David Granjon, <dgranjon@ymail.com>

## Examples

```
if(interactive()){  
  library(shiny)  
  library(bs4Dash)  
  
  shiny::shinyApp(  
    ui = bs4DashPage(  
      navbar = bs4DashNavbar(),  
      sidebar = bs4DashSidebar(),  
      controlbar = bs4DashControlbar(),
```

```

footer = bs4DashFooter(),
title = "test",
body = bs4DashBody(
  fluidRow(
    bs4ListGroup(
      bs4ListGroupItem(
        type = "basic",
        "Cras justo odio"
      ),
      bs4ListGroupItem(
        type = "basic",
        "Dapibus ac facilisis in"
      ),
      bs4ListGroupItem(
        type = "basic",
        "Morbi leo risus"
      )
    ),
    bs4ListGroup(
      bs4ListGroupItem(
        "Cras justo odio",
        active = TRUE,
        disabled = FALSE,
        type = "action",
        src = "http://www.google.fr"
      ),
      bs4ListGroupItem(
        active = FALSE,
        disabled = FALSE,
        type = "action",
        "Dapibus ac facilisis in",
        src = "http://www.google.fr"
      ),
      bs4ListGroupItem(
        "Morbi leo risus",
        active = FALSE,
        disabled = TRUE,
        type = "action",
        src = "http://www.google.fr"
      )
    ),
    bs4ListGroup(
      bs4ListGroupItem(
        "Donec id elit non mi porta gravida at eget metus.
        Maecenas sed diam eget risus varius blandit.",
        active = TRUE,
        disabled = FALSE,
        type = "heading",
        title = "List group item heading",
        subtitle = "3 days ago",
        footer = "Donec id elit non mi porta."
      ),
      bs4ListGroupItem(

```

```

        "Donec id elit non mi porta gravida at eget metus.
        Maecenas sed diam eget risus varius blandit.",
        active = FALSE,
        disabled = FALSE,
        type = "heading",
        title = "List group item heading",
        subtitle = "3 days ago",
        footer = "Donec id elit non mi porta."
    )
)
)
)
),
server = function(input, output) {}
)
}

```

---

bs4ListGroupItem

*BS4 list group item for AdminLTE3*


---

## Description

Create a list group item

## Usage

```

bs4ListGroupItem(
  ...,
  active = FALSE,
  disabled = FALSE,
  type = c("basic", "action", "heading"),
  src = "#",
  title = NULL,
  subtitle = NULL,
  footer = NULL
)

```

## Arguments

...	Item content.
active	Whether the item is active or not. FALSE by default. Only if type is "action" or "heading".
disabled	Whether the item is disabled or not. FALSE by default. Only if type is "action" or "heading".
type	Item type. Choose between "basic", "action" and "heading".
src	Item external link.

title	Item title (only if type is "heading").
subtitle	Item subtitle (only if type is "heading").
footer	Item footer content (only if type is "heading").

**Author(s)**

David Granjon, <dgranjon@gmail.com>

---

bs4Loading

*AdminLTE3 loading state element*

---

**Description**

When a section is still work in progress or a computation is running

**Usage**

bs4Loading()

**Note**

Loading state can be programmatically used when a computation is running for instance.

**Author(s)**

David Granjon, <dgranjon@gmail.com>

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        title = "Callouts",
        bs4Card(
          title = "Loading State",
          bs4Loading()
        )
      )
    ),
    server = function(input, output) {}
  )
}
```

```
)  
}
```

---

bs4ProgressBar

*Create a Bootstrap 4 progress bar*

---

## Description

AdminLTE3 progress bar

## Usage

```
bs4ProgressBar(  
  value,  
  min = 0,  
  max = 100,  
  vertical = FALSE,  
  striped = FALSE,  
  status = c("primary", "warning", "danger", "info", "success"),  
  size = NULL  
)
```

## Arguments

value	Progress bar value
min	Progress bar minimum value.
max	Progress bar maximum value.
vertical	Whether to display the progress bar in vertical mode. FALSE by default.
striped	Whether the progress bar is striped or not. FALSE by default.
status	Progress bar status. "primary", "success", "warning", "danger" or "info".
size	Progress bar size. NULL, "sm", "xs" or "xxs".

## Author(s)

David Granjon, <dgranjon@ymail.com>

## Examples

```
if(interactive()){  
  library(shiny)  
  library(bs4Dash)  
  
  shiny::shinyApp(  
    ui = bs4DashPage(  
      navbar = bs4DashNavbar(),  
      sidebar = bs4DashSidebar(),
```

```
controlbar = bs4DashControlbar(),
footer = bs4DashFooter(),
title = "test",
body = bs4DashBody(
  title = "Progress bars",
  bs4Card(
    title = "Progress bars",
    footer = tagList(
      bs4ProgressBar(
        value = 5,
        size = "xxs",
        striped = FALSE,
        status = "info"
      ),
      bs4ProgressBar(
        value = 25,
        striped = TRUE,
        status = "warning"
      )
    ),
    bs4ProgressBar(
      value = 80,
      vertical = TRUE,
      status = "success"
    ),
    bs4ProgressBar(
      value = 100,
      vertical = TRUE,
      striped = TRUE,
      size = "sm",
      status = "danger"
    )
  )
),
server = function(input, output) {}
}
```

---

bs4Quote

*Create a Bootstrap 4 block quote*

---

### **Description**

Build a bootstrap 4 block quote

### **Usage**

bs4Quote(..., status)



**Arguments**

...	Content.
status	Block status: "primary", "danger", "success", "warning", "info" and "secondary" or any other supported colors. See <a href="https://adminlte.io/themes/dev/AdminLTE/pages/UI/general.html">https://adminlte.io/themes/dev/AdminLTE/pages/UI/general.html</a> .

**Author(s)**

David Granjon, <dgranjon@gmail.com>

**Examples**

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)
  shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      body = bs4DashBody(
        fluidRow(
          bs4Quote("Blablabla", status = "indigo"),
          bs4Quote("Blablabla", status = "danger"),
          bs4Quote("Blablabla", status = "teal"),
          bs4Quote("Blablabla", status = "orange"),
          bs4Quote("Blablabla", status = "warning"),
          bs4Quote("Blablabla", status = "fuchsia")
        )
      ),
      footer = bs4DashFooter()
    ),
    server = function(input, output) { }
  )
}
```

---

 bs4RemoveTab

*Remove a [bs4TabPanel](#) in a [bs4TabSetPanel](#)*


---

**Description**

Remove a [bs4TabPanel](#) in a [bs4TabSetPanel](#)

**Usage**

```
bs4RemoveTab(inputId, target, session = shiny::getDefaultReactiveDomain())
```

**Arguments**

`inputId` [bs4TabSetPanel](#) id.  
`target` [bs4TabPanel](#) to remove.  
`session` Shiny session object.

**Examples**

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)

  ui <- bs4DashPage(
    sidebar_collapsed = TRUE,
    sidebar = bs4DashSidebar(),
    bs4DashFooter(),
    body = bs4DashBody(
      actionButton("remove1", "Remove tab 1"),
      bs4TabSetPanel(
        id = "tabset1",
        side = "left",
        bs4TabPanel(
          tabName = "Tab 1",
          active = TRUE,
          p("Text 1"),
        ),
        bs4TabPanel(
          tabName = "Tab 2",
          active = FALSE,
          p("Text 2"),
        )
      ),
      actionButton("remove2", "Remove tab 2"),
      bs4TabSetPanel(
        id = "tabset2",
        side = "left",
        bs4TabPanel(
          tabName = "Tab 1",
          active = TRUE,
          p("Text 1"),
        ),
        bs4TabPanel(
          tabName = "Tab 2",
          active = FALSE,
          p("Text 2"),
        )
      )
    )
  )
)

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

```
observeEvent(input$remove1, {
  bs4RemoveTab(
    inputId = "tabset1",
    target = "Tab 1"
  )
})

observeEvent(input$remove2, {
  bs4RemoveTab(
    inputId = "tabset2",
    target = "Tab 2",
  )
})

}
shinyApp(ui, server)
}
```

---

bs4Ribbon

*Create a Bootstrap 4 ribbon*

---

## Description

Build a bootstrap 4 ribbon

## Usage

```
bs4Ribbon(
  text,
  status = c("primary", "danger", "success", "warning", "info", "secondary"),
  size = NULL
)
```

## Arguments

text	Ribbon text.
status	Ribbon status: "primary", "danger", "success", "warning", "info" and "secondary".
size	NULL by default: "lg" or "xl".

## Author(s)

David Granjon, <dgranjon@gmail.com>

**Examples**

```
if (interactive()) {  
  library(shiny)  
  library(bs4Dash)  
  shinyApp(  
    ui = bs4DashPage(  
      navbar = bs4DashNavbar(),  
      sidebar = bs4DashSidebar(),  
      body = bs4DashBody(  
        fluidRow(  
          bs4Box(  
            width = 4,  
            title = "Normal ribbon",  
            bs4Ribbon(  
              text = "New",  
              status = "primary"  
            )  
          ),  
          bs4Box(  
            width = 4,  
            title = "Large ribbon",  
            bs4Ribbon(  
              text = "New",  
              status = "secondary",  
              size = "lg"  
            )  
          ),  
          bs4Box(  
            width = 4,  
            title = "XLarge ribbon",  
            bs4Ribbon(  
              text = "New",  
              status = "danger",  
              size = "xl"  
            )  
          )  
        )  
      ),  
      footer = bs4DashFooter()  
    ),  
    server = function(input, output) { }  
  )  
}
```

**Description**

Build an adminLTE3 dashboard main sidebar header

**Usage**

```
bs4SidebarHeader(title)
```

```
sidebarHeader(title)
```

**Arguments**

title            SidebarHeader title.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

bs4SidebarMenu	<i>Create a Bootstrap 4 dashboard main sidebar menu</i>
----------------	---

---

**Description**

Build an adminLTE3 dashboard main sidebar menu

**Usage**

```
bs4SidebarMenu(
  ...,
  id = NULL,
  flat = FALSE,
  compact = FALSE,
  child_indent = TRUE
)
```

```
sidebarMenu(..., id = NULL, flat = FALSE, compact = FALSE, child_indent = TRUE)
```

**Arguments**

...	Slot for <a href="#">bs4SidebarMenuItem</a> or <a href="#">bs4SidebarHeader</a> .
id	For <a href="#">bs4SidebarMenu</a> , if id is present, this id will be used for a Shiny input value, and it will report which tab is selected. For example, if id="tabs", then input\$tabs will be the tabName of the currently-selected <a href="#">bs4SidebarMenuItem</a> .
flat	Whether sidebar items should have a flat design. FALSE by default.
compact	Whether items should be compacted. FALSE by default.
child_indent	Whether to indent children. TRUE by default

**Author(s)**

David Granjon, <dgranjon@ymail.com>

**Examples**

```

if (interactive()) {
  library(shiny)
  library(bs4Dash)

  shinyApp(
    ui = bs4DashPage(
      sidebar_collapsed = TRUE,
      controlbar_collapsed = TRUE,
      enable_preloader = FALSE,
      loading_duration = 2,
      navbar = bs4DashNavbar(skin = "light"),
      body = bs4DashBody(

        ),
      sidebar = bs4DashSidebar(
        skin = "light",
        bs4SidebarMenu(
          id = "test",
          bs4SidebarMenuItem(
            tabName = "tab1",
            text = "Tab 1"
          ),
          bs4SidebarMenuItem(
            tabName = "tab2",
            text = "Tab 2"
          ),
          bs4SidebarMenuItem(
            text = "Click me pleaaaaaase",
            bs4SidebarMenuSubItem(
              tabName = "subtab1",
              text = "Tab 3"
            ),bs4SidebarMenuSubItem(
              tabName = "subtab2",
              text = "Tab 4"
            )
          )
        )
      ),
      controlbar = bs4DashControlbar(skin = "light"),
      footer = bs4DashFooter()
    ),
    server = function(input, output, session) {
      observeEvent(input$test, {
        if (input$test == "subtab1") {
          showModal(modalDialog(
            title = "Thank you so much",
            "You clicked me! This event is the result of

```

```
        an input bound to the menu. By adding an id to the
        bs4SidebarMenu, input$id will give the currently selected
        tab. This is useful to trigger some events.",
        easyClose = TRUE,
        footer = NULL
    ))
  }
})
}
)
```

---

bs4SidebarMenuItem	<i>Create a Bootstrap 4 dashboard main sidebar menu item</i>
--------------------	--

---

## Description

Build an adminLTE3 dashboard main sidebar menu item

## Usage

```
bs4SidebarMenuItem(
  text,
  ...,
  tabName = NULL,
  icon = NULL,
  startExpanded = FALSE
)
```

```
menuItem(text, ..., tabName = NULL, icon = NULL, startExpanded = FALSE)
```

## Arguments

text	Item name.
...	<a href="#">bs4SidebarMenuSubItem</a> .
tabName	Should correspond exactly to the tabName given in <a href="#">bs4TabItem</a> .
icon	Item icon.
startExpanded	Whether to expand the <a href="#">bs4SidebarMenuItem</a> at start.

## Author(s)

David Granjon, <dgranjon@gmail.com>

---

bs4SidebarMenuSubItem *Create a Bootstrap 4 dashboard main sidebar menu sub-item*

---

### Description

Build an adminLTE3 dashboard main sidebar menu sub-item

### Usage

```
bs4SidebarMenuSubItem(text, tabName = NULL, icon = NULL)
```

```
menuSubItem(text, tabName = NULL, icon = NULL)
```

### Arguments

text	Item name.
tabName	Should correspond exactly to the tabName given in <a href="#">bs4TabItem</a> .
icon	Item icon.

### Author(s)

David Granjon, <dgranjon@gmail.com>

### Examples

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(
        bs4SidebarMenu(
          bs4SidebarHeader("List of items 1"),
          bs4SidebarMenuItem(
            text = "Item List",
            icon = "bars",
            startExpanded = TRUE,
            bs4SidebarMenuSubItem(
              text = "Item 1",
              tabName = "item1",
              icon = "circle-thin"
            ),
            bs4SidebarMenuSubItem(
              text = "Item 2",
              tabName = "item2",
              icon = "circle-thin"
            )
          )
        )
      )
    )
  )
}
```



```
    ),
    bs4SidebarHeader("Classic Items"),
    bs4SidebarMenuItem(
        text = "Item 3",
        tabName = "item3"
    ),
    bs4SidebarHeader("List of items 2"),
    bs4SidebarMenuItem(
        text = "Item List 2",
        icon = "bars",
        startExpanded = FALSE,
        #active = FALSE,
        bs4SidebarMenuSubItem(
            text = "Item 4",
            tabName = "item4",
            icon = "circle-thin"
        ),
        bs4SidebarMenuSubItem(
            text = "Item 5",
            tabName = "item5",
            icon = "circle-thin"
        )
    )
)
),
controlbar = bs4DashControlbar(),
footer = bs4DashFooter(),
title = "test",
body = bs4DashBody(
    bs4TabItems(
        bs4TabItem(
            tabName = "item1",
            bs4Card(
                title = "Card 1",
                closable = TRUE,
                width = 6,
                solidHeader = TRUE,
                status = "primary",
                collapsible = TRUE,
                p("Box Content")
            )
        ),
        bs4TabItem(
            tabName = "item2",
            bs4Card(
                title = "Card 2",
                closable = TRUE,
                width = 6,
                solidHeader = TRUE,
                status = "warning",
                collapsible = TRUE,
                p("Box Content")
            )
        )
    )
)
```

```
    ),
    bs4TabItem(
      tabName = "item3",
      bs4Card(
        title = "Card 3",
        closable = TRUE,
        width = 6,
        solidHeader = TRUE,
        status = "danger",
        collapsible = TRUE,
        p("Box Content")
      )
    ),
    bs4TabItem(
      tabName = "item4",
      bs4Card(
        title = "Card 4",
        closable = TRUE,
        width = 6,
        solidHeader = TRUE,
        status = "info",
        collapsible = TRUE,
        p("Box Content")
      )
    ),
    bs4TabItem(
      tabName = "item5",
      bs4Card(
        title = "Card 5",
        closable = TRUE,
        width = 6,
        solidHeader = TRUE,
        status = "success",
        collapsible = TRUE,
        p("Box Content")
      )
    )
  )
),
server = function(input, output) {}
}
```

---

bs4SidebarUserPanel *Create a Bootstrap 4 dashboard main sidebar user panel*

---

### **Description**

Build an adminLTE3 dashboard main sidebar user panel

**Usage**

```
bs4SidebarUserPanel(img = NULL, text = NULL)
```

```
sidebarUserPanel(img = NULL, text = NULL)
```

**Arguments**

img	User panel image path or url.
text	User panel text.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

bs4SocialCard	<i>AdminLTE3 social card</i>
---------------	------------------------------

---

**Description**

Create social card

**Usage**

```
bs4SocialCard(
  ...,
  src = NULL,
  title = NULL,
  subtitle = NULL,
  width = 6,
  height = NULL,
  collapsible = TRUE,
  closable = TRUE,
  comments = NULL,
  footer = NULL
)
```

**Arguments**

...	Body content. May include <a href="#">attachmentBlock</a> for instance.
src	Header image, if any.
title	Card title.
subtitle	card subtitle.
width	Card width (between 1 and 12).
height	Card height.

collapsible	If TRUE, display a button in the upper right that allows the user to collapse the card.
closable	If TRUE, display a button in the upper right that allows the user to close the card.
comments	Slot for <a href="#">cardComment</a> .
footer	Card footer, if any.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

**Examples**

```

if (interactive()) {
  library(shiny)
  library(bs4Dash)
  shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(
        bs4SidebarMenu(
          bs4SidebarHeader("Main content"),
          bs4SidebarMenuItem(
            "Social Card",
            tabName = "social_card",
            icon = "desktop"
          )
        )
      ),
      footer = bs4DashFooter(),
      body = bs4DashBody(
        bs4TabItems(
          bs4TabItem(
            tabName = "profile_card",
            bs4SocialCard(
              title = "Social Card",
              subtitle = "example-01.05.2018",
              src = "https://adminlte.io/themes/AdminLTE/dist/img/user4-128x128.jpg",
              "Some text here!",
              comments = tagList(
                lapply(X = 1:10, FUN = function(i) {
                  cardComment(
                    src = "https://adminlte.io/themes/AdminLTE/dist/img/user3-128x128.jpg",
                    title = paste("Comment", i),
                    date = "01.05.2018",
                    paste0("The ", i, "-th comment")
                  )
                })
              )
            ),
            footer = "The footer here!"
          )
        )
      )
    )
  }

```

```
    )
  )
),
title = "socialCard"
),
server = function(input, output) { }
)
}
```

---

bs4Sortable

*BS4 sortable section*

---

## Description

Create a sortable UI section

## Usage

```
bs4Sortable(..., width = NULL)
```

## Arguments

...	Slot for UI elements such as <a href="#">bs4Card</a> .
width	Section width: between 1 and 12.

## Author(s)

David Granjon, <dgranjon@gmail.com>

## Examples

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)
  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        bs4TabItems(
          bs4TabItem(
            tabName = "sorted",
            fluidRow(
              lapply(1:3, FUN = function(i) {
                bs4Sortable(
                  width = 4,
```

```

    p(class = "text-center", paste("Column", i)),
    lapply(1:2, FUN = function(j) {
      bs4Card(
        title = paste0("I am the ", j, "-th card of the ", i, "-th column"),
        width = 12,
        "Click on my header"
      )
    })
  )
)
),
server = function(input, output) {}
)
}

```

---

 bs4Stars

*AdminLTE3 stars*


---

## Description

Create a block of stars (ideal for rating)

## Usage

```
bs4Stars(maxstar = 5, grade, status = "warning")
```

## Arguments

maxstar	Maximum number of stars by block.
grade	Current score. Should be positive and lower or equal to maxstar.
status	Star color: see <code>validColors()</code> in the documentation.

## Author(s)

David Granjon, <dgranjon@ymail.com>

## Examples

```

if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(

```

```

    navbar = bs4DashNavbar(),
    sidebar = bs4DashSidebar(),
    controlbar = bs4DashControlbar(),
    footer = bs4DashFooter(),
    title = "test",
    body = bs4DashBody(
      bs4Card(
        title = "Stars",
        bs4Stars(grade = 5),
        bs4Stars(grade = 5, status = "success"),
        bs4Stars(grade = 1, status = "danger"),
        bs4Stars(grade = 3, status = "info")
      )
    ),
    server = function(input, output) {}
  )
}

```

---

bs4TabCard

*Create a Bootstrap 4 tabCard*


---

## Description

Build an adminLTE3 card with tabs

## Usage

```

bs4TabCard(
  ...,
  id,
  title = NULL,
  status = NULL,
  elevation = NULL,
  solidHeader = FALSE,
  headerBorder = TRUE,
  gradientColor = NULL,
  tabStatus = NULL,
  width = 6,
  height = NULL,
  collapsible = TRUE,
  collapsed = FALSE,
  closable = TRUE,
  maximizable = FALSE,
  overflow = FALSE,
  side = c("left", "right")
)

```

```

tabBox(
  ...,
  id,
  title = NULL,
  status = NULL,
  elevation = NULL,
  solidHeader = FALSE,
  headerBorder = TRUE,
  gradientColor = NULL,
  tabStatus = NULL,
  width = 6,
  height = NULL,
  collapsible = TRUE,
  collapsed = FALSE,
  closable = TRUE,
  maximizable = FALSE,
  overflow = FALSE,
  side = c("left", "right")
)

```

### Arguments

...	Contents of the box: should be <a href="#">bs4TabPanel</a> .
id	Unique <a href="#">bs4TabSetPanel</a> id.
title	TabCard title.
status	The status of the card header. "primary", "secondary", "success", "warning", "danger", "white", "light", "dark", "transparent". NULL by default.
elevation	tabCard elevation.
solidHeader	Should the header be shown with a solid color background?
headerBorder	Whether to display a border between the header and body. TRUE by default
gradientColor	If NULL (the default), the background of the box will be white. Otherwise, a color string. "primary", "success", "warning" or "danger".
tabStatus	The status of the tabs buttons over header. "primary", "secondary", "success", "warning", "danger", "white", "light", "dark", "transparent". NULL by default, "light" if status is set. A vector is possible with a colour for each tab button
width	The width of the box, using the Bootstrap grid system. This is used for row-based layouts. The overall width of a region is 12, so the default width of 4 occupies 1/3 of that width. For column-based layouts, use NULL for the width; the width is set by the column that contains the box.
height	The height of a box, in pixels or other CSS unit. By default the height scales automatically with the content.
collapsible	If TRUE, display a button in the upper right that allows the user to collapse the box.
collapsed	If TRUE, start collapsed. This must be used with collapsible=TRUE.



<code>closable</code>	If TRUE, display a button in the upper right that allows the user to close the box.
<code>maximizable</code>	If TRUE, the card can be displayed in full screen mode.
<code>overflow</code>	Whether to enable overflow in the card body and footer. FALSE by default.
<code>side</code>	Side of the box the tabs should be on ("left" or "right").

**Author(s)**

David Granjon, <dgranjon@gmail.com>

**See Also**

Other cards: [bs4Card\(\)](#), [bs4InfoBox\(\)](#), [bs4ValueBox\(\)](#)

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        bs4TabCard(
          id = "tabcard",
          title = "A card with tabs",
          bs4TabPanel(
            tabName = "Tab 1",
            active = FALSE,
            "Content 1"
          ),
          bs4TabPanel(
            tabName = "Tab 2",
            active = TRUE,
            "Content 2"
          ),
          bs4TabPanel(
            tabName = "Tab 3",
            active = FALSE,
            "Content 3"
          )
        )
      )
    ),
    server = function(input, output) {}
  )
}
```

---

bs4TabItem	<i>One tab to put inside a tab items container</i>
------------	--

---

**Description**

One tab to put inside a tab items container

Alias to [bs4TabItem](#) See [bs4TabItem](#) for full details

**Usage**

```
bs4TabItem(tabName = NULL, ...)
```

```
tabItem(tabName = NULL, ...)
```

**Arguments**

tabName	The name of a tab. This must correspond to the tabName of a <a href="#">bs4SidebarMenuItem</a> .
...	Contents of the tab.

---

bs4TabItems	<i>A container for tab items</i>
-------------	----------------------------------

---

**Description**

A container for tab items

Alias to [bs4TabItems](#) See [bs4TabItems](#) for full details

**Usage**

```
bs4TabItems(...)
```

```
tabItems(...)
```

**Arguments**

... Items to put in the container. Each item should be a [bs4TabItem](#).

---

bs4Table	<i>Create a Bootstrap 4 table container</i>
----------	---

---

**Description**

Build an argon table container

**Usage**

```
bs4Table(
  ...,
  cardWrap = FALSE,
  headTitles,
  bordered = FALSE,
  striped = FALSE,
  width = 12
)
```

**Arguments**

...	<a href="#">bs4TableItems</a> .
cardWrap	Whether to wrap the table in a card. FALSE by default.
headTitles	Table header names. Must have the same length as the number of <a href="#">bs4TableItem</a> in <a href="#">bs4TableItems</a> . Set "" to have an empty title field.
bordered	Whether to display border between elements. FALSE by default.
striped	Whether to displayed striped in elements. FALSE by default.
width	Table width. 12 by default.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

**Examples**

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)
  shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      body = bs4DashBody(
        bs4Table(
          cardWrap = TRUE,
          bordered = TRUE,
          striped = TRUE,
          headTitles = c(
```

```

        "PROJECT",
        "BUDGET",
        "STATUS",
        "USERS",
        "COMPLETION",
        ""
    ),
    bs4TableItems(
        bs4TableWidgetItem("bs4 Design System"),
        bs4TableWidgetItem(dataCell = TRUE, "$2,500 USD"),
        bs4TableWidgetItem(
            dataCell = TRUE,
            bs4Badge(
                "Pending",
                position = "right",
                status = "danger",
                rounded = TRUE
            )
        ),
        bs4TableWidgetItem(
            progressBar(id = "pb1", value = 50, size = "xxs")
        ),
        bs4TableWidgetItem(
            dataCell = TRUE,
            "test"
        ),
        bs4TableWidgetItem(
            actionButton(
                "go",
                "Go"
            )
        )
    ),
    footer = bs4DashFooter()
),
server = function(input, output) { }
}

```

---

bs4TableWidgetItem

*Create a Bootstrap 4 table item*


---

### Description

Build an bs4 table item

**Usage**

```
bs4TableWidgetItem(..., dataCell = FALSE)
```

**Arguments**

... Any HTML element.

dataCell Whether the cell should be contain data or text. <td> by default.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

bs4TableItems      *Create a Bootstrap 4 table item row*

---

**Description**

Build an bs4 table item row

**Usage**

```
bs4TableItems(...)
```

**Arguments**

... Slot for [bs4TableWidgetItem](#).

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

bs4TabPanel      *Create a tabPanel*

---

**Description**

To be included in a bs4TabCard

**Usage**

```
bs4TabPanel(..., tabName, active = FALSE)
```

```
bs4DashControlbarItem(..., tabName, active = FALSE)
```

```
tabPanel(..., tabName, active = FALSE)
```

**Arguments**

...	Tab content
tabName	Tab name: it will be also passed as the id argument. Should be unique.
active	Whether the tab is active or not. FALSE bu default.

**Author(s)**

David Granjon, <dgranjon@gmail.com>

---

bs4TabSetPanel      *Create a tabSetPanel*

---

**Description**

Imported by [bs4TabCard](#) but can be used alone.

**Usage**

```
bs4TabSetPanel(  
  ...,  
  id,  
  side,  
  status = NULL,  
  tabStatus = NULL,  
  .list = NULL,  
  vertical = FALSE  
)  
  
bs4DashControlbarMenu(  
  ...,  
  id,  
  side,  
  status = NULL,  
  tabStatus = NULL,  
  .list = NULL,  
  vertical = FALSE  
)  
  
tabsetPanel(  
  ...,  
  id,  
  side,  
  status = NULL,  
  tabStatus = NULL,  
  .list = NULL,  
  vertical = FALSE  
)
```

**Arguments**

...	Slot for <code>bs4TabPanel</code> .
<code>id</code>	Unique <code>bs4TabSetPanel</code> id.
<code>side</code>	Side of the box the tabs should be on ("left" or "right").
<code>status</code>	The status of the card header. "primary", "secondary", "success", "warning", "danger", "white", "light", "dark", "transparent". NULL by default.
<code>tabStatus</code>	The status of the tabs buttons over header. "primary", "secondary", "success", "warning", "danger", "white", "light", "dark", "transparent". NULL by default, "light" if status is set. A vector is possible with a colour for each tab button
<code>.list</code>	When elements are programmatically added, pass them here instead of in ...
<code>vertical</code>	Whether to display tabs in a vertical mode. FALSE by default.

**Author(s)**

David Granjon, <dgranjon@gmail.com>

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(

        # manually inserted panels
        bs4TabSetPanel(
          id = "tabcard",
          side = "left",
          bs4TabPanel(
            tabName = "Tab 1",
            active = FALSE,
            "Content 1"
          ),
          bs4TabPanel(
            tabName = "Tab 2",
            active = TRUE,
            "Content 2"
          ),
          bs4TabPanel(
            tabName = "Tab 3",
            active = FALSE,
            "Content 3"
          )
        )
      )
    )
}
```

```

    )
  ),

  br(), br(),
  # programmatically inserted panels
  bs4TabSetPanel(
    id = "tabset",
    side = "left",
    .list = lapply(1:3, function(i) {
      bs4TabPanel(
        tabName = paste0("Tab", i),
        active = FALSE,
        paste("Content", i)
      )
    })
  ),

  br(), br(),
  # vertical tabset
  bs4TabSetPanel(
    id = "verttabset",
    side = "left",
    vertical = TRUE,
    .list = lapply(1:3, function(i) {
      bs4TabPanel(
        tabName = paste0("Tab", i),
        active = FALSE,
        paste("Content", i)
      )
    })
  )
),
server = function(input, output) {}
)
}

```

---

 bs4Timeline

*AdminLTE3 timeline block*


---

### Description

Create a timeline block

### Usage

```
bs4Timeline(..., reversed = TRUE, width = 6)
```



### Arguments

...	slot for <code>bs4TimelineLabel</code> or <code>bs4TimelineItem</code> .
<code>reversed</code>	Whether the timeline is reversed or not.
<code>width</code>	Timeline width. Between 1 and 12.

### Note

`reversed` is useful when the user wants to use the timeline inside a box.

### Author(s)

David Granjon, <dgranjon@gmail.com>

### Examples

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        bs4Card(
          title = "Timeline",
          bs4Timeline(
            width = 12,
            reversed = TRUE,
            bs4TimelineEnd(status = "danger"),
            bs4TimelineLabel("10 Feb. 2014", status = "info"),
            bs4TimelineItem(
              elevation = 4,
              title = "Item 1",
              icon = "gears",
              status = "success",
              time = "now",
              footer = "Here is the footer",
              "This is the body"
            ),
            bs4TimelineItem(
              title = "Item 2",
              border = FALSE
            ),
            bs4TimelineLabel("3 Jan. 2014", status = "primary"),
            bs4TimelineItem(
              elevation = 2,
              title = "Item 3",
```

```

        icon = "paint-brush",
        status = "warning",
        bs4TimelineItemMedia(src = "http://placeholder.it/150x100"),
        bs4TimelineItemMedia(src = "http://placeholder.it/150x100")
    ),
    bs4TimelineStart(status = "danger")
)
)
)
),
server = function(input, output) {}
)
}

```

---

bs4TimelineEnd      *AdminLTE3 timeline ending point*

---

### Description

Create a timeline ending point

### Usage

```
bs4TimelineEnd(icon = "hourglass-end", status = NULL)
```

### Arguments

icon	Item icon such as "clock-o", "times", ...
status	Item status: see here for a list of valid colors <a href="https://adminlte.io/themes/AdminLTE/pages/UI/general.html">https://adminlte.io/themes/AdminLTE/pages/UI/general.html</a> .

### Author(s)

David Granjon, <dgranjon@gmail.com>

---

bs4TimelineItem      *AdminLTE3 timeline item*

---

### Description

Create a timeline item

**Usage**

```
bs4TimelineItem(
    ...,
    elevation = NULL,
    icon = NULL,
    status = NULL,
    time = NULL,
    title = NULL,
    border = TRUE,
    footer = NULL
)
```

**Arguments**

...	Any element such as <a href="#">bs4TimelineItemMedia</a> ...
elevation	Timeline elevation (numeric). NULL by default.
icon	Item icon such as "clock-o", "times", ...
status	Item status: see here for a list of valid colors <a href="https://adminlte.io/themes/AdminLTE/pages/UI/general.html">https://adminlte.io/themes/AdminLTE/pages/UI/general.html</a> .
time	Item date or time.
title	Item title.
border	Whether to display a border between the header and the body. TRUE by default.
footer	Item footer if any.

**Author(s)**

David Granjon, <dgranjon@gmail.com>

---

bs4TimelineItemMedia *AdminLTE2 timeline media item*

---

**Description**

Create a timeline media item

**Usage**

```
bs4TimelineItemMedia(src = NULL, height = NULL, width = NULL)
```

**Arguments**

src	Media url or path.
height	Media height in pixels.
width	Media width in pixels.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

bs4TimelineLabel      *AdminLTE3 timeline label*

---

**Description**

Create a timeline label

**Usage**

```
bs4TimelineLabel(..., status = NULL)
```

**Arguments**

...	Any element.
status	Label status: see here for a list of valid colors <a href="https://adminlte.io/themes/AdminLTE/pages/UI/general.html">https://adminlte.io/themes/AdminLTE/pages/UI/general.html</a> .

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

bs4TimelineStart      *AdminLTE3 timeline starting point*

---

**Description**

Create a timeline starting point

**Usage**

```
bs4TimelineStart(icon = "clock-o", status = NULL)
```

**Arguments**

icon	Item icon such as "clock-o", "times", ...
status	Item status: see here for a list of valid colors <a href="https://adminlte.io/themes/AdminLTE/pages/UI/general.html">https://adminlte.io/themes/AdminLTE/pages/UI/general.html</a> .

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

`bs4UserCard`*AdminLTE3 widget user card*

---

## Description

Create widget user card

## Usage

```
bs4UserCard(  
  ...,  
  type = NULL,  
  src = NULL,  
  elevation = NULL,  
  imageElevation = NULL,  
  status = c("primary", "warning", "danger", "info", "success"),  
  title = NULL,  
  subtitle = NULL,  
  width = 6  
)
```

## Arguments

<code>...</code>	Footer content.
<code>type</code>	User card type. Either NULL or 2.
<code>src</code>	User image url or path.
<code>elevation</code>	User card elevation (numeric). NULL by default.
<code>imageElevation</code>	User card image elevation (numeric). NULL by default.
<code>status</code>	User card color. "primary", "warning", "danger", "info" or "success".
<code>title</code>	User card title.
<code>subtitle</code>	User card subtitle.
<code>width</code>	The width of the card, using the Bootstrap grid system.

## Author(s)

David Granjon, <dgranjon@gmail.com>

## Examples

```
if(interactive()){  
  library(shiny)  
  library(bs4Dash)  
  
  shiny::shinyApp(  
    ui = bs4DashPage(  
      ...  
    )  
  )  
}
```

```
navbar = bs4DashNavbar(),
sidebar = bs4DashSidebar(),
controlbar = bs4DashControlbar(),
footer = bs4DashFooter(),
title = "test",
body = bs4DashBody(
  fluidRow(
    bs4UserCard(
      src = "https://adminlte.io/themes/AdminLTE/dist/img/user1-128x128.jpg",
      status = "info",
      title = "User card type 1",
      subtitle = "a subtitle here",
      elevation = 4,
      "Any content here"
    ),
    bs4UserCard(
      type = 2,
      src = "https://adminlte.io/themes/AdminLTE/dist/img/user7-128x128.jpg",
      status = "success",
      imageElevation = 4,
      title = "User card type 2",
      subtitle = "a subtitle here",
      bs4ProgressBar(
        value = 5,
        striped = FALSE,
        status = "info"
      ),
      bs4ProgressBar(
        value = 5,
        striped = TRUE,
        status = "warning",
        width = "20%"
      )
    )
  )
),
server = function(input, output) {}
}
```

---

bs4UserMenu

*Create a Bootstrap 4 user profile.*

---

### **Description**

Create a Bootstrap 4 user profile.

**Usage**

```
bs4UserMenu(
  ...,
  name = NULL,
  src = NULL,
  title = NULL,
  subtitle = NULL,
  footer = NULL,
  status = c("primary", "danger", "success", "warning", "info", "secondary")
)
```

**Arguments**

...	Body content.
name	User name.
src	User profile picture.
title	A title.
subtitle	A subtitle.
footer	Footer is any.
status	Ribbon status: "primary", "danger", "success", "warning", "info" and "secondary".

**Examples**

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)

  shinyApp(
    ui = dashboardPage(
      navbar = dashboardHeader(
        rightUi = bs4UserMenu(
          name = "Divad Nojnarg",
          status = "primary",
          src = "https://adminlte.io/themes/AdminLTE/dist/img/user2-160x160.jpg",
          title = "bs4Dash",
          subtitle = "Author",
          footer = p("The footer", class = "text-center"),
          "This is the menu content."
        )
      ),
      sidebar = dashboardSidebar(),
      body = dashboardBody(),
      title = "bs4UserMenu"
    ),
    server = function(input, output) {}
  )
}
```

---

`bs4ValueBox`*Bootstrap 4 value box*

---

### Description

A beautiful AdminLTE3 value box.

### Usage

```
bs4ValueBox(  
    value,  
    subtitle,  
    icon = NULL,  
    elevation = NULL,  
    status = NULL,  
    width = 3,  
    footer = NULL,  
    href = NULL  
)
```

```
valueBox(  
    value,  
    subtitle,  
    icon = NULL,  
    elevation = NULL,  
    status = NULL,  
    width = 3,  
    footer = NULL,  
    href = NULL  
)
```

### Arguments

<code>value</code>	The value to display in the box. Usually a number or short text.
<code>subtitle</code>	Subtitle text.
<code>icon</code>	An icon tag, created by <a href="#">icon</a> .
<code>elevation</code>	Value box elevation.
<code>status</code>	A color for the box. "primary", "info", "success", "warning", "danger" or NULL.
<code>width</code>	The width of the box, using the Bootstrap grid system. This is used for row-based layouts. The overall width of a region is 12, so the default width of 4 occupies 1/3 of that width. For column-based layouts, use NULL for the width; the width is set by the column that contains the box.
<code>footer</code>	Optional html content for the footer of the box.
<code>href</code>	An optional URL to link to in the footer. Should both 'footer' and this parameter be set, 'footer' will take precedence.



**Author(s)**

David Granjon, <dgranjon@gmail.com>

**See Also**

Other cards: [bs4Card\(\)](#), [bs4InfoBox\(\)](#), [bs4TabCard\(\)](#)

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        fluidRow(
          bs4ValueBox(
            value = 150,
            subtitle = "New orders",
            status = "primary",
            icon = "shopping-cart",
            href = "#"
          ),
          bs4ValueBox(
            value = "53%",
            subtitle = "New orders",
            status = "danger",
            icon = "cogs",
            footer = shiny::div("Hello World")
          ),
          bs4ValueBox(
            value = "44",
            subtitle = "User Registrations",
            status = "warning",
            icon = "sliders"
          )
        )
      )
    ),
    server = function(input, output) {}
  )
}
```

---

bs4ValueBoxOutput	<i>Create a value box output (client side)</i>
-------------------	--

---

**Description**

This is the UI-side function for creating a dynamic [bs4ValueBox](#).

**Usage**

```
bs4ValueBoxOutput(outputId, width = 4)
```

```
bs4InfoBoxOutput(outputId, width = 4)
```

```
valueBoxOutput(outputId, width = 4)
```

**Arguments**

outputId	Output variable name.
width	The width of the box, using the Bootstrap grid system. This is used for row-based layouts. The overall width of a region is 12, so the default width of 4 occupies 1/3 of that width. For column-based layouts, use NULL for the width; the width is set by the column that contains the box.

**See Also**

[renderbs4ValueBox](#) for the corresponding server-side function and examples.

---

cardComment	<i>BS4 card comment container</i>
-------------	-----------------------------------

---

**Description**

Create a card comment to insert in [bs4SocialCard](#)

**Usage**

```
cardComment(..., src = NULL, title = NULL, date = NULL)
```

**Arguments**

...	Comment content.
src	Author image, if any.
title	Comment title.
date	Date of publication.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

cardPad	<i>AdminLTE3 vertical block container</i>
---------	---

---

**Description**

Create a vertical container for [descriptionBlock](#) or other elements

**Usage**

```
cardPad(..., color = NULL, style = NULL)
```

**Arguments**

...	Any element such as <a href="#">descriptionBlock</a> .
color	Background color: see here for a list of valid colors <a href="https://adminlte.io/themes/AdminLTE/pages/UI/general.html">https://adminlte.io/themes/AdminLTE/pages/UI/general.html</a> .
style	Custom CSS, if any.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

**Examples**

```
if (interactive()) {  
  library(shiny)  
  library(bs4Dash)  
  shinyApp(  
    ui = bs4DashPage(  
      bs4DashNavbar(),  
      bs4DashSidebar(),  
      bs4DashControlbar(),  
      bs4DashBody(  
        bs4Card(  
          title = "Box with right pad",  
          status = "warning",  
          fluidRow(  
            column(width = 6),  
            column(  
              width = 6,  
              cardPad(  
                color = "info",  
                descriptionBlock(  
                  header = "8390",  
                  text = "VISITS",
```

```

        right_border = FALSE,
        margin_bottom = TRUE
    ),
    descriptionBlock(
        header = "30%",
        text = "REFERRALS",
        right_border = FALSE,
        margin_bottom = TRUE
    ),
    descriptionBlock(
        header = "70%",
        text = "ORGANIC",
        right_border = FALSE,
        margin_bottom = FALSE
    )
)
)
)
)
),
title = "cardPad"
),
server = function(input, output) { }
)
}

```

---

cardProfile

*AdminLTE3 card profile*


---

### Description

Create card profile

### Usage

```
cardProfile(..., src = NULL, title = NULL, subtitle = NULL)
```

### Arguments

...	Any element such as <a href="#">cardProfileItemList</a> .
src	Profile image, if any.
title	Title.
subtitle	Subtitle.

### Author(s)

David Granjon, <dgranjon@ymail.com>

**Examples**

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(
        bs4SidebarMenu(
          bs4SidebarHeader("Main content"),
          bs4SidebarMenuItem(
            "Profile Card",
            tabName = "profile_card",
            icon = "desktop"
          )
        )
      ),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        bs4TabItems(
          bs4TabItem(
            tabName = "profile_card",
            bs4Card(
              status = "primary",
              solidHeader = TRUE,
              cardProfile(
                src = "https://adminlte.io/themes/AdminLTE/dist/img/user4-128x128.jpg",
                title = "Nina Mcintire",
                subtitle = "Software Engineer",
                cardProfileItemList(
                  bordered = TRUE,
                  cardProfileItem(
                    title = "Followers",
                    description = 1322
                  ),
                  cardProfileItem(
                    title = "Following",
                    description = 543
                  ),
                  cardProfileItem(
                    title = "Friends",
                    description = 13287
                  )
                )
              )
            )
          )
        )
      )
    )
  )
}
```

```
    ),  
    server = function(input, output) {}  
  )  
}
```

---

cardProfileItem      *AdminLTE3 card profile item*

---

### Description

Create card profile item

### Usage

```
cardProfileItem(title = NULL, description = NULL)
```

### Arguments

title	Item title.
description	Item info.

### Author(s)

David Granjon, <dgranjon@ymail.com>

---

cardProfileItemList      *AdminLTE3 card profile item container*

---

### Description

Create card profile item list

### Usage

```
cardProfileItemList(..., bordered = FALSE)
```

### Arguments

...	Slot for <a href="#">cardProfileItem</a> .
bordered	Whether the container should have a border or not. FALSE by default.

### Author(s)

David Granjon, <dgranjon@ymail.com>

---

column	<i>Bootstrap 4 column system</i>
--------	----------------------------------

---

### Description

This function overwrites that of Shiny since there are differences between the Bootstrap 3 and Bootstrap 4 grid systems

### Usage

```
column(width, ..., offset = 0)
```

### Arguments

width	The grid width of the column (must be between 1 and 12).
...	Elements to include within the column.
offset	The number of columns to offset this column from the end of the previous column.

### Examples

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)

  ui <- bs4DashPage(
    navbar = bs4DashNavbar(),
    sidebar = dashboardSidebar(
      bs4SidebarMenu(
        bs4SidebarMenuItem(
          "Welcome!",
          tabName = "tab_welcome",
          icon = "home"
        )
      )
    ),

    body = bs4DashBody(
      bs4TabItems(
        bs4TabItem(
          tabName = "tab_welcome",
          fluidRow(
            column(
              width = 1,
              offset = 11,
              actionButton(
                "mybutton", label = "", icon = icon("question-circle")
              )
            )
          )
        )
      )
    )
  )
}
```

```

    ),
    fluidRow(
      h2("Placeholder")
    )
  )
)
)
)
)
)

server <- function(input, output, session) {}
shinyApp(ui = ui, server = server)
}

```

---

controlbarItem	<i>Alias to <a href="#">bs4DashControlbarItem</a> See <a href="#">bs4DashControlbarItem</a> for full details</i>
----------------	--

---

### Description

Alias to [bs4DashControlbarItem](#) See [bs4DashControlbarItem](#) for full details

### Usage

```
controlbarItem(..., tabName, active = FALSE)
```

### Arguments

...	Tab content
tabName	Tab name: it will be also passed as the id argument. Should be unique.
active	Whether the tab is active or not. FALSE bu default.

---

controlbarMenu	<i>Alias to <a href="#">bs4DashControlbarMenu</a> See <a href="#">bs4DashControlbarMenu</a> for full details</i>
----------------	--

---

### Description

Alias to [bs4DashControlbarMenu](#) See [bs4DashControlbarMenu](#) for full details



**Usage**

```
controlbarMenu(
  ...,
  id,
  side,
  status = NULL,
  tabStatus = NULL,
  .list = NULL,
  vertical = FALSE
)
```

**Arguments**

...	Slot for <a href="#">bs4TabPanel</a> .
id	Unique <a href="#">bs4TabSetPanel</a> id.
side	Side of the box the tabs should be on ("left" or "right").
status	The status of the card header. "primary", "secondary", "success", "warning", "danger", "white", "light", "dark", "transparent". NULL by default.
tabStatus	The status of the tabs buttons over header. "primary", "secondary", "success", "warning", "danger", "white", "light", "dark", "transparent". NULL by default, "light" if status is set. A vector is possible with a colour for each tab button
.list	When elements are programmatically added, pass them here instead of in ...
vertical	Whether to display tabs in a vertical mode. FALSE by default.

---

descriptionBlock	<i>AdminLTE3 description block</i>
------------------	------------------------------------

---

**Description**

Create a description block, perfect for writing statistics

**Usage**

```
descriptionBlock(
  number = NULL,
  number_color = NULL,
  number_icon = NULL,
  header = NULL,
  text = NULL,
  right_border = TRUE,
  margin_bottom = FALSE
)
```

**Arguments**

number	Any number.
number_color	Number color: see here for a list of valid colors <a href="https://adminlte.io/themes/AdminLTE/pages/UI/general.html">https://adminlte.io/themes/AdminLTE/pages/UI/general.html</a> .
number_icon	Number icon, if any. Should be written like "fa fa-times".
header	Bold text.
text	Additional text.
right_border	TRUE by default. Whether to display a right border to separate two blocks. The last block on the right should not have a right border.
margin_bottom	FALSE by default. Set it to TRUE when the descriptionBlock is used in a <a href="#">card-Pad</a> context.

**Author(s)**

David Granjon, <dgranjon@gmail.com>

**Examples**

```

if (interactive()) {
  library(shiny)
  library(bs4Dash)
  shinyApp(
    ui = bs4DashPage(
      bs4DashNavbar(),
      bs4DashSidebar(),
      bs4DashControlbar(),
      bs4DashBody(
        bs4Card(
          solidHeader = FALSE,
          title = "Status summary",
          background = NULL,
          width = 4,
          status = "danger",
          footer = fluidRow(
            column(
              width = 6,
              descriptionBlock(
                number = "17%",
                number_color = "success",
                number_icon = "fa fa-caret-up",
                header = "$35,210.43",
                text = "TOTAL REVENUE",
                right_border = TRUE,
                margin_bottom = FALSE
              )
            ),
            column(
              width = 6,
              descriptionBlock(

```

```

        number = "18%",
        number_color = "danger",
        number_icon = "fa fa-caret-down",
        header = "1200",
        text = "GOAL COMPLETION",
        right_border = FALSE,
        margin_bottom = FALSE
    )
)
)
),
title = "Description Blocks"
),
server = function(input, output) { }
)
}

```

---

dropdownDivider      *Create a box dropdown divider*

---

**Description**

Create a box dropdown divider

**Usage**

dropdownDivider()

**Note**

Useful to separate 2 sections of dropdown items.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

dropdownItem      *Create a box dropdown item*

---

**Description**

Create a box dropdown item

**Usage**

dropdownItem(url = NULL, name = NULL)

**Arguments**

url	Target url or page.
name	Item name.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

dropdownItemList	<i>Create a box dropdown item list</i>
------------------	--

---

**Description**

Can be used to add dropdown items to a cardtool.

**Usage**

```
dropdownItemList(...)
```

**Arguments**

...	Slot for <a href="#">dropdownItem</a> .
-----	---

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

getAdminLTEColors	<i>Get all AdminLTE colors.</i>
-------------------	---------------------------------

---

**Description**

Get all AdminLTE colors.

**Usage**

```
getAdminLTEColors()
```

---

infoBoxOutput	<i>Alias to <a href="#">bs4InfoBoxOutput</a> See <a href="#">bs4InfoBoxOutput</a> for full details</i>
---------------	--

---

### Description

Alias to [bs4InfoBoxOutput](#) See [bs4InfoBoxOutput](#) for full details

### Usage

```
infoBoxOutput(outputId, width = 4)
```

### Arguments

outputId	Output variable name.
width	The width of the box, using the Bootstrap grid system. This is used for row-based layouts. The overall width of a region is 12, so the default width of 4 occupies 1/3 of that width. For column-based layouts, use NULL for the width; the width is set by the column that contains the box.

---

ionicon	<i>BS4 ionicons</i>
---------	---------------------

---

### Description

Create a ionicon.

### Usage

```
ionicon(name)
```

### Arguments

name	Name of icon. See <a href="https://ionicons.com">https://ionicons.com</a> .
------	---

### Note

Similar to the icon function from shiny.

### Author(s)

David Granjon, <[dgranjon@gmail.com](mailto:dgranjon@gmail.com)>

## Examples

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        ionicon(name = "heart"),
        ionicon(name = "beer")
      )
    ),
    server = function(input, output) {}
  )
}
```

---

renderbs4ValueBox      *Create a value box (server side)*

---

## Description

This is the server-side function for creating a dynamic [bs4ValueBox](#).

## Usage

```
renderbs4ValueBox(expr, env = parent.frame(), quoted = FALSE)
```

```
renderbs4InfoBox(expr, env = parent.frame(), quoted = FALSE)
```

```
renderValueBox(expr, env = parent.frame(), quoted = FALSE)
```

## Arguments

expr	An expression that returns a Shiny tag object, <a href="#">HTML()</a> , or a list of such objects.
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with <code>quote()</code> )? This is useful if you want to save an expression in a variable.

## Author(s)

David Granjon, <dgranjon@gmail.com>

**See Also**

[bs4ValueBoxOutput](#) for the corresponding UI-side function.

**Examples**

```
if(interactive()){
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = bs4DashPage(
      navbar = bs4DashNavbar(),
      sidebar = bs4DashSidebar(),
      controlbar = bs4DashControlbar(),
      footer = bs4DashFooter(),
      title = "test",
      body = bs4DashBody(
        fluidRow(
          bs4ValueBoxOutput("vbox"),
          bs4InfoBoxOutput("ibox")
        )
      )
    ),
    server = function(input, output) {

      output$vbox <- renderbs4ValueBox({
        bs4ValueBox(
          value = 150,
          subtitle = "New orders",
          status = "primary",
          icon = "shopping-cart",
          href = "#"
        )
      })

      output$ibox <- renderbs4InfoBox({
        bs4InfoBox(
          title = "Comments",
          gradientColor = "success",
          value = 41410,
          icon = "comments"
        )
      })
    }
  )
}
```

---

renderInfoBox	<i>Alias to <a href="#">renderbs4InfoBox</a> See <a href="#">renderbs4InfoBox</a> for full details</i>
---------------	--

---

### Description

Alias to [renderbs4InfoBox](#) See [renderbs4InfoBox](#) for full details

### Usage

```
renderInfoBox(expr, env = parent.frame(), quoted = FALSE)
```

### Arguments

expr	An expression that returns a Shiny tag object, <a href="#">HTML()</a> , or a list of such objects.
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with <code>quote()</code> )? This is useful if you want to save an expression in a variable.

---

updatebs4Card	<i>update an AdminLTE3 card from the server side</i>
---------------	--

---

### Description

update an AdminLTE3 card from the server side

### Usage

```
updatebs4Card(
  inputId,
  session,
  action = c("remove", "toggle", "toggleMaximize", "restore")
)
```

### Arguments

inputId	Card inputId
session	Shiny session
action	Action to trigger: <code>c("remove", "toggle", "toggleMaximize", "restore")</code> .



**Examples**

```

if (interactive()) {
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = dashboardPage(
      sidebar_collapsed = TRUE,
      navbar = dashboardHeader(),
      sidebar = dashboardSidebar(),
      body = dashboardBody(
        actionButton(inputId = "triggerCard", label = "Trigger Card Action"),
        selectInput(
          inputId = "cardAction",
          label = "Card action",
          choices = c(
            "remove",
            "toggle",
            "toggleMaximize",
            "restore"
          )
        ),
        bs4Card(
          inputId = "mycard",
          title = "The plot is visible when you maximize the card",
          closable = TRUE,
          maximizable = TRUE,
          width = 12,
          status = "warning",
          solidHeader = FALSE,
          collapsible = TRUE,
          sliderInput("obs", "Number of observations:",
            min = 0, max = 1000, value = 500
          ),
          plotOutput("distPlot")
        )
      ),
    server = function(input, output, session) {

      output$distPlot <- renderPlot({
        if (input$mycard$maximized) {
          hist(rnorm(input$obs))
        }
      })

      observeEvent(input$triggerCard, {
        updatebs4Card(inputId = "mycard", session = session, action = input$cardAction)
      })

      observe({

```

```

    print(
      list(
        collapsed = input$mycard$collapsed,
        maximized = input$mycard$maximized,
        visible = input$mycard$visible
      )
    )
  })
}
)
}

```

---

updatebs4Controlbar     *Function to programmatically toggle the state of the controlbar*

---

### Description

Function to programmatically toggle the state of the controlbar

### Usage

```
updatebs4Controlbar(inputId, session)
```

### Arguments

inputId	Controlbar id.
session	Shiny session object.

### Examples

```

if (interactive()) {
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = dashboardPage(
      controlbar_collapsed = FALSE,
      controlbar_overlay = TRUE,
      navbar = dashboardHeader(),
      sidebar = dashboardSidebar(),
      body = dashboardBody(
        actionButton(inputId = "controlbarToggle", label = "Toggle Controlbar")
      ),
      controlbar = dashboardControlbar(inputId = "controlbar")
    ),
    server = function(input, output, session) {

      observeEvent(input$controlbar, {
        if (input$controlbar) {
          showModal(modalDialog(

```

```

        title = "Alert",
        "The controlbar is opened.",
        easyClose = TRUE,
        footer = NULL
      ))
    }
  })

  observeEvent(input$controlbarToggle, {
    updatebs4Controlbar(inputId = "controlbar", session = session)
  })

  observe({
    print(input$controlbar)
  })
}
)
}

```

---

updatebs4Sidebar      *Function to programmatically toggle the state of the sidebar*

---

### Description

Function to programmatically toggle the state of the sidebar

### Usage

```
updatebs4Sidebar(inputId, session)
```

### Arguments

inputId	Sidebar id.
session	Shiny session object.

### Examples

```

if (interactive()) {
  library(shiny)
  library(bs4Dash)

  shiny::shinyApp(
    ui = dashboardPage(
      controlbar_collapsed = FALSE,
      controlbar_overlay = TRUE,
      navbar = dashboardHeader(),
      sidebar = dashboardSidebar(inputId = "sidebar"),
      body = dashboardBody(
        actionButton(inputId = "controlbarToggle", label = "Toggle Sidebar")
      )
    )
  )
}

```

```

),
server = function(input, output, session) {

  observeEvent(input$sidebar, {
    if (input$sidebar) {
      showModal(modalDialog(
        title = "Alert",
        "The sidebar is opened.",
        easyClose = TRUE,
        footer = NULL
      ))
    }
  })

  observeEvent(input$controlbarToggle, {
    updatebs4Sidebar(inputId = "sidebar", session = session)
  })

  observe({
    print(input$sidebar)
  })
}
)
}

```

---

updatebs4TabItems

*Change the selected sidebar tab on the client*


---

### Description

This function controls the active tab of [bs4TabItems](#) from the server. It behaves just like [updatebs4TabSetPanel](#).

### Usage

```
updatebs4TabItems(session, inputId, selected = NULL)
```

```
updateTabItems(session, inputId, selected = NULL)
```

### Arguments

session	shiny session.
inputId	<a href="#">bs4TabSetPanel</a> unique id.
selected	the tab to be selected.

**Examples**

```

## Only run this example in interactive R sessions
if (interactive()) {
  library(shiny)
  library(bs4Dash)

  shinyApp(
    ui = bs4DashPage(
      sidebar_collapsed = FALSE,
      controlbar_collapsed = TRUE,
      enable_preloader = FALSE,
      navbar = bs4DashNavbar(skin = "dark"),
      body = bs4DashBody(
        bs4TabItems(
          bs4TabItem(
            tabName = "tab1",
            sliderInput("obs", "Number of observations:",
              min = 0, max = 1000, value = 500
            ),
            plotOutput("distPlot")
          ),
          bs4TabItem(
            tabName = "tab2",
            checkboxGroupInput("variable", "Variables to show:",
              c("Cylinders" = "cyl",
                "Transmission" = "am",
                "Gears" = "gear")),
            tableOutput("data")
          ),
          bs4TabItem(
            tabName = "tab3",
            checkboxInput("val", "Some value", FALSE),
            textOutput("value")
          )
        )
      ),
    sidebar = bs4DashSidebar(
      skin = "light",
      bs4SidebarMenu(
        id = "sidebar",
        bs4SidebarMenuItem(
          text = "Tab 1",
          tabName = "tab1",
          icon = "card"
        ),
        bs4SidebarMenuItem(
          text = "Tab 2",
          tabName = "tab2",
          icon = "card"
        ),
        bs4SidebarMenuItem(
          text = "Tab 3",

```

```

        tabName = "tab3",
        icon = "card"
      )
    )
  ),
  controlbar = bs4DashControlbar(
    skin = "light",
    sliderInput(
      inputId = "controller",
      label = "Update the first tabset",
      min = 1,
      max = 3,
      value = 2
    )
  ),
  footer = bs4DashFooter()
),
server = function(input, output, session) {

  # update tabset1
  observeEvent(input$controller, {
    updatebs4TabSetPanel(
      session,
      inputId = "sidebar",
      selected = input$controller
    )
  })

  output$distPlot <- renderPlot({
    hist(rnorm(input$obs))
  })

  output$data <- renderTable({
    mtcars[, c("mpg", input$variable), drop = FALSE]
  }, rownames = TRUE)

  output$value <- renderText({ input$val })

}
)
}

```

---

updatebs4TabSetPanel *Update a [bs4TabSetPanel](#)*

---

### Description

Update a [bs4TabSetPanel](#)

Alias to [updatebs4TabSetPanel](#) See [updatebs4TabSetPanel](#) for full details

**Usage**

```
updatebs4TabSetPanel(session, inputId, selected = NULL)

updatebs4ControlbarMenu(session, inputId, selected = NULL)

updateTabsetPanel(session, inputId, selected = NULL)
```

**Arguments**

session	shiny session.
inputId	<a href="#">bs4TabSetPanel</a> unique id.
selected	the tab to be selected.

**Examples**

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)

  shinyApp(
    ui = bs4DashPage(
      sidebar_collapsed = FALSE,
      controlbar_collapsed = TRUE,
      enable_preloader = FALSE,
      loading_duration = 2,
      navbar = bs4DashNavbar(skin = "dark"),
      body = bs4DashBody(
        bs4TabSetPanel(
          id = "tabset1",
          side = "left",
          bs4TabPanel(
            tabName = "Tab 1",
            active = FALSE,
            numericInput("val", "Value:", 10, min = 1, max = 100),
            verbatimTextOutput("value")
          ),
          bs4TabPanel(
            tabName = "Tab 2",
            active = TRUE,
            "Content 2"
          ),
          bs4TabPanel(
            tabName = "Tab 3",
            active = FALSE,
            checkboxGroupInput(
              inline = TRUE,
              "variable", "Variables to show:",
              c("Cylinders" = "cyl",
                "Transmission" = "am",
                "Gears" = "gear")
            )
          )
        )
      )
    )
  )
}
```

```

    ),
    tableOutput("data")
  )
),
bs4TabSetPanel(
  id = "tabset2",
  side = "left",
  vertical = TRUE,
  bs4TabPanel(
    tabName = "Tab 1",
    active = FALSE,
    sliderInput(
      "obs",
      "Number of observations:",
      min = 0,
      max = 1000,
      value = 500
    ),
    plotOutput("distPlot")
  ),
  bs4TabPanel(
    tabName = "Tab 2",
    active = TRUE,
    radioButtons(
      inline = TRUE,
      "rb", "Choose one:",
      choiceNames = list(
        icon("calendar"),
        HTML("<p style='color:red;'>Red Text</p>"),
        "Normal text"
      ),
      choiceValues = list(
        "icon", "html", "text"
      )
    ),
    textOutput("txt")
  ),
  bs4TabPanel(
    tabName = "Tab 3",
    active = FALSE,
    "Content 3"
  )
),
),
sidebar = bs4DashSidebar(
  skin = "light",
  sliderInput(
    inputId = "controller",
    label = "Update the first tabset",
    min = 1,
    max = 3,
    value = 2
  ),
),

```



```
    br(),
    sliderInput(
      inputId = "controller2",
      label = "Update the second tabset",
      min = 1,
      max = 3,
      value = 3
    )
  ),
  controlbar = bs4DashControlbar(skin = "light"),
  footer = bs4DashFooter()
),
server = function(input, output, session) {

  # update tabset1
  observeEvent(input$controller, {
    updatebs4TabSetPanel(
      session,
      inputId = "tabset1",
      selected = input$controller
    )
  })

  # update tabset 2
  observeEvent(input$controller2, {
    updatebs4TabSetPanel(
      session,
      inputId = "tabset2",
      selected = input$controller2
    )
  })

  output$distPlot <- renderPlot({
    hist(rnorm(input$obs))
  })

  output$data <- renderTable({
    mtcars[, c("mpg", input$variable), drop = FALSE]
  }, rownames = TRUE)

  output$txt <- renderText({
    paste("You chose", input$rb)
  })

  output$value <- renderText({ input$val })

}
}
```

---

updateControlbarMenu *Alias to [updatebs4ControlbarMenu](#) See [updatebs4ControlbarMenu](#) for full details*

---

### Description

Alias to [updatebs4ControlbarMenu](#) See [updatebs4ControlbarMenu](#) for full details

### Usage

```
updateControlbarMenu(session, inputId, selected = NULL)
```

### Arguments

session	shiny session.
inputId	<a href="#">bs4TabSetPanel</a> unique id.
selected	the tab to be selected.

---

userMessage *AdminLTE3 user message*

---

### Description

Create a user message

### Usage

```
userMessage(..., author = NULL, date = NULL, src = NULL, side = NULL)
```

### Arguments

...	Message text.
author	Message author.
date	Message date.
src	Message author image path or url.
side	Side where author is displayed: NULL (left, by default) or "right".

### Author(s)

David Granjon, <dgranjon@gmail.com>

---

userMessages	<i>AdminLTE3 user message container</i>
--------------	---

---

### Description

Create a user message container

### Usage

```
userMessages(..., status, width = 4)
```

### Arguments

...	Slot for <code>userMessage</code> .
status	Messages status. See here for a list of valid colors <a href="https://adminlte.io/themes/AdminLTE/pages/UI/general.html">https://adminlte.io/themes/AdminLTE/pages/UI/general.html</a> .
width	Container width: between 1 and 12.

### Note

Better to include in a `bs4Card`.

### Author(s)

David Granjon, <dgranjon@gmail.com>

### Examples

```
if (interactive()) {
  library(shiny)
  library(bs4Dash)
  shinyApp(
    ui = bs4DashPage(
      bs4DashNavbar(),
      bs4DashSidebar(),
      bs4DashControlbar(),
      bs4DashBody(
        userMessages(
          status = "success",
          userMessage(
            author = "Alexander Pierce",
            date = "20 Jan 2:00 pm",
            src = "https://adminlte.io/themes/AdminLTE/dist/img/user1-128x128.jpg",
            side = NULL,
            "Is this template really for free? That's unbelievable!"
          )
        )
      )
    ),
    title = "user Message"
```

```

    ),
    server = function(input, output) { }
  )
}

```

---

 userPost

*AdminLTE3 user post*


---

### Description

Create a user post

### Usage

```

userPost(
  ...,
  id = NULL,
  src = NULL,
  author = NULL,
  description = NULL,
  collapsible = TRUE,
  collapsed = FALSE,
  collapse_status = NULL
)

```

### Arguments

...	Post content, slot for <a href="#">userPostTagItems</a> , <a href="#">userPostMedia</a> .
id	Unique id of the post.
src	Profile image, if any.
author	Post author.
description	Post description.
collapsible	If TRUE, display a button in the upper right that allows the user to collapse the comment.
collapsed	Whether the comment is collapsed when the application starts, FALSE by default.
collapse_status	Color of the collapse button. NULL by default but also "primary", "info", "danger", "warning", "success", "secondary" ...

### Author(s)

David Granjon, <dgranjon@ymail.com>

**Examples**

```

if (interactive()) {
  library(shiny)
  library(bs4Dash)
  shinyApp(
    ui = bs4DashPage(
      bs4DashNavbar(),
      bs4DashSidebar(),
      bs4DashControlbar(),
      bs4DashBody(
        bs4Card(
          title = "Box with user comment",
          status = "primary",
          userPost(
            id = 1,
            src = "https://adminlte.io/themes/AdminLTE/dist/img/user1-128x128.jpg",
            author = "Jonathan Burke Jr.",
            description = "Shared publicly - 7:30 PM today",
            "Lorem ipsum represents a long-held tradition for designers,
            typographers and the like. Some people hate it and argue for
            its demise, but others ignore the hate as they create awesome
            tools to help create filler text for everyone from bacon
            lovers to Charlie Sheen fans.",
            userPostTagItems(
              userPostTagItem(bs4Badge("item 1", status = "warning")),
              userPostTagItem(bs4Badge("item 2", status = "danger"))
            )
          ),
          userPost(
            id = 2,
            src = "https://adminlte.io/themes/AdminLTE/dist/img/user6-128x128.jpg",
            author = "Adam Jones",
            description = "Shared publicly - 5 days ago",
            userPostMedia(src = "https://adminlte.io/themes/AdminLTE/dist/img/photo2.png"),
            userPostTagItems(
              userPostTagItem(bs4Badge("item 1", status = "info")),
              userPostTagItem(bs4Badge("item 2", status = "danger"))
            )
          )
        )
      ),
      title = "userPost"
    ),
    server = function(input, output) { }
  )
}

```

**Description**

Create a user post media (image)

**Usage**

```
userPostMedia(src = NULL, height = NULL, width = NULL)
```

**Arguments**

src	Image path or url ...
height	Media height in pixels.
width	Media width in pixels.

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

userPostTagItem	<i>AdminLTE3 user post tool item</i>
-----------------	--------------------------------------

---

**Description**

Create a user post tool item

**Usage**

```
userPostTagItem(...)
```

**Arguments**

...	Tool content such as label, button, ...
-----	---

**Author(s)**

David Granjon, <dgranjon@ymail.com>

---

*userPostTagItems*      *AdminLTE3 user post tool item container*

---

**Description**

Create a user post tool item container

**Usage**

`userPostTagItems(...)`

**Arguments**

...      Slot for [userPostTagItem](#).

**Author(s)**

David Granjon, <[dgranjon@gmail.com](mailto:dgranjon@gmail.com)>

# Index

attachmentBlock, [4](#), [51](#)

box (bs4Card), [13](#)

bs4Accordion, [5](#)

bs4AccordionItem, [5](#), [6](#)

bs4Alert, [7](#)

bs4Badge, [8](#)

bs4Box, [9](#)

bs4Callout, [11](#)

bs4Card, [13](#), [31](#), [53](#), [57](#), [73](#), [99](#)

bs4Carousel, [17](#), [18](#)

bs4CarouselItem, [17](#), [18](#)

bs4DashBody, [19](#), [23](#)

bs4DashControlbar, [19](#), [23](#)

bs4DashControlbarItem, [80](#)

bs4DashControlbarItem (bs4TabPanel), [61](#)

bs4DashControlbarMenu, [80](#)

bs4DashControlbarMenu (bs4TabSetPanel), [62](#)

bs4DashFooter, [20](#), [23](#)

bs4DashGallery, [21](#)

bs4DashNavbar, [21](#), [23](#)

bs4DashPage, [22](#)

bs4DashSidebar, [23](#), [26](#)

bs4DropdownMenu, [27](#)

bs4DropdownMenuItem, [28](#), [29](#)

bs4InfoBox, [15](#), [30](#), [30](#), [57](#), [73](#)

bs4InfoBoxOutput, [85](#)

bs4InfoBoxOutput (bs4ValueBoxOutput), [74](#)

bs4InsertTab, [32](#)

bs4Jumbotron, [34](#)

bs4ListGroup, [35](#)

bs4ListGroupItem, [35](#), [37](#)

bs4Loading, [38](#)

bs4ProgressBar, [39](#)

bs4Quote, [40](#)

bs4RemoveTab, [41](#)

bs4Ribbon, [43](#)

bs4SidebarHeader, [44](#), [45](#)

bs4SidebarMenu, [26](#), [45](#), [45](#)

bs4SidebarMenuItem, [45](#), [47](#), [47](#), [58](#)

bs4SidebarMenuSubItem, [47](#), [48](#)

bs4SidebarUserPanel, [50](#)

bs4SocialCard, [51](#), [74](#)

bs4Sortable, [53](#)

bs4Stars, [54](#)

bs4TabCard, [15](#), [31](#), [55](#), [62](#), [73](#)

bs4TabItem, [47](#), [48](#), [58](#), [58](#)

bs4TabItems, [19](#), [58](#), [58](#), [92](#)

bs4Table, [59](#)

bs4TableItem, [59](#), [60](#), [61](#)

bs4TableItems, [59](#), [61](#)

bs4TabPanel, [32](#), [41](#), [42](#), [56](#), [61](#), [63](#), [81](#)

bs4TabSetPanel, [32](#), [41](#), [42](#), [56](#), [62](#), [63](#), [81](#), [92](#), [94](#), [95](#), [98](#)

bs4Timeline, [64](#)

bs4TimelineEnd, [66](#)

bs4TimelineItem, [65](#), [66](#)

bs4TimelineItemMedia, [67](#), [67](#)

bs4TimelineLabel, [65](#), [68](#)

bs4TimelineStart, [68](#)

bs4UserCard, [69](#)

bs4UserMenu, [70](#)

bs4ValueBox, [15](#), [31](#), [57](#), [72](#), [74](#), [86](#)

bs4ValueBoxOutput, [74](#), [87](#)

cardComment, [52](#), [74](#)

cardPad, [75](#), [82](#)

cardProfile, [76](#)

cardProfileItem, [78](#), [78](#)

cardProfileItemList, [76](#), [78](#)

column, [79](#)

controlbarItem, [80](#)

controlbarMenu, [80](#)

dashboardBody (bs4DashBody), [19](#)

dashboardControlbar (bs4DashControlbar), [19](#)

dashboardFooter (bs4DashFooter), [20](#)

dashboardHeader (bs4DashNavbar), [21](#)



- dashboardPage (bs4DashPage), 22
- dashboardSidebar (bs4DashSidebar), 26
- descriptionBlock, 75, 81
- dropdownDivider, 83
- dropdownItem, 83, 84
- dropdownItemList, 15, 84
- dropdownMenu (bs4DropdownMenu), 27
- dropdownMenuItem (bs4DropdownMenuItem), 29
  
- getAdminLTEColors, 22, 29, 84
  
- HTML(), 86, 88
  
- icon, 30, 72
- infoBox (bs4InfoBox), 30
- infoBoxOutput, 85
- ionicon, 85
  
- menuItem (bs4SidebarMenuItem), 47
- menuSubItem (bs4SidebarMenuSubItem), 48
  
- renderbs4InfoBox, 88
- renderbs4InfoBox (renderbs4ValueBox), 86
- renderbs4ValueBox, 74, 86
- renderInfoBox, 88
- renderValueBox (renderbs4ValueBox), 86
  
- sidebarHeader (bs4SidebarHeader), 44
- sidebarMenu (bs4SidebarMenu), 45
- sidebarUserPanel (bs4SidebarUserPanel), 50
  
- tabBox (bs4TabCard), 55
- tabItem (bs4TabItem), 58
- tabItems (bs4TabItems), 58
- tabPanel (bs4TabPanel), 61
- tabsetPanel (bs4TabSetPanel), 62
  
- updatebs4Card, 88
- updatebs4Controlbar, 90
- updatebs4ControlbarMenu, 98
- updatebs4ControlbarMenu (updatebs4TabSetPanel), 94
- updatebs4Sidebar, 91
- updatebs4TabItems, 92
- updatebs4TabSetPanel, 92, 94, 94
- updateControlbarMenu, 97
- updateTabItems (updatebs4TabItems), 92
  
- updateTabsetPanel (updatebs4TabSetPanel), 94
- userMessage, 98, 99
- userMessages, 99
- userPost, 100
- userPostMedia, 100, 101
- userPostTagItem, 102, 103
- userPostTagItems, 100, 103
  
- valueBox (bs4ValueBox), 72
- valueBoxOutput (bs4ValueBoxOutput), 74