

Package ‘bs4Dashkit’

July 13, 2026

Type Package

Title Branding, Theme Application and Navigation Utilities for
'bs4Dash' Dashboards

Version 0.3.0

Description Provides branding, theme application, and navigation utilities for applications built with 'bs4Dash' and 'shiny'. Supports configurable sidebar brand display modes, hover-expand behavior, and theme customization using CSS variables. Includes complete navbar item helpers, navbar structure validation, reusable brand configuration, prototype top-navigation support, and helpers for common navigation bar and footer layouts.

License MIT + file LICENSE

URL <https://github.com/PrigasG/bs4Dashkit>,
<https://PrigasG.github.io/bs4Dashkit/>

BugReports <https://github.com/PrigasG/bs4Dashkit/issues>

Imports bs4Dash, htmltools, jsonlite, shiny

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr, rmarkdown

Config/testthat/edition 3

Encoding UTF-8

Language en-US

RoxygenNote 7.3.3

NeedsCompilation no

Author George Arthur [aut, cre]

Maintainer George Arthur <prigasgenthian48@gmail.com>

Repository CRAN

Date/Publication 2026-07-13 07:10:13 UTC

Contents

bs4dashkit_demo_app	2
bs4dashkit_example_app	3
bs4dashkit_theme_presets	3
dash_brand_options	4
dash_brand_ui	5
dash_footer	6
dash_nav_help_button	7
dash_nav_help_item	7
dash_nav_item	8
dash_nav_refresh_button	9
dash_nav_refresh_item	9
dash_nav_status_badge	10
dash_nav_status_item	11
dash_nav_title	11
dash_titles	12
dash_titles_from	14
dash_topnav_options	15
dash_user_menu	16
use_bs4Dashkit	17
use_bs4Dashkit_core	17
use_dash_sidebar_behavior	19
use_dash_sidebar_brand_divider	19
use_dash_sidebar_brand_mode	20
use_dash_theme	21
use_dash_theme_preset	23
use_dash_topnav	23
validate_bs4dash_navbar	25
Index	27

bs4dashkit_demo_app	<i>Interactive bs4Dashkit demo app</i>
---------------------	--

Description

Returns a richer runnable shiny.appobj for testing sidebar brand states, hover-expand behavior, theme presets, navbar controls, and footer styling in one place.

Usage

```
bs4dashkit_demo_app()
```

Value

A shiny.appobj.

`bs4dashkit_example_app`*Minimal bs4Dashkit example app*

Description

Returns a tiny runnable shiny.appobj that demonstrates the recommended dash_titles() plus use_bs4Dashkit_core() flow.

Usage

```
bs4dashkit_example_app()
```

Value

A shiny.appobj.

`bs4dashkit_theme_presets`*Built-in bs4Dashkit theme presets*

Description

Returns the theme presets shipped with bs4Dashkit so they are easy to discover in code completion, validation errors, and documentation.

Usage

```
bs4dashkit_theme_presets(values = FALSE)
```

Arguments

values	Logical. If TRUE, also returns the preset theme values as a list-column named tokens.
--------	---

Value

A data frame with one row per preset and columns preset and description. If values = TRUE, the returned data frame also includes a tokens list-column.

dash_brand_options *Brand options for bs4Dashkit titles*

Description

Creates a reusable brand configuration that can be converted to `dash_titles()` output with `dash_titles_from()`.

Usage

```
dash_brand_options(
  name,
  icon = NULL,
  collapsed = NULL,
  expanded = NULL,
  app_name = NULL,
  collapsed_mode = NULL,
  expanded_mode = NULL,
  ...
)
```

Arguments

name	Visible brand label.
icon	Font Awesome icon name or simple shiny::icon() tag.
collapsed	Short label used when the sidebar brand is collapsed.
expanded	Label used when the sidebar brand is expanded.
app_name	Browser tab title.
collapsed_mode	Sidebar brand mode when collapsed. If NULL, dash_titles() uses its normal default.
expanded_mode	Sidebar brand mode when expanded. If NULL, dash_titles() uses its normal default.
...	Additional options forwarded to dash_titles() by dash_titles_from().

Value

A bs4dashkit_brand_options object.

dash_brand_ui	Create sidebar / navbar brand UI
---------------	----------------------------------

Description

Produces an icon + text block for bs4Dash brand slots with rich styling and optional visual effects. Supports Font Awesome icons or image logos.

Usage

```
dash_brand_ui(
  brand_text,
  icon = NULL,
  icon_img = NULL,
  icon_shape = c("circle", "rounded", "square"),
  icon_size = NULL,
  icon_color = NULL,
  weight = 700,
  spacing = "-0.02em",
  size = NULL,
  italic = FALSE,
  font_family = NULL,
  color = NULL,
  gradient = NULL,
  effect = c("none", "glow", "shimmer", "emboss"),
  glow_color = NULL
)
```

Arguments

brand_text	Visible brand label (character). May be NULL for icon-only brands.
icon	Font Awesome icon name e.g. "shield-halved", or a simple icon("shield-halved") tag. NULL for none.
icon_img	URL or www-relative path to an image logo. Overrides icon when both are supplied.
icon_shape	Shape mask for image icons: "circle", "rounded", "square". Ignored for FA icons. Default "circle".
icon_size	Size of icon/image as CSS string e.g. "20px", "1.3em". NULL = inherits sidebar default.
icon_color	CSS color for FA icon e.g. "#2f6f8f". NULL = inherit. For image icons, applies a CSS tint via mix-blend-mode (subtle).
weight	CSS font-weight: numeric or keyword. Default 700.
spacing	CSS letter-spacing. Default "-0.02em".
size	CSS font-size for label e.g. "13px". NULL = inherit.

<code>italic</code>	Logical. Italicise the label. Default FALSE.
<code>font_family</code>	CSS font-family string e.g. "'Inter', sans-serif".
<code>color</code>	Solid CSS color for label text. NULL = inherit. Ignored when gradient is set.
<code>gradient</code>	Character vector of 2 hex colors for gradient text effect e.g. <code>c("#2f6f8f", "#5ba3c9")</code> . Automatically enables <code>effect = "gradient"</code> .
<code>effect</code>	Visual effect on label: "none", "glow", "shimmer", "emboss". "gradient" is set automatically when gradient is supplied.
<code>glow_color</code>	CSS color for glow effect. Defaults to first gradient color or color or the package accent blue.

Value

A named list with components:

ui A shiny.tag.list containing the generated brand UI.

dep A shiny.tag object containing scoped CSS dependencies for the brand styling.

<code>dash_footer</code>	<i>Standard footer for bs4Dash dashboards</i>
--------------------------	---

Description

Standard footer for bs4Dash dashboards

Usage

```
dash_footer(
  logo_src = NULL,
  left_text = NULL,
  right_text = NULL,
  right_date = NULL,
  date_format = "%B %d, %Y",
  logo_height = 20,
  logo_position = c("left", "right"),
  fixed = TRUE
)
```

Arguments

<code>logo_src</code>	Path under <code>www/</code> or an external URL.
<code>left_text</code>	Text rendered on the left side (optional).
<code>right_text</code>	Text rendered on the right side (optional).
<code>right_date</code>	Optional Date to render on the right. Ignored if <code>right_text</code> is set.
<code>date_format</code>	Format used when <code>right_date</code> is provided.
<code>logo_height</code>	Height in px for the logo image.
<code>logo_position</code>	Where the logo should appear: "left" or "right".
<code>fixed</code>	Logical. Fixed footer (TRUE/FALSE).

Value

A shiny.tag object representing a bs4DashFooter UI component.

dash_nav_help_button	<i>Navbar help button (styled)</i>
----------------------	------------------------------------

Description

Navbar help button (styled)

Usage

```
dash_nav_help_button(  
  id,  
  label = "Help",  
  icon = "circle-question",  
  class = NULL,  
  ...  
)
```

Arguments

id	inputId for actionButton
label	Button label
icon	Font Awesome icon name
class	Additional classes
...	Passed to shiny::actionButton

Value

A shiny.tag object representing a styled navbar action button.

dash_nav_help_item	<i>Navbar help item (complete bs4Dash rightUi child)</i>
--------------------	--

Description

Navbar help item (complete bs4Dash rightUi child)

Usage

```
dash_nav_help_item(  
  id,  
  label = "Help",  
  icon = "circle-question",  
  class = NULL,  
  ...  
)
```

Arguments

id	inputId for actionButton
label	Button label
icon	Font Awesome icon name
class	Additional classes
...	Passed to shiny::actionButton

Value

A complete <li class="nav-item dropdown"> navbar child for use in bs4DashNavbar(rightUi = ...).

dash_nav_item	<i>Wrap a navbar control in a list item (bs4Dash rightUi convention)</i>
---------------	--

Description

Wrap a navbar control in a list item (bs4Dash rightUi convention)

Usage

```
dash_nav_item(...)
```

Arguments

...	UI elements
-----	-------------

Value

A shiny.tag object containing an HTML list item for use in bs4Dash navbar UI.

dash_nav_refresh_button
Navbar refresh button (styled)

Description

Navbar refresh button (styled)

Usage

```
dash_nav_refresh_button(
  id,
  label = "Refresh",
  icon = "rotate-right",
  class = NULL,
  ...
)
```

Arguments

id	inputId for actionButton
label	Button label
icon	Font Awesome icon name
class	Additional classes
...	Passed to shiny::actionButton

Value

A shiny.tag object representing a styled navbar refresh button.

dash_nav_refresh_item *Navbar refresh item (complete bs4Dash rightUi child)*

Description

Navbar refresh item (complete bs4Dash rightUi child)

Usage

```
dash_nav_refresh_item(
  id,
  label = "Refresh",
  icon = "rotate-right",
  class = NULL,
  ...
)
```

Arguments

id	inputId for actionButton
label	Button label
icon	Font Awesome icon name
class	Additional classes
...	Passed to shiny::actionButton

Value

A complete `<li class="nav-item dropdown">` navbar child for use in `bs4DashNavbar(rightUi = ...)`.

dash_nav_status_badge *Navbar status badge*

Description

Navbar status badge

Usage

```
dash_nav_status_badge(
  label,
  status = c("success", "primary", "secondary", "info", "warning", "danger", "light",
    "dark"),
  icon = NULL,
  class = NULL
)
```

Arguments

label	Badge label.
status	Bootstrap status color. One of "primary", "secondary", "success", "info", "warning", "danger", "light", or "dark".
icon	Optional Font Awesome icon name or simple shiny::icon() tag.
class	Additional classes.

Value

A shiny.tag badge suitable for wrapping in dash_nav_item().

dash_nav_status_item *Navbar status item (complete bs4Dash rightUi child)*

Description

Navbar status item (complete bs4Dash rightUi child)

Usage

```
dash_nav_status_item(
  label,
  status = c("success", "primary", "secondary", "info", "warning", "danger", "light",
    "dark"),
  icon = NULL,
  class = NULL
)
```

Arguments

label	Badge label.
status	Bootstrap status color. One of "primary", "secondary", "success", "info", "warning", "danger", "light", or "dark".
icon	Optional Font Awesome icon name or simple shiny::icon() tag.
class	Additional classes.

Value

A complete <li class="nav-item dropdown"> navbar child for use in bs4DashNavbar(rightUi = ...).

dash_nav_title *Navbar title (center, right, or left)*

Description

Creates a consistent title block for bs4Dash navbars, with optional subtitle and icon. Alignment can be "center", "right", or "left".

Usage

```
dash_nav_title(
  title,
  subtitle = NULL,
  icon = NULL,
  align = c("center", "right", "left")
)
```

Arguments

title	Main title (character or tag)
subtitle	Optional subtitle (character or tag)
icon	Optional Font Awesome icon name or simple icon("shield-halved") tag.
align	One of c("center","right","left")

Details

IMPORTANT:

- align = "left" is intended to be placed in dashboardHeader(leftUi = ...)
- align = "right" is intended to be placed in dashboardHeader(rightUi = ...)
- align = "center" can be placed anywhere (it is positioned by CSS/JS)

Value

A shiny.tag object representing the navbar title UI.

dash_titles	<i>Dashboard brand and sidebar configuration</i>
-------------	--

Description

Single entry point for branding and sidebar brand behavior. Creates a reusable object you place into bs4DashPage(), bs4DashNavbar(), and bs4DashBody().

Usage

```
dash_titles(  
  brand_text,  
  app_name = NULL,  
  icon = NULL,  
  icon_img = NULL,  
  icon_shape = c("circle", "rounded", "square"),  
  icon_size = NULL,  
  icon_color = NULL,  
  weight = 700,  
  spacing = "-0.02em",  
  size = NULL,  
  italic = FALSE,  
  font_family = NULL,  
  color = NULL,  
  gradient = NULL,  
  effect = c("none", "glow", "shimmer", "emboss"),  
  glow_color = NULL,  
  collapsed = NULL,
```

```

    expanded = NULL,
    collapsed_text = NULL,
    expanded_text = NULL,
    collapsed_text_size = NULL,
    expanded_text_size = NULL,
    collapsed_text_weight = NULL,
    expanded_text_weight = NULL,
    brand_divider = NULL,
    debug = NULL
)

```

Arguments

brand_text	Visible brand label shown in the navbar/sidebar. May be NULL only when both sidebar modes are "icon-only" and an icon or image logo is supplied.
app_name	Browser tab title. If NULL, defaults to brand_text, or "bs4Dashkit" when brand_text is empty.
icon	Font Awesome icon name for the brand (for example "project-diagram") or a simple icon("project-diagram") tag. Ignored when icon_img is supplied.
icon_img	Path (www relative) or URL to an image logo. Overrides icon.
icon_shape	Shape mask for image logos. One of "circle", "rounded", "square".
icon_size	CSS size for the icon/image (e.g. "20px", "1.2em").
icon_color	CSS color for the Font Awesome icon. For image icons, a subtle tint may be applied.
weight	CSS font-weight for the brand text.
spacing	CSS letter-spacing for the brand text.
size	CSS font-size for the brand text (e.g. "14px").
italic	Logical. If TRUE, renders brand text in italics.
font_family	CSS font-family string (e.g. "'Inter', sans-serif").
color	Solid CSS text color for the brand label (ignored when gradient is set).
gradient	Character vector of length 2 giving gradient colors for the brand label. When set, gradient styling is applied.
effect	Visual effect for the brand label. One of "none", "glow", "shimmer", "emboss". If gradient is set, gradient styling is used.
glow_color	Color used for glow/shimmer effects (when applicable).
collapsed	Sidebar brand mode when the sidebar is collapsed. One of "icon-only", "icon-text", "text-only". If NULL, uses option bs4Dashkit.sidebar.collapsed.
expanded	Sidebar brand mode when the sidebar is expanded. One of "icon-only", "icon-text", "text-only". If NULL, uses option bs4Dashkit.sidebar.expanded.
collapsed_text	Short label used in collapsed mode (recommended <= 8 chars).
expanded_text	Label used in expanded mode (recommended <= 30 chars).
collapsed_text_size	Optional CSS font-size applied only to the sidebar brand label while collapsed (for example "11px" or "0.75rem").

expanded_text_size	Optional CSS font-size applied only to the sidebar brand label while expanded.
collapsed_text_weight	Optional CSS font-weight applied only to the sidebar brand label while collapsed.
expanded_text_weight	Optional CSS font-weight applied only to the sidebar brand label while expanded.
brand_divider	Logical. If TRUE, shows a divider under the brand block. If NULL, uses option <code>bs4Dashkit.brand_divider</code> .
debug	Logical. If TRUE, emits console warnings for missing icons, etc. If NULL, uses option <code>bs4Dashkit.debug</code> .

Value

A named list with components:

app_name A character string for use in `bs4DashPage(title = ...)`.

brand A `shiny.tag` object for use in `bs4DashNavbar(title = ...)`. In a standard `bs4DashPage()`, the sidebar brand mirrors the navbar brand automatically.

deps A `shiny.tag.list` containing CSS and JavaScript dependencies to include once in `bs4DashBody(...)`.

dash_titles_from	<i>Create dashboard titles from brand options</i>
------------------	---

Description

Create dashboard titles from brand options

Usage

```
dash_titles_from(brand, ...)
```

Arguments

brand	A <code>bs4dashkit_brand_options</code> object created by <code>dash_brand_options()</code> .
...	Additional arguments passed to <code>dash_titles()</code> , overriding values stored in <code>brand</code> .

Value

A named list with `app_name`, `brand`, and `deps`; see `dash_titles()`.

dash_topnav_options *Top navigation options*

Description

Bundles the top-navigation settings used by `use_bs4Dashkit_core(layout = "topnav")`. This keeps app setup compact while preserving the same options accepted by `use_dash_topnav()`.

Usage

```
dash_topnav_options(
  align = c("left", "center", "right"),
  gap = 0,
  style = c("underline", "pill", "compact"),
  mobile = c("collapse", "scroll"),
  overflow = c("auto", "more", "scroll"),
  more_after = Inf,
  title = c("auto", "show", "compact", "hide"),
  page_title = c("none", "tab"),
  brand = TRUE,
  debug = FALSE
)
```

Arguments

<code>align</code>	Horizontal alignment of the mirrored menu inside the navbar: "left" (default), "center", or "right".
<code>gap</code>	Space between mirrored top-nav items. Numeric values are treated as pixels; CSS lengths such as "0.5rem" are also accepted.
<code>style</code>	Visual style for top-nav tabs: "underline" (default), "pill", or "compact".
<code>mobile</code>	Mobile behavior: "collapse" (default) shows a hamburger that opens the mirrored tabs; "scroll" keeps tabs in one horizontally scrollable row.
<code>overflow</code>	Desktop overflow behavior. "auto" (default) keeps the tab row intact. "more" moves items after <code>more_after</code> into a More dropdown. "scroll" allows horizontal scrolling.
<code>more_after</code>	Number of top-level items to keep visible when <code>overflow = "more"</code> . Use <code>Inf</code> to disable moving items.
<code>title</code>	How a centered <code>dash_nav_title()</code> behaves when top-nav tabs need space: "auto" (default) compacts then hides as needed, "show" always shows it, "compact" hides the subtitle, and "hide" hides it in top-nav mode. In <code>mobile = "scroll"</code> , "auto" hides the centered title earlier so the tab row has room; "show" is allowed but may crowd the scroll row.
<code>page_title</code>	Optional content title synced from the active top-nav tab. Use "none" (default) to disable or "tab" to show the selected tab label above the content area.

brand	Logical. If TRUE (default), the sidebar brand (icon + label) is mirrored into the navbar. Ignored when bs4Dash::bs4DashSidebar(disable = TRUE) already placed the brand in the navbar.
debug	Logical. If TRUE, prints diagnostic messages to the browser console when the navbar or sidebar menu cannot be found.

Value

A named list with class bs4dashkit_topnav_options.

Examples

```
dash_topnav_options(  
  align = "left",  
  gap = 6,  
  style = "compact",  
  mobile = "collapse",  
  overflow = "more",  
  more_after = 4,  
  title = "auto",  
  page_title = "tab"  
)
```

dash_user_menu	<i>Standard wrapper for a navbar user menu</i>
----------------	--

Description

Standard wrapper for a navbar user menu

Usage

```
dash_user_menu(ui)
```

Arguments

ui	A UI object, typically a dropdown menu produced by the app
----	--

Value

A shiny.tag object representing the user menu UI for the navbar.

use_bs4Dashkit	<i>Load core CSS and JS for bs4Dashkit</i>
----------------	--

Description

Adds the package's bundled CSS (core, theme, sidebar) and optional JS helpers. All assets are attached as named `htmltools::htmlDependency` objects, so they are included once per page regardless of how many `bs4Dashkit` helpers request them.

Usage

```
use_bs4Dashkit(include_center_js = TRUE)
```

Arguments

`include_center_js`

Logical. If `TRUE` (default), also includes the JavaScript helper that supports the centered navbar title layout.

Value

A `htmltools::tagList` containing `htmltools::htmlDependency` objects for the package CSS and optional JavaScript helpers.

use_bs4Dashkit_core	<i>Load bs4Dashkit core dependencies in one call</i>
---------------------	--

Description

Recommended entry point for `bs4Dashkit` in `bs4DashBody()`. For most apps, this should be the first element inside `bs4DashBody(...)`.

Usage

```
use_bs4Dashkit_core(
  ttl,
  preset = NULL,
  accent = NULL,
  ...,
  topbar_h = 56,
  collapsed_w = 4.2,
  expanded_w = 250,
  layout = c("sidebar", "topnav"),
  topnav = NULL,
  topnav_align = c("left", "center", "right"),
  topnav_gap = 0,
```

```

topnav_style = c("underline", "pill", "compact"),
topnav_mobile = c("collapse", "scroll"),
topnav_overflow = c("auto", "more", "scroll"),
topnav_more_after = Inf,
topnav_title = c("auto", "show", "compact", "hide"),
topnav_page_title = c("none", "tab"),
topnav_brand = TRUE,
topnav_debug = FALSE
)

```

Arguments

<code>ttitle</code>	Output from <code>dash_titles()</code>
<code>preset</code>	Optional theme preset name (e.g. "professional"). If NULL, uses option <code>bs4Dashkit.theme_preset</code> (if set).
<code>accent</code>	Optional accent override. If NULL, uses option <code>bs4Dashkit.accent</code> . If preset is used, this overrides the preset accent.
<code>...</code>	Optional overrides passed to <code>use_dash_theme_preset()</code> when preset is used.
<code>topbar_h</code>	Height for topbar + brand strip. Numeric values are treated as pixels; CSS lengths such as "3.5rem" are also accepted.
<code>collapsed_w</code>	Sidebar collapsed width. Numeric values are treated as rem; CSS lengths such as "4.25rem" are also accepted.
<code>expanded_w</code>	Sidebar expanded width. Numeric values are treated as pixels; CSS lengths such as "270px" are also accepted.
<code>layout</code>	Layout behavior to apply. "sidebar" keeps the default bs4Dash sidebar with hover-expand behavior. "topnav" hides the sidebar and mirrors its menu into the navbar with <code>use_dash_topnav()</code> .
<code>topnav</code>	Optional output from <code>dash_topnav_options()</code> . When supplied, its values are used for the top-navigation settings below.
<code>topnav_align</code>	Horizontal alignment for the mirrored top-nav menu when <code>layout = "topnav"</code> : "left", "center", or "right". If a centered <code>dash_nav_title()</code> is also present, "left" or "right" usually leaves the title more readable.
<code>topnav_gap</code>	Space between mirrored top-nav items. Numeric values are treated as pixels; CSS lengths such as "0.5rem" are also accepted.
<code>topnav_style</code>	Visual style for top-nav tabs: "underline", "pill", or "compact".
<code>topnav_mobile</code>	Mobile behavior: "collapse" shows a hamburger for tabs; "scroll" keeps a horizontal tab row.
<code>topnav_overflow</code>	Desktop overflow behavior: "auto", "more", or "scroll".
<code>topnav_more_after</code>	Number of top-level items kept visible when <code>topnav_overflow = "more"</code> .
<code>topnav_title</code>	How centered <code>dash_nav_title()</code> behaves when tabs need space: "auto", "show", "compact", or "hide".
<code>topnav_page_title</code>	Optional content title synced from the active top-nav tab: "none" or "tab".

topnav_brand	Logical. If TRUE, mirror the sidebar brand into the navbar when bs4Dash has not already placed one there.
topnav_debug	Logical. If TRUE, print top-nav diagnostics to the browser console.

Value

A shiny.tag.list containing core CSS and/or JavaScript dependencies for bs4Dashkit.

use_dash_sidebar_behavior

Sidebar behavior: icon-only collapse + hover expand + ellipsis labels

Description

Sidebar behavior: icon-only collapse + hover expand + ellipsis labels

Usage

```
use_dash_sidebar_behavior(topbar_h = 56, collapsed_w = 4.2, expanded_w = 250)
```

Arguments

topbar_h	Height for navbar and sidebar brand strip. Numeric values are treated as pixels; CSS lengths such as "3.5rem" are also accepted.
collapsed_w	Width for icon-only sidebar. Numeric values are treated as rem; CSS lengths such as "4.25rem" are also accepted.
expanded_w	Expanded sidebar width. Numeric values are treated as pixels; CSS lengths such as "270px" are also accepted.

Value

A shiny.tag or shiny.tag.list containing sidebar behavior dependencies to include in the UI.

use_dash_sidebar_brand_divider

Toggle the sidebar header divider (line under the brand area)

Description

Toggle the sidebar header divider (line under the brand area)

Usage

```
use_dash_sidebar_brand_divider(show = TRUE)
```

Arguments

show Logical. TRUE keeps divider (default). FALSE hides it.

Value

If show is TRUE, an empty `shiny.tag.list`. If show is FALSE, a `shiny.tag` object containing CSS that hides the sidebar brand divider.

use_dash_sidebar_brand_mode

Set sidebar brand display mode for expanded and collapsed states

Description

Controls icon visibility, text visibility, and text content in both collapsed and expanded sidebar states.

Usage

```
use_dash_sidebar_brand_mode(
  icon = NULL,
  collapsed = c("icon-only", "icon-text", "text-only"),
  expanded = c("icon-text", "icon-only", "text-only"),
  collapsed_text = NULL,
  expanded_text = NULL,
  collapsed_text_size = NULL,
  expanded_text_size = NULL,
  collapsed_text_weight = NULL,
  expanded_text_weight = NULL,
  debug = FALSE
)
```

Arguments

icon	Font Awesome icon name used in the brand, e.g. "shield-halved", or a simple <code>icon("shield-halved")</code> tag. Required when any mode includes "icon".
collapsed	Display mode when sidebar is collapsed: "icon-only", "icon-text", or "text-only"
expanded	Display mode when sidebar is expanded: "icon-text", "icon-only", or "text-only"
collapsed_text	Short label for collapsed "icon-text" or "text-only" mode. Keep ≤ 8 chars. NULL falls back to the current brand label already present in the DOM. When using <code>dash_titles()</code> , the higher-level helper usually resolves this to <code>brand_text</code> for visible text modes.
expanded_text	Label for expanded "icon-text" or "text-only" mode. Keep ≤ 30 chars. NULL falls back to the current brand label already present in the DOM. When using <code>dash_titles()</code> , the higher-level helper usually resolves this to <code>brand_text</code> for visible text modes.

collapsed_text_size	Optional CSS font-size applied to the sidebar brand label while collapsed.
expanded_text_size	Optional CSS font-size applied to the sidebar brand label while expanded.
collapsed_text_weight	Optional CSS font-weight applied to the sidebar brand label while collapsed.
expanded_text_weight	Optional CSS font-weight applied to the sidebar brand label while expanded.
debug	Logical. If TRUE, prints helpful messages to the browser console for diagnosing missing icons or brand label elements.

Details

Modes:

- "icon-only" : icon shown, label hidden — requires icon
- "text-only" : label shown, icon hidden — collapsed_text / expanded_text optional
- "icon-text" : both shown — requires icon; text args optional

Value

A shiny.tag or shiny.tag.list containing sidebar brand mode styling or dependencies.

use_dash_theme	<i>Apply a theme via CSS variables</i>
----------------	--

Description

Sets bs4Dashkit CSS custom properties (variables) for background, surfaces, borders, text, and accent color.

Usage

```
use_dash_theme(
  bg = "#f5f6f8",
  surface = "#ffffff",
  border = "#e2e3e7",
  text = "#1d1f23",
  muted = "#6b6f76",
  accent = "#2f6f8f",
  accent_soft = "#e8f1f5",
  navbar_bg = "#ffffff",
  navbar_text = "#1d1f23",
  sidebar_bg = "#ffffff",
  sidebar_text = "#1d1f23",
  sidebar_hover = "#eef4f7",
```

```

    footer_bg = "#ffffff",
    footer_text = "#6b6f76",
    success = "#2d8a56",
    warning = "#c37a14",
    danger = "#b94a48",
    radius = 12,
    shadow = "0 1px 3px rgba(0,0,0,0.07)"
)

```

Arguments

bg	Page background color.
surface	Card and panel background color.
border	Border color used on cards, separators, and outlines.
text	Primary text color.
muted	Muted text color.
accent	Accent color used for highlights and emphasis.
accent_soft	Softer accent tone used for subtle fills, hover states, and focus treatments.
navbar_bg	Navbar background color.
navbar_text	Navbar text color.
sidebar_bg	Sidebar background color.
sidebar_text	Sidebar text color.
sidebar_hover	Sidebar hover and active background color.
footer_bg	Footer background color.
footer_text	Footer text color.
success	Success accent color.
warning	Warning accent color.
danger	Danger accent color.
radius	Corner radius in pixels.
shadow	Box shadow CSS string used for cards and surfaces.

Value

A shiny.tag.list containing stylesheet dependencies and inline CSS variables for the app theme.

use_dash_theme_preset *Apply a theme preset*

Description

Apply a theme preset

Usage

```
use_dash_theme_preset(preset = c("professional", "modern", "dark-lite"), ...)
```

Arguments

preset	One of: "professional", "modern", "dark-lite"
...	Passed to use_dash_theme() to override preset values

Value

A shiny.tag or shiny.tag.list containing theme preset CSS and related UI dependencies.

use_dash_topnav *Top navigation layout: replace the sidebar with a navbar menu*

Description

Turns a regular bs4Dash app into a top-navigation app, similar to bslib::page_navbar(). The sidebar is hidden and its menu is mirrored as a horizontal menu inside the navbar. Clicks on the horizontal menu are delegated back to the original sidebar links, so everything wired to the sidebar keeps working: input\$<sidebarMenu id>, bs4Dash::updateTabItems(), badges, and bookmarking.

Usage

```
use_dash_topnav(
  topbar_h = 56,
  align = c("left", "center", "right"),
  gap = 0,
  style = c("underline", "pill", "compact"),
  mobile = c("collapse", "scroll"),
  overflow = c("auto", "more", "scroll"),
  more_after = Inf,
  title = c("auto", "show", "compact", "hide"),
  page_title = c("none", "tab"),
  brand = TRUE,
  debug = FALSE
)
```

Arguments

topbar_h	Height for the navbar. Numeric values are treated as pixels; CSS lengths such as "3.5rem" are also accepted. The mirrored menu can be aligned with align. If you also place a centered dash_nav_title() in the navbar, the title is centered in the remaining space between the mirrored menu and right-side controls. In that case align = "left" or "right" usually gives the cleanest result; align = "center" is best for apps without a centered title or with only a small number of tabs.
align	Horizontal alignment of the mirrored menu inside the navbar: "left" (default), "center", or "right".
gap	Space between mirrored top-nav items. Numeric values are treated as pixels; CSS lengths such as "0.5rem" are also accepted.
style	Visual style for top-nav tabs: "underline" (default), "pill", or "compact".
mobile	Mobile behavior: "collapse" (default) shows a hamburger that opens the mirrored tabs; "scroll" keeps tabs in one horizontally scrollable row.
overflow	Desktop overflow behavior. "auto" (default) keeps the tab row intact. "more" moves items after more_after into a More dropdown. "scroll" allows horizontal scrolling.
more_after	Number of top-level items to keep visible when overflow = "more". Use Inf to disable moving items.
title	How a centered dash_nav_title() behaves when top-nav tabs need space: "auto" (default) compacts then hides as needed, "show" always shows it, "compact" hides the subtitle, and "hide" hides it in top-nav mode. In mobile = "scroll", "auto" hides the centered title earlier so the tab row has room; "show" is allowed but may crowd the scroll row.
page_title	Optional content title synced from the active top-nav tab. Use "none" (default) to disable or "tab" to show the selected tab label above the content area.
brand	Logical. If TRUE (default), the sidebar brand (icon + label) is mirrored into the navbar. Ignored when bs4Dash::bs4DashSidebar(disable = TRUE) already placed the brand in the navbar.
debug	Logical. If TRUE, prints diagnostic messages to the browser console when the navbar or sidebar menu cannot be found.

Details

Keep your app exactly as it is - bs4DashSidebar(), bs4SidebarMenu(), bs4TabItems() - and add use_dash_topnav() (or use_bs4Dashkit_core(layout = "topnav")) to the body. The sidebar stays in the DOM as the single source of truth but is never shown.

Menu items with sub-items (bs4SidebarMenuSubItem()) are rendered as navbar dropdowns. Menus rendered on the fly (renderMenu()) and updates via updateTabItems() are picked up automatically.

On screens narrower than 992px the menu stays on one horizontally scrollable row instead of collapsing into a hamburger.

Value

A shiny.tag.list with the top-nav CSS/JS dependencies and the initialization script. Include it once in bs4DashBody(...).

Examples

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

  ttl <- dash_titles(brand_text = "Top-nav demo", icon = icon("bolt"))

  ui <- bs4DashPage(
    title = ttl$app_name,
    header = bs4DashNavbar(title = ttl$brand),
    sidebar = bs4DashSidebar(
      bs4SidebarMenu(
        id = "sidebar",
        bs4SidebarMenuItem("Home", tabName = "home", icon = icon("house")),
        bs4SidebarMenuItem("Data", tabName = "data", icon = icon("table"))
      )
    ),
    body = bs4DashBody(
      use_bs4Dashkit_core(ttl, preset = "professional", layout = "topnav"),
      bs4TabItems(
        bs4TabItem(tabName = "home", "Home content"),
        bs4TabItem(tabName = "data", "Data content")
      )
    )
  )

  shinyApp(ui, function(input, output, session) {})
```

 validate_bs4dash_navbar

Validate bs4Dash navbar rightUi structure

Description

Checks that each direct child supplied to bs4DashNavbar(rightUi = ...) is a complete navbar list item with the dropdown class expected by bs4Dash/AdminLTE.

Usage

```
validate_bs4dash_navbar(rightUi)
```

Arguments

`rightUi` A `shiny::tagList()` or tag intended for `bs4DashNavbar(rightUi = ...)`.

Value

Invisibly returns TRUE when valid; otherwise errors with a targeted message.

Index

[bs4dashkit_demo_app](#), [2](#)
[bs4dashkit_example_app](#), [3](#)
[bs4dashkit_theme_presets](#), [3](#)

[dash_brand_options](#), [4](#)
[dash_brand_ui](#), [5](#)
[dash_footer](#), [6](#)
[dash_nav_help_button](#), [7](#)
[dash_nav_help_item](#), [7](#)
[dash_nav_item](#), [8](#)
[dash_nav_refresh_button](#), [9](#)
[dash_nav_refresh_item](#), [9](#)
[dash_nav_status_badge](#), [10](#)
[dash_nav_status_item](#), [11](#)
[dash_nav_title](#), [11](#)
[dash_titles](#), [12](#)
[dash_titles_from](#), [14](#)
[dash_topnav_options](#), [15](#)
[dash_user_menu](#), [16](#)

[use_bs4Dashkit](#), [17](#)
[use_bs4Dashkit_core](#), [17](#)
[use_dash_sidebar_behavior](#), [19](#)
[use_dash_sidebar_brand_divider](#), [19](#)
[use_dash_sidebar_brand_mode](#), [20](#)
[use_dash_theme](#), [21](#)
[use_dash_theme_preset](#), [23](#)
[use_dash_topnav](#), [23](#)

[validate_bs4dash_navbar](#), [25](#)