

Package: micromodal (via r-universe)

October 7, 2026

Title Create Simple and Elegant Modal Dialogs in 'shiny'

Version 1.0.1

Description Enables you to create accessible modal dialogs, with confidence and with minimal configuration.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

URL <https://kennedymwavu.github.io/micromodal/>

BugReports <https://github.com/kennedymwavu/micromodal/issues>

Imports htmltools (>= 0.5.5)

Suggests bslib (>= 0.5.0), knitr (>= 1.43), rmarkdown (>= 2.23), shiny (>= 1.7.4.1)

VignetteBuilder knitr

Config/roxygen2/version 8.0.0

Repository <https://kennedymwavu.r-universe.dev>

Date/Publication 2026-07-09 13:58:19 UTC

RemoteUrl <https://github.com/kennedymwavu/micromodal>

RemoteRef HEAD

RemoteSha 1a91df350a4c368b39c10711516cf3e64e94c094

Contents

get_dependencies	2
micromodal	2
use_micromodal	5

Index	6
--------------	----------

get_dependencies	<i>Get package dependencies</i>
------------------	---------------------------------

Description

Get package dependencies

Usage

```
get_dependencies()
```

micromodal	<i>Create a modal dialog</i>
------------	------------------------------

Description

Create a modal dialog

Usage

```
micromodal(id, title = NULL, content = NULL, footer = NULL)
```

Arguments

id	A unique id for the modal
title	The modal's title
content	The modal's content
footer	The modal's footer

Value

```
htmltools::tagList()
```

Examples

```
# Example 1----
if (interactive()) {
  library(shiny)
  library(micromodal)

  ui <- fluidPage(
    use_micromodal(),
    actionButton(
      inputId = "trigger",
      label = "Trigger modal",
      `data-micromodal-trigger` = "modal-1"
```

```

    ),
    micromodal(
      id = "modal-1",
      title = "Hello, World!",
      content = tagList(
        tags$p("Hi Mom,"),
        tags$p("Come see my first modal!")
      )
    )
  )
}

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

shinyApp(ui, server)
}

# Example 2----
if (interactive()) {
  library(shiny)
  library(micromodal)

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

  }

  ui <- bslib::page(
    title = "Micromodal",
    theme = bslib::bs_theme(version = 5),
    # inform shiny to use {micromodal}:
    use_micromodal(),
    # your normal UI code:
    tags$div(
      class = "container",
      tags$div(
        class = "container my-5",
        tags$h1("Micromodal.js in Shiny", class = "mb-5"),
        # trigger for "modal-1"
        actionButton(
          inputId = "show_modal_1",
          label = "Exhibit 1",
          class = "btn-outline-primary px-3",
          `data-micromodal-trigger` = "modal-1"
        ),
        # trigger for "modal-2"
        actionButton(
          inputId = "show_modal_2",
          label = "Exhibit 2",
          class = "btn-outline-primary px-3",
          `data-micromodal-trigger` = "modal-2"
        )
      ),
    ),
    # modal-1:
    micromodal(

```

```

id = "modal-1",
title = "Login",
content = tagList(
  textInput(
    inputId = "name",
    label = "Name",
    width = "400px"
  ),
  tagAppendAttributes(
    passwordInput(
      inputId = "password",
      label = tags$div(
        tags$span("Password"),
        tags$span("(required)", class = "text-muted fw-light")
      ),
      width = "400px"
    ),
    class = "mb-0"
  ),
  tags$div(
    "Must be atleast 6 characters long.",
    class = "text-muted fw-light"
  )
),
footer = tagList(
  tags$button(
    class = "modal__btn modal__btn-primary",
    "Continue"
  ),
  tags$a(
    href = "#",
    class = "ms-3",
    `data-micromodal-close` = NA,
    `aria-label` = "Close this dialog window",
    "Cancel"
  )
),
# modal-2:
micromodal(
  id = "modal-2",
  title = "Micromodal",
  content = tagList(
    tags$p("This is a completely accessible modal."),
    tags$p(
      "Try hitting the",
      tags$code("tab"),
      "key* and notice how the focus stays",
      "within the modal itself. To close modal hit the",
      tags$code("esc"),
      "button, click on the overlay or just click the close button."
    ),
    tags$p("*", tags$code("alt + tab"), "in safari")
  )
)

```

```

    ),
    footer = tagList(
      tags$button(
        class = "modal__btn modal__btn-primary",
        "Continue"
      ),
      tags$button(
        class = "modal__btn ms-2",
        `data-micromodal-close` = NA,
        `aria-label` = "Close this dialog window",
        "Close"
      )
    )
  )
)
)
)
shinyApp(ui, server)
}

```

use_micromodal

Use micromodal

Description

Call this function once in your app's UI.

Usage

```
use_micromodal()
```

Details

This function adds the dependencies needed for the modals.

See [micromodal\(\)](#) for a complete example.

Value

`htmltools::tagList()`

Examples

```

if (interactive()) {
  library(shiny)
  library(micromodal)

  ui <- fluidPage(
    use_micromodal(),
    # the rest of your UI code
  )
}

```

Index

`get_dependencies`, [2](#)

`htmltools::tagList()`, [2](#), [5](#)

`micromodal`, [2](#)

`micromodal()`, [5](#)

`use_micromodal`, [5](#)