

Package: shinyhtr (via r-universe)

June 11, 2026

Type Package

Title Progress Bars for Downloads in 'shiny' Apps

Version 1.1.0

Description Modifies the progress() function from 'htr' package to let it send output to progressBar() function from 'shinyWidgets' package. It is just a tweak at the original functions from 'htr' package to make it smooth for 'shiny' developers.

License MIT + file LICENSE

BugReports <https://github.com/curso-r/shinyhtr/issues>

URL <https://github.com/curso-r/shinyhtr>

Encoding UTF-8

LazyData true

Imports shinyWidgets, utils, htr

RoxygenNote 7.1.1

Enhances shiny

NeedsCompilation no

Author Athos Damiani [aut, cre], Hadley Wickham [aut], RStudio [cph]

Maintainer Athos Damiani <athos.damiani@curso-r.com>

Config/pak/sysreqs cmake make libuv1-dev libssl-dev zlib1g-dev

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

Date/Publication 2026-06-11 01:02:00 UTC

RemoteUrl <https://github.com/cranhaven/cranhaven.r-universe.dev>

RemoteRef package/shinyhtr

RemoteSha 2a8e5bc597d3821277c0d0cfea821ff443421580

RemoteSubdir shinyhtr

Contents

progress	2
progress_bar	3
runExample	4
Index	6

progress	<i>Add a progress bar compatible with 'shinyWidgets::updateProgressBar()'.</i>
----------	--

Description

Add a progress bar to request just like the vanilla `'httr::progress()'` but with capability to talk to `'shinyWidgets::updateProgressBar()'` by giving the session and id arguments to it.

Usage

```
progress(
  session,
  id,
  type = c("down", "up"),
  con = stdout(),
  title = NULL,
  status = NULL,
  range_value = NULL,
  unit_mark = "%"
)
```

Arguments

session	(from <code>'shinyWidgets::updateProgressBar()'</code> doc) The 'session' object passed to function given to shinyServer.
id	(from <code>'shinyWidgets::updateProgressBar()'</code> doc) An id used to update the progress bar.
type	(from <code>'httr::progress()'</code> doc) Type of progress to display: either number of bytes uploaded or downloaded.
con	(from <code>'httr::progress()'</code> doc) Connection to send output too. Usually <code>stdout()</code> or <code>stderr</code> .
title	(from <code>'shinyWidgets::updateProgressBar()'</code> doc) character, optional title.
status	(from <code>'shinyWidgets::updateProgressBar()'</code> doc) Color, must be a valid Bootstrap status : primary, info, success, warning, danger.
range_value	(from <code>'shinyWidgets::updateProgressBar()'</code> doc) Default is to display percentage ([0, 100]), but you can specify a custom range, e.g. -50, 50.
unit_mark	(from <code>'shinyWidgets::updateProgressBar()'</code> doc) Unit for value displayed on the progress bar, default to "%".

See Also

[progress](#), [progressBar](#), [updateProgressBar](#)

Examples

```
if (interactive()) {  
  
  library(shiny)  
  library(shinyWidgets)  
  library(shinyhttr)  
  
  ui <- fluidPage(  
  
    sidebarLayout(  
  
      NULL,  
  
      mainPanel(  
        actionButton('download', 'Download 100MB file...'),  
        tags$p("see R console to compare both progress bars."),  
        progressBar(  
          id = "pb",  
          value = 0,  
          title = "",  
          display_pct = TRUE  
        )  
      )  
    )  
  )  
)  
  
  server <- function(input, output, session) {  
    observeEvent(input$download, {  
      httr::GET(  
        url = "https://speed.hetzner.de/100MB.bin",  
        progress(session, id = "pb")  
      )  
    })  
  }  
  
  shinyApp(ui, server)  
}
```

progress_bar

progress_bar

Description

Same as `'httr::progress_bar()'` but with capability to talk to `'shinyWidgets::progressBar()'`.

Usage

```

progress_bar(
  type,
  con,
  session,
  id,
  title = NULL,
  status = NULL,
  range_value = NULL,
  unit_mark = "%"
)

```

Arguments

type	(from 'httr::progress()' doc) Type of progress to display: either number of bytes uploaded or downloaded.
con	(from 'httr::progress()' doc) Connection to send output too. Usually stdout() or stderr.
session	(from 'shinyWidgets::updateProgressBar()' doc) The 'session' object passed to function given to shinyServer.
id	(from 'shinyWidgets::updateProgressBar()' doc) An id used to update the progress bar.
title	(from 'shinyWidgets::updateProgressBar()' doc) character, optional title.
status	(from 'shinyWidgets::updateProgressBar()' doc) Color, must be a valid Bootstrap status : primary, info, success, warning, danger.
range_value	(from 'shinyWidgets::updateProgressBar()' doc) Default is to display percentage ([0, 100]), but you can specify a custom range, e.g. -50, 50.
unit_mark	(from 'shinyWidgets::updateProgressBar()' doc) Unit for value displayed on the progress bar, default to "%".

Value

a function with rules to print out the progress.

See Also

[progress](#), [progressBar](#)

runExample

runExample

Description

Launch shiny example application using shinyhttr::progress_bar. This app also uses module to show that it works with it too.

Usage

```
runExample(display.mode = "showcase")
```

Arguments

`display.mode` The mode in which to display the example. Defaults to showcase, but may be set to normal to see the example without code or commentary.

Index

progress, [2](#), [3](#), [4](#)

progress_bar, [3](#)

progressBar, [3](#), [4](#)

runExample, [4](#)

updateProgressBar, [3](#)