Package 'shinydrive'

October 14, 2022

Type Package

Title File Sharing Shiny Module

Version 0.1.3

Description Shiny module for easily sharing files between users. Admin can add, remove, edit and download file.
User can only download file. It's also possible to manage files using R functions directly.

License GPL-3

Encoding UTF-8

RoxygenNote 7.1.1

Imports htmltools, shiny, tools, yaml, DT, R.utils, knitr

Suggests testthat

NeedsCompilation no

Author Benoit Thieurmel [aut, cre],
Thibaut Dubois [aut]

Maintainer Benoit Thieurmel <bthieurmel@gmail.com>

Repository CRAN

Date/Publication 2022-09-29 06:10:02 UTC

R topics documented:

| add_file_in_dir | *Add / suppress / edit a file* |

Description

Add / suppress / edit a file

Usage

```
add_file_in_dir(
  file,
  dir,
  yml,
  name,
  description = "",
  date_time_format = "%Y%m%d_%H%M%s"
)

edit_file_in_dir(
  id,
  dir,
  yml,
  name = NULL,
  description = NULL,
  file = NULL,
  date_time_format = "%Y%m%d_%H%M%s"
)

suppress_file_in_dir(id, dir, yml)

get_yaml_info(
  yml,
  recorded_name = TRUE,
  date_time_format = "%Y%m%d_%H%M%s",
  add_img = FALSE,
  img_size = 30
)
```

Arguments

file	character file (to copy) path.
dir	character directory path.
yml	character yaml configuration file full path.
name	character file name.
description	character file description.

```
date_time_format
                character DateTime format.
id              character file id in yaml.
recorded_name   logical : add recorded name (with date_time extension) in output ?
add_img         logical : Use in shiny module for adding file extension img.
img_size        integer : Use in shiny module for adding file extension img.
```

Value

These functions return a `logical` indicating if operation succeeded or not

Examples

```
## Not run:

yml <- file.path(getwd(), "test_sfm/config.yml") # will be created

dir <- file.path(getwd(), "test_sfm")
dir.create(dir)

file <- system.file("translate/translate.csv", package = "shinydrive")

# add one first file
add_file_in_dir(
  file = file,
  dir = dir,
  yml = yml,
  name = "translate1",
  description = ""
)

yaml::yaml.load_file(yml)
list.files(dir)
get_yaml_info(yml)

# add second file
add_file_in_dir(
  file = file,
  dir = dir,
  yml = yml,
  name = "translate_2",
  description = "This is cool"
)

yaml::yaml.load_file(yml)
list.files(dir)
get_yaml_info(yml, recorded_name = F)

# modify first file
edit_file_in_dir(
  id = "2",
```

```
    dir = dir,
    yml = yml,
    name = "translate_2_mod",
    description = "So cool"
  )

  yaml::yaml.load_file(yml)
  list.files(dir)

  # suppress first file
  suppress_file_in_dir(id = "1", dir = dir, yml = yml)

  yaml::yaml.load_file(yml)
  list.files(dir)


  ## End(Not run)
```

---

shiny_drive_ui   *File management shiny module.*

Description

File management shiny module.

Usage

```
shiny_drive_ui(id)

shiny_drive_server(
  input,
  output,
  session,
  id,
  save_dir,
  dir_access = NULL,
  admin_user = TRUE,
  force_desc = FALSE,
  lan = "EN",
  file_translate = read.csv(system.file("translate/translate.csv", package =
    "shinydrive"), sep = ";", encoding = "UTF-8", check.names = FALSE),
  datatable_options = list(),
  yml = "files_desc.yaml",
  date_time_format = "%Y%m%d_%H%M%s"
)
```

Arguments

`id`	character. An ID string
`input`	shiny input
`output`	shiny input
`session`	shiny input
`save_dir`	`character/reactive`. Main directory of the files.
`dir_access`	`character/reactive` vector for dir(s) access. Default to `NULL` (all directories)
`admin_user`	`boolean/reactive` (TRUE). Admin user or not.
`force_desc`	`boolean/reactive` (FALSE). Force to add an entry description ?
`lan`	`character/reactive` ("EN"). Language to be used in the module (FR, EN and CN availabled... contributions are welcome :)).
`file_translate`	`data.frame/reactive` File for translation.
`datatable_options`	`list/reactive`. `DT::datatable` options argument.
`yml`	`characte/reactiver` yaml configuration file name.
`date_time_format`	character DateTime format.

Value

Shiny module without return value.

Examples

```
## Not run:

if(require(shiny)){
  ui <- fluidPage(
    shiny_drive_ui(id = "idm")
  )
  server <- function(input, output, session) {
    callModule(module = shiny_drive_server,
               id = "idm",
               session = session,
               admin_user = TRUE,
               save_dir =  getwd(),
               lan = "FR")
  }
  shinyApp(ui, server)
}


## End(Not run)
```

Index