
```
library(dplyr)
```

```
rladies_global %>%  
  filter(city == 'Rio de Janeiro')
```



Shiny Dashboard

- Aplicado a

fundos de

investimentos





1. Introdução



Introdução



Pacote que constrói web app via R;

Estrutura interface com o usuário quanto do processamento dos dados;

Rodamos um código em R em um servidor que envia páginas web, recebe informações do usuário e processa os dados.





Estrutura Básica

- `shinyApp(ui = ui, server = server)`

'Visualização'
Tudo que será apresentado para o usuário

`dashboardPage`

`dashboardHeader`

`dashboardSidebar`

`dashboardBody`

'Informação'
Os outputs que aparecerão no UI

Base de dados

Imagens

Textos

Gráficos

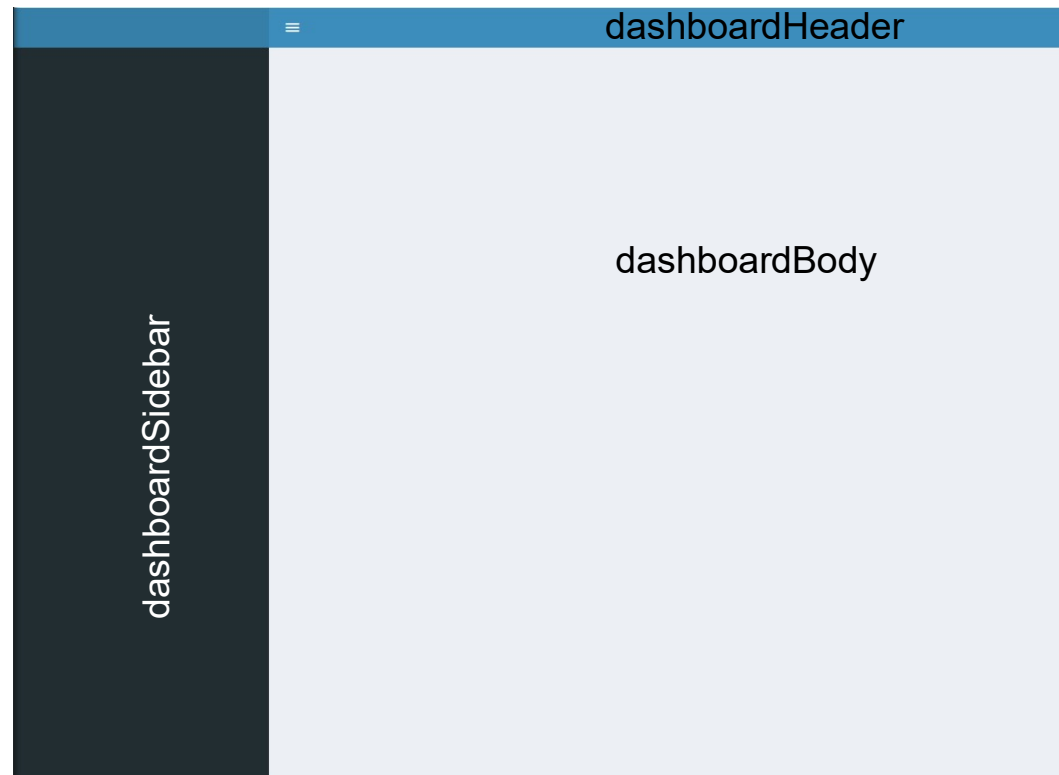




UI – User Side

```
ui <- dashboardPage(  
  
  dashboardHeader(),  
  
  dashboardSidebar(),  
  dashboardBody()  
)
```

dashboardPage





2. Inputs



Inputs



Integração do usuário com o app.

O usuário escolha um input que pode ser utilizado no app.

Button



`actionButton()`

Single checkbox



`checkboxInput()`

Checkbox group



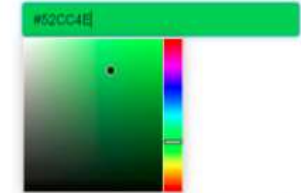
`checkboxGroupInput()`

Date input



`dateInput()`

Colour input



`colourpicker::colourInput()`

Date range



`dateRangeInput()`

File input



`fileInput()`

Numeric input



`numericInput()`

Password Input



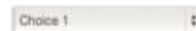
`passwordInput()`

Radio buttons



`radioButtons()`

Select box



`selectInput()`

Sliders



`sliderInput()`

Text input



`textInput()`

Text area



`textAreaInput()`



Inputs- Usados no APP de Fundos

```
radioButtons(inputId = "fundos",label = "",  
choices = unique((dados$NOME)),inline=T,  
selected = unique((dados$NOME))[1])
```





Inputs- Usados no APP de Fundos

```
dateRangeInput("simula_dadas", "Data  
Entrada", start = min(retornos_tb$date), end =  
max(retornos_tb$date), min =  
min(retornos_tb$date), max =  
max(retornos_tb$date), format = "yyyy-mm-  
dd")
```

```
actionButton("simula_botao",  
"Click here!")
```

Simulador dos Fundos

Escolha o Fundo

☒ ALASK BLACK ☐ TREND IBOVESPA ☐ WESTERN ASSET ☐ AZ QUEST ☐ TREND INFLAÇÃO

Data Entrada

2019-01-03 to 2019-12-31

Clica que calcula

Click here!



3. Output





Output

Base de Dados

`DTOutput('nome_base')`

`dataTableOutput()`

`tableOutput()`

Outros

`imageOutput()`

`textOutput()`

`htmlOutput()`

`valueBoxOutput()`

Gráficos

`PlotOutput("nome_grafico")`

`highchartOutput()`

`plotlyOutput()`





Server

- Instruções para gerar outputs a partir da interação do usuário no UI;
- Argumentos input e output;
- Render_()

```
server <- function(input, output) {  
  # Código  
}
```

Base de Dados

renderDataTable()

renderDT()

renderTable()

Outros

renderImage()

renderText()

renderUI()

renderValueBox()

Gráficos

renderPlot()

renderHighchart()

renderPlotly()



UI + Server

Ui

```
highchartOutput('desempenho_fundos')  
  
DTOutput('df_resumo')  
  
valueBoxOutput("rentabilidade_simulacao")
```

Server

```
output$desempenho_fundos <- renderHighchart({ })  
  
output$df_resumo <- renderDT({ })  
  
output$rentabilidade_simulacao <-  
  renderValueBox({ })
```



Rentabilidade por Fundos

date	Retorno a.p	Fundo
2019-12-31	20.54%	ALASK_BLACK
2019-12-31	11.17%	TREND_IBOVESPA
2019-12-31	21.38%	TREND_IBOVESPA
2019-12-31	1.12%	WESTERN_ASSET
2019-12-31	0.45%	AZ_QUEST
2019-12-31	4.20%	TREND_INFLAÇÃO
2019-12-31	11.84%	TREND_INFLAÇÃO

27.79%

Retorno Simulação





4. Reatividade





Reatividade



```
eventReactive(input$simula_botao, { })
```

Clica que calcula

Click here!

```
df_final_tt <- reactive({
```

```
  df_dados <- dados %>% dplyr::filter(NOME == input$fundos) %>% arrange(date) %>%  
  slice((1))  
})
```



5. App Fundos





Pacotes Favoritos

```
library(shiny)
library(shinydashboard)
library(DT)
library(tidyverse)
library(quadprog)
library(highcharter)
library(readxl)
```

```
library(plotly)
library(magrittr)
library(timeSeries)
library(corrplot)
library(tbl2xts)
library(bizdays)
library(PerformanceAnalytics)
library(rsconnect)
library(ggplot2)
```





Carregando base de dados

```
dados <- readRDS("input//dados.RDS")
```

```
retornos <- readRDS("input//retornos.RDS")
```

Name	Date modified	Type	Size
fundos	22/01/2020 22:31	File folder	
input	12/01/2020 12:36	File folder	
rsconnect	04/01/2020 20:37	File folder	
www	24/01/2020 21:11	File folder	
RData	18/01/2020 22:39	R Workspace	38 KB



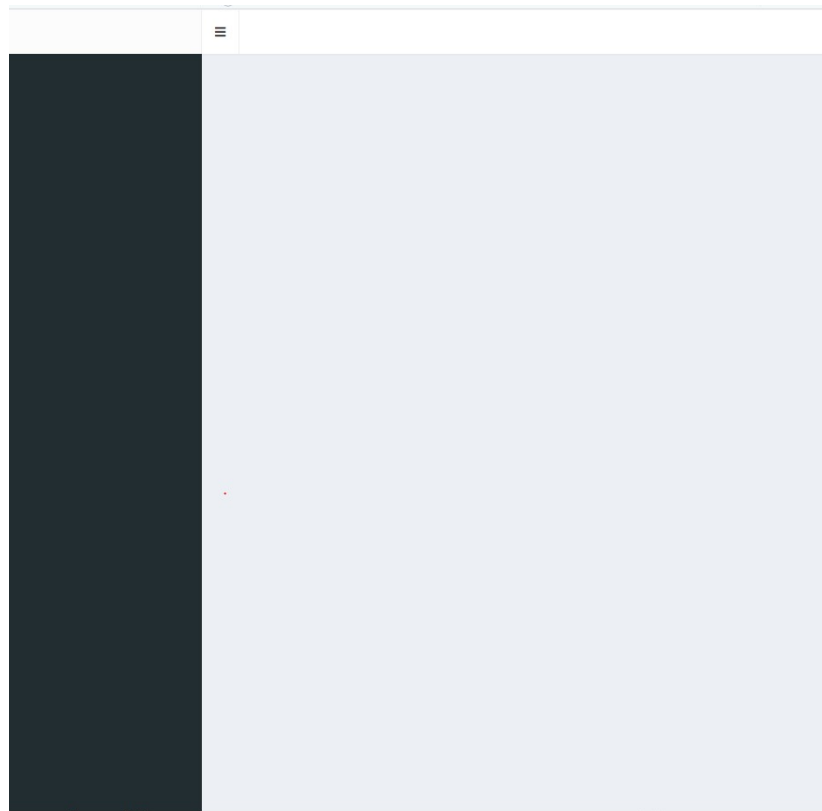
6. App Fundos - UI





UI – User Side

```
ui <- dashboardPage(  
  skin = "black",  
  dashboardHeader(),  
  dashboardSidebar(  
  ),  
  dashboardBody(  
  )  
)
```



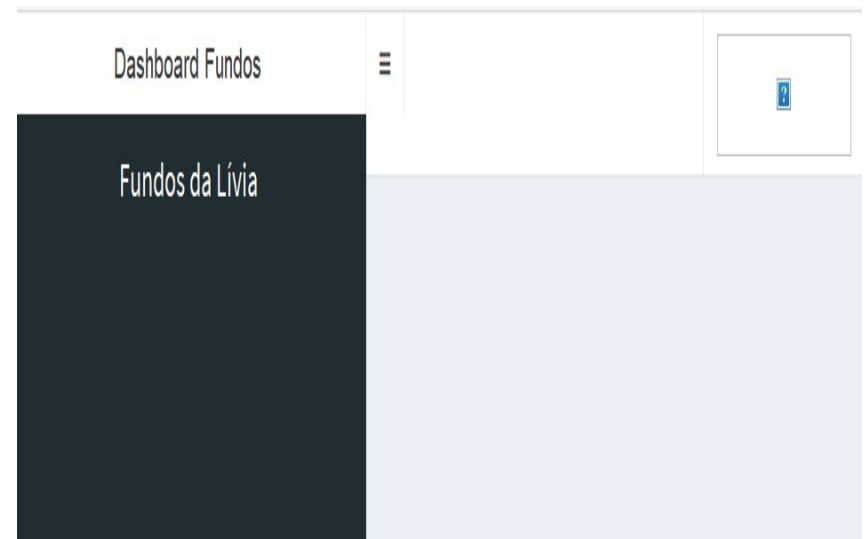
Skin = purple
Skin = red
Skin = green
Skin = yellow.





dashboardHeader

```
dashboardHeader(  
  title = "Dashboard Fundos", titleWidth = 400,  
  tags$li(a(href = 'https://games.crossfit.com/sanctionals',  
    img(src = 'bull.jpg', height='60', width='150'),  
    style = "padding-top:10px; padding-bottom:10px;"),  
  class = "dropdown")  
)
```

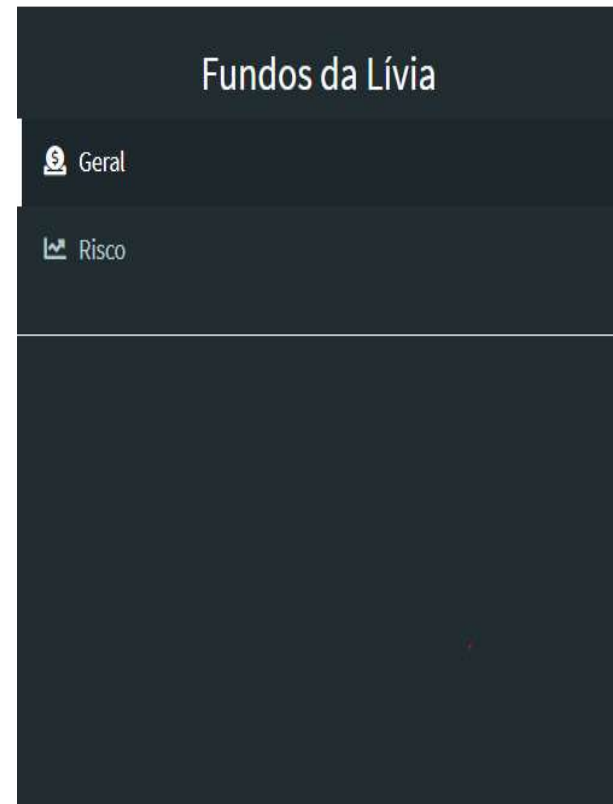


dashboardSidebar

```
dashboardSidebar(  
  width = 400,  
  h3('Fundos da Lívia', align = 'center'),  
  sidebarMenu(menuItem(text = 'Geral', icon = icon('donate'),  
    tabName = 'geral_fundos')),  
  
  sidebarMenu(menuItem(text = 'Risco', icon = icon('line-chart'),  
    tabName = 'risco')),  
  hr()  
)
```



Dashboard Fundos





dashboardBody

```
tabPanel("Simula Fundos",  
  fluidRow(h1('Simulador dos Fundos', align = 'center')),  
  br(),  
  br(),  
  radioButtons(inputId = "fundo_simula", label = " Escolha o Fundo", choices =  
unique((dados$NOME)), inline=T, selected = unique((dados$NOME))[1]),  
  dateRangeInput("simula_datas", "Data Entrada", start = min(retornos_tb$date), end =  
max(retornos_tb$date), min = min(retornos_tb$date), max = max(retornos_tb$date), format = "yyyy-  
mm-dd"),  
  h3("Clica que calcula"),  
  actionButton("simula_botao", "Click here!"),  
  fluidRow(column(9, highchartOutput('compara_fundos'))  
    ),  
  fluidRow(valueBoxOutput("rentabilidade_simulacao"),  
    valueBoxOutput("rentabilidade_livia"))  
)
```





7. App Fundos - Server





Geral – Nivel 1 Carteira (Gráfico)

```
# grafico de desempenho
output$desempenho <- renderHighchart({

  highchart(type = "stock") %>%
    hc_title(text = "Gráfico de Rentabilidade", style = list(color = "black", fontWeight = "bold")) %>%
    hc_add_series(sum_pat[,1],
                  name = names(sum_pat)[1], id= "fundo") %>%
    hc_scrollbar(enabled = FALSE) %>%
    hc_yAxis(opposite = FALSE)

})
```





Geral – Nivel 1 Carteira (Tabela)

```
output$df_resumo <- renderDT({  
  
  datatable(df_final_tt(), rownames = F, options = list(pageLength = 20, dom = 'pd')) %>%  
    formatPercentage(columns = 2, 2)  
})
```

date	Retorno a.p	Fundo
2019-12-31	20.54%	ALASK_BLACK
2019-12-31	11.17%	TREND_IBOVESPA
2019-12-31	21.38%	TREND_IBOVESPA
2019-12-31	1.12%	WESTERN_ASSET
2019-12-31	0.45%	AZ_QUEST
2019-12-31	4.20%	TREND_INFLAÇÃO
2019-12-31	11.84%	TREND_INFLAÇÃO



Geral – Nivel 2 Fundos (Gráfico)

```
# grafico de desempenho de cada fundo
output$desempenho_fundos<- renderHighchart({
```

```
  fundos_in <- input$fundos
  df_dados <- dados %>% dplyr::filter(NOME == fundos_in) %>% arrange(date) %>% slice((1))
# melhor debug de shiny
print(head(df_dados))
# dados$NOME
df_filtro <- retornos_tb %>%
  select(date,str_trim(str_replace_all(fundos_in," ","_"))) %>%
  dplyr::filter(date >= df_dados$date[1])
```

```
df_filtro_ts <- df_filtro%>% tbl_xts()
df_filtro_ts <- (cumprod(1 + df_filtro_ts)-1)
```

```
highchart(type = "stock") %>%
  hc_title(text = "Gráfico de Rentabilidade", style = list(color = "black", fontWeight = "bold")) %>%
  hc_add_series(100*df_filtro_ts[,1],
    name = names(df_filtro_ts)[1], id= "fundo") %>%
  hc_legend(enabled = TRUE) %>%
  hc_scrollbar(enabled = FALSE) %>%
  hc_yAxis(opposite = FALSE, labels = list(format = "{value}%"))
})
```



Geral – Simula Fundos

```
retornos_simula_app <- eventReactive(input$simula_botao, {

  retornos_tb %>%
    select(date, str_trim(str_replace_all(input$fundo_simula, " ", "_"))) %>%
    dplyr::filter(between(date, input$simula_datas[1], input$simula_datas[2]))

})

simulador_acum_ret <- reactive({

  simulador_acum_ret <- retornos_simula_app() %>% tbl_xts()
  simulador_acum_ret <- (cumprod(1 + simulador_acum_ret) - 1)
  simulador_acum_ret

})

output$rentabilidade_simulacao <- renderValueBox({
  valueBox(
    paste0(round(as.vector(simulador_acum_ret()[nrow(simulador_acum_ret())]) *
      100, 2), "%", sep = ""),
    'Retorno Simulacao'
    , icon = icon("star", lib = 'glyphicon')
    , color = "red")
})
```



```
output$compara_fundos <- renderHighchart({

  df_cum_ret <- simulador_acum_ret()

  highchart(type = "stock") %>%
    hc_title(text = "Gráfico de Rentabilidade", style = list(color =
      "black", fontWeight = "bold")) %>%
    hc_add_series(100 * df_cum_ret[, 1],
      name = names(df_cum_ret)[1], id = "fundo") %>%
    hc_legend(enabled = TRUE) %>%
    hc_scrollbar(enabled = FALSE) %>%
    hc_yAxis(opposite = FALSE, labels = list(format = "{value}%"))

})
```





Geral – Simula Fundos

tabela com medidas de risco de 21 dias

```
output$risco_21d <- DT::renderDataTable({datatable(kpis_riscos_21d, rownames = F
) %>%
  formatPercentage(columns = c(2,3,9),digits = 2) %>%
  formatCurrency(c(4,5,6,7,8), currency = "", interval = 3, mark = ",",
    digits = 2, dec.mark = getOption("OutDec"), before = TRUE)
})
```

tabela com medidas de risco de 63 dias

```
output$risco_63d <- DT::renderDataTable({datatable(kpis_riscos_63d, rownames = F
) %>%
  formatPercentage(columns = c(2,3,9),digits = 2) %>%
  formatCurrency(c(4,5,6,7,8), currency = "", interval = 3, mark = ",",
    digits = 2, dec.mark = getOption("OutDec"), before = TRUE)
})
```

21 Dias						
Fundo	Retorno aa	Vol aa	Sharpe aa	Observacoes	Assimetria	Curtose
WESTERN_ASSET	2.72%	3.65%	0.69	21.00	-0.06	-0.50
ALASKA_BLACK	2.06%	5.36%	0.39	21.00	0.23	-0.06
TREND_BOVESPA	1.67%	5.07%	0.39	21.00	0.05	-0.85
AZ_QUEST	1.85%	5.26%	0.35	21.00	0.22	-1.22
TREND_INFLAÇÃO	0.94%	0.56%	1.67	21.00	-1.12	0.88

Showing 1 to 5 of 5 entries

Previous 1 Next

63 Dias						
Fundo	Retorno aa	Vol aa	Sharpe aa	Observacoes	Assimetria	Curtose
WESTERN_ASSET	8.36%	6.65%	1.22	63.00	-0.48	0.54
ALASKA_BLACK	6.32%	9.29%	0.68	63.00	-0.10	0.16
TREND_BOVESPA	6.04%	8.79%	0.69	63.00	-0.81	1.16
AZ_QUEST	5.67%	9.12%	0.62	63.00	-0.50	0.28
TREND_INFLAÇÃO	2.85%	0.67%	2.62	63.00	-0.25	-0.13

Showing 1 to 5 of 5 entries

Previous 1 Next



8. Extra





Como Publicar?

- Shinyapps.io

Sistema R studio para hospedagem de aplicativos em Shiny.

- 1) Criar a conta no <https://www.shinyapps.io/>

- 1) Pegar seu token

- 2) <https://docs.rstudio.com/shinyapps.io/>

- 2) R local

- 1) `devtools::install_github("rstudio/rsconnect")`

- 2) `library(rsconnect)`

- 3) `rsconnect::setAccountInfo(name='XXXX', token='1XXX', secret=XXX')`



Como Publicar?

- Shiny Server

Programa gratuito que monta o servidor web em Linux feito para hospedar aplicativos em Shiny



Material Apoio

- [1] <https://www.curso-r.com/material/shiny/>
- [2] <https://deanattali.com/blog/building-shiny-apps-tutorial/>
- [3] https://rstudio.github.io/shinydashboard/get_started.html
- [4] <https://r4ds.had.co.nz/>
- [5] <https://pt.stackoverflow.com/>





Obrigada



Livia Rodrigues



<https://www.linkedin.com/in/l%C3%ADvia-rodriques-49554b93/>

