

ShinyWYSIWYG: a Shiny What You See Is What You Get editor

Lic. Juan Cruz Rodriguez
Ing. Claudio Vargas Rojas
Dr. Elmer A. Fernández

 /jcrodriguez1989
 @IACCancu

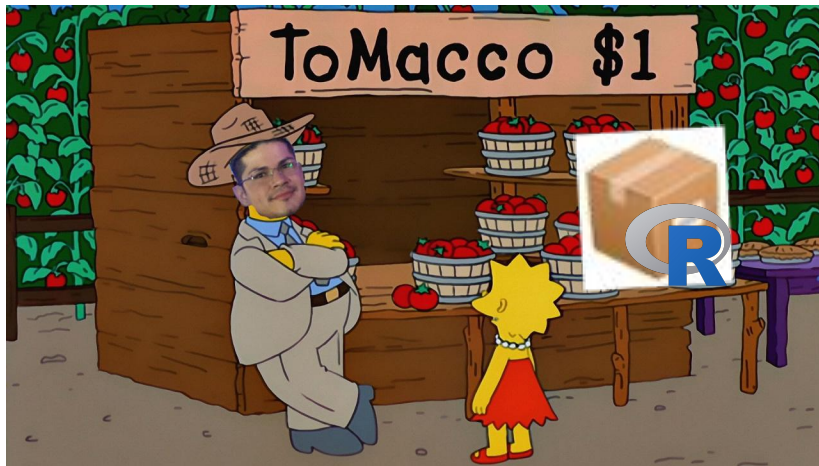


Motivation

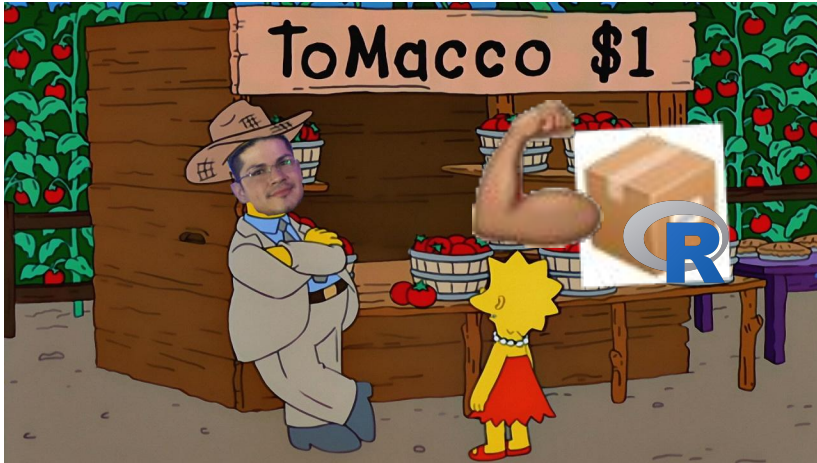
Motivation



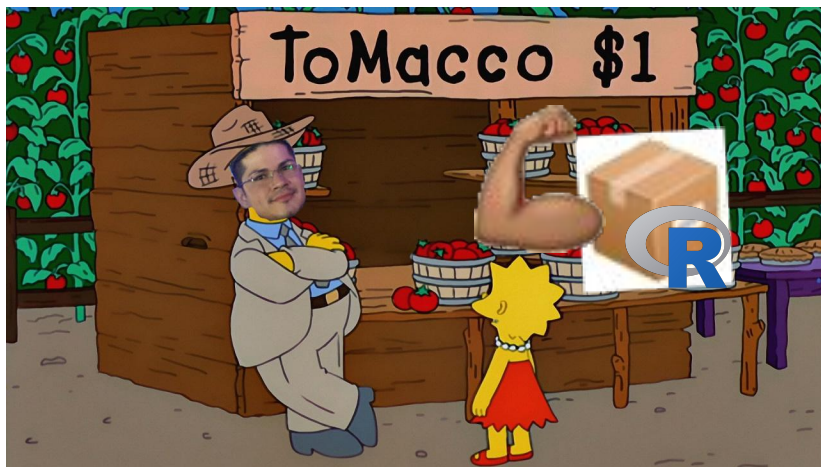
Motivation



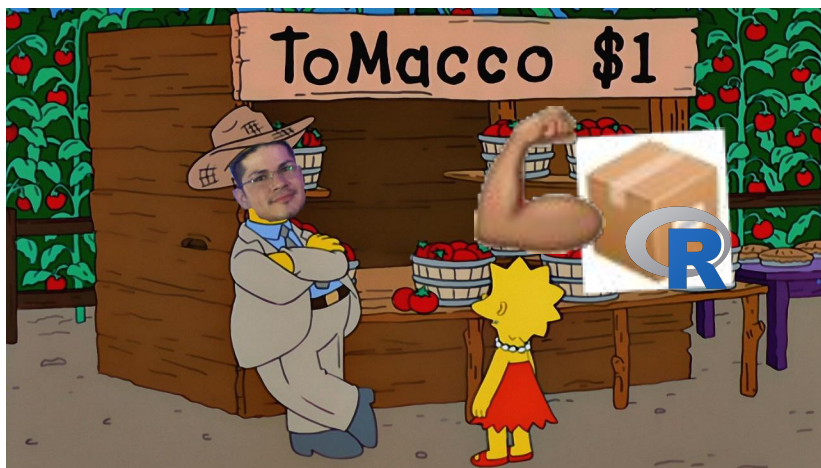
Motivation



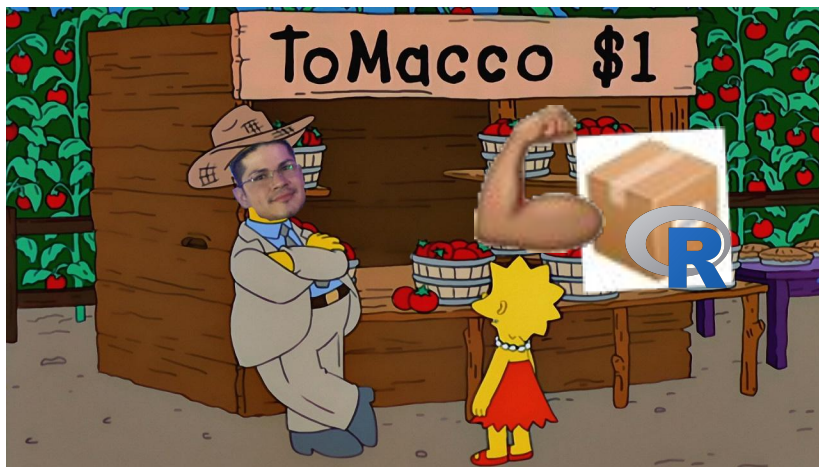
Motivation



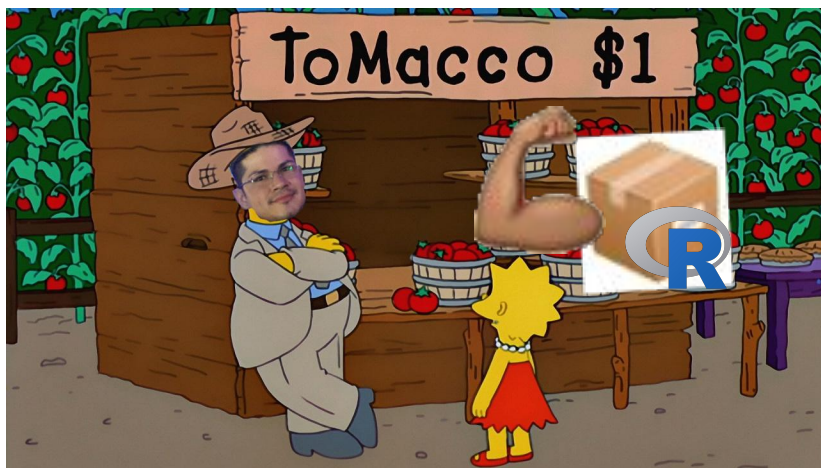
Motivation



Motivation



Motivation



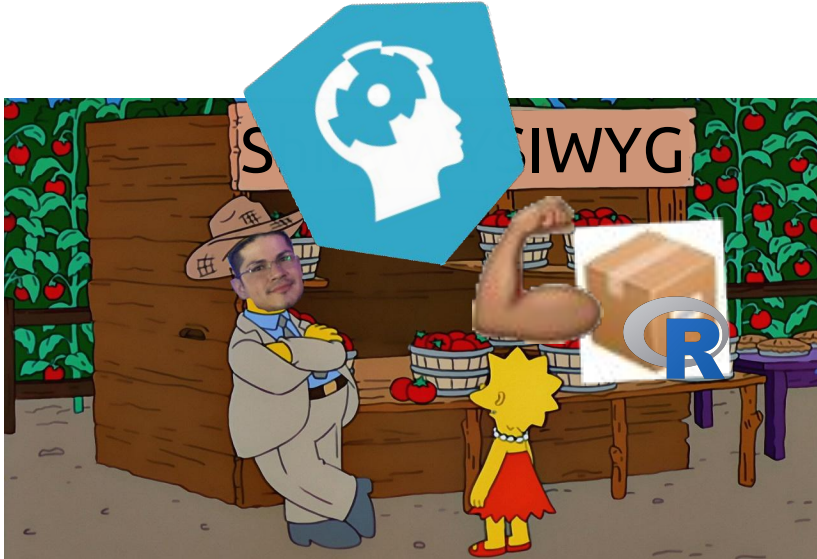
Motivation



Motivation



Motivation



About

- Create a Shiny UI by *Drag & Drop*

About

- Create a Shiny UI by *Drag & Drop*
- Minimalistic: Imports *ggplot2*, *shiny*, *shinyjs*

About

- Create a Shiny UI by *Drag & Drop*
- Minimalistic: Imports *ggplot2*, *shiny*, *shinyjs*
- R and Shiny version independent

getNamespaceExports ('shiny')



Shiny functions

Inputs

Outputs

Renderers

Installation

```
> library('devtools');  
  
> install_github('jcrodriguez1989/shinyWYSIWYG');  
  
>  
  
> library('shinyWYSIWYG');  
  
> shinyWYSIWYG();
```

ShinyWYSIWYG - UI

ID

Position:

Size:

Functions reference

dateInput

dateRangeInput

fileInput

numericInput

passwordInput

radioButtons

selectInput

selectizeInput

sliderInput

Canvas



4/10

ShinyWYSIWYG - UI

The image shows a screenshot of the ShinyWYSIWYG UI editor. On the left, there is a form with the following elements:

- ID:** A text input field containing "bins".
- Position:** Two numeric input fields, both containing "0".
- Size:** Two numeric input fields, containing "374" and "301".
- Buttons:** "Ok" and "De" (partially visible).

Below the form, there is a "Functions reference" panel. This panel is also shown as a larger, detailed inset in the center of the image. The inset panel has the title "Functions reference" and a section titled "Inputs". The list of input functions in the inset is:

- actionButton
- actionLink
- checkboxGroupInput
- checkboxInput
- dateInput
- dateRangeInput
- fileInput
- numericInput

The main "Functions reference" panel at the bottom left shows a scrollable list of input functions, with "sliderInput" currently selected and highlighted.

4/10

ShinyWYSIWYG - Server

Global variables

CreateDel

Events

01_hello

OkDel

Event
Name

01_hello

When

bins

With

bins

What

```
x <- faithful$waiting;  
bins <- seq(min(x), max(x), length.out=bins + 1);  
  
hist(x, breaks = bins, col = "#75AADB", border = "white",  
      xlab = "Waiting time to next eruption (in mins)",  
      main = "Histogram of waiting times")
```


Output

distPlot

Rendering

renderPlot

ShinyWYSIWYG - Server

Global variables

CreateDel

Event
Name
01_hello
When

When
bins
With
bins

OkDel

```
hist(x, breaks = bins, col = "#75AADB", border = "white",  
      xlab = "Waiting time to next eruption (in mins)",  
      main = "Histogram of waiting times")
```

Output
distPlot
Rendering
renderPlot

ShinyWYSIWYG - Server

Global variables		Event Name	
<div></div>			
What			
<pre>x <- faithful\$waiting; bins <- seq(min(x), max(x), length.out=bins + 1); hist(x, breaks = bins, col = "#75AADB", border = "white", xlab = "Waiting time to next eruption (in mins)", main = "Histogram of waiting times")</pre>			
Output		Rendering	
<div>distPlot</div>		<div>renderPlot</div>	
Output		Rendering	
<div>distPlot</div>		<div>renderPlot</div>	

ShinyWYSIWYG - Generate

Ok

Generate

Fill objects args.

Accept

sliderInput

inputId

"bins"

label

"Number of bins:"

min

1

max

50

value

30

ShinyWYSIWYG - Generate

Generate

```
library(shiny);

ui <- fluidPage(
  absolutePanel(wellPanel(sliderInput(sep=",", post=NULL, round=FALSE, pre=NULL, animate=FALSE,
dragRange=TRUE, locale=NULL, width=NULL, label="Number of bins:", min=1, step=NULL, ticks=TRUE,
timeFormat=NULL, inputId="bins", max=50, format=NULL, value=30, timezone=NULL)), top=0, height=301, left=0,
width=374),
  absolutePanel(wellPanel(plotOutput(hoverId=NULL, height="400px", inline=FALSE, width="100%", click=NULL,
clickId=NULL, outputId="distPlot", brush=NULL, dblclick=NULL, hover=NULL, hoverDelayType=NULL,
hoverDelay=NULL)), top=0, height=600, left=375, width=649)
)

server <- function(input, output) {
  # 01_hello
  observeEvent(input$bins, {
    output$distPlot <- renderPlot({
      bins <- input$bins;
      x <- faithful$waiting;
      bins <- seq(min(x), max(x), length.out=bins + 1);

      hist(x, breaks = bins, col = "#75AADB", border = "white",
          xlab = "Waiting time to next eruption (in mins)",
          main = "Histogram of waiting times")
    })
  })
}
```

ShinyWYSIWYG - Enjoy

```
$ mkdir myShinyApp;
```

```
$ cd myShinyApp;
```

```
$ vim app.R;
```

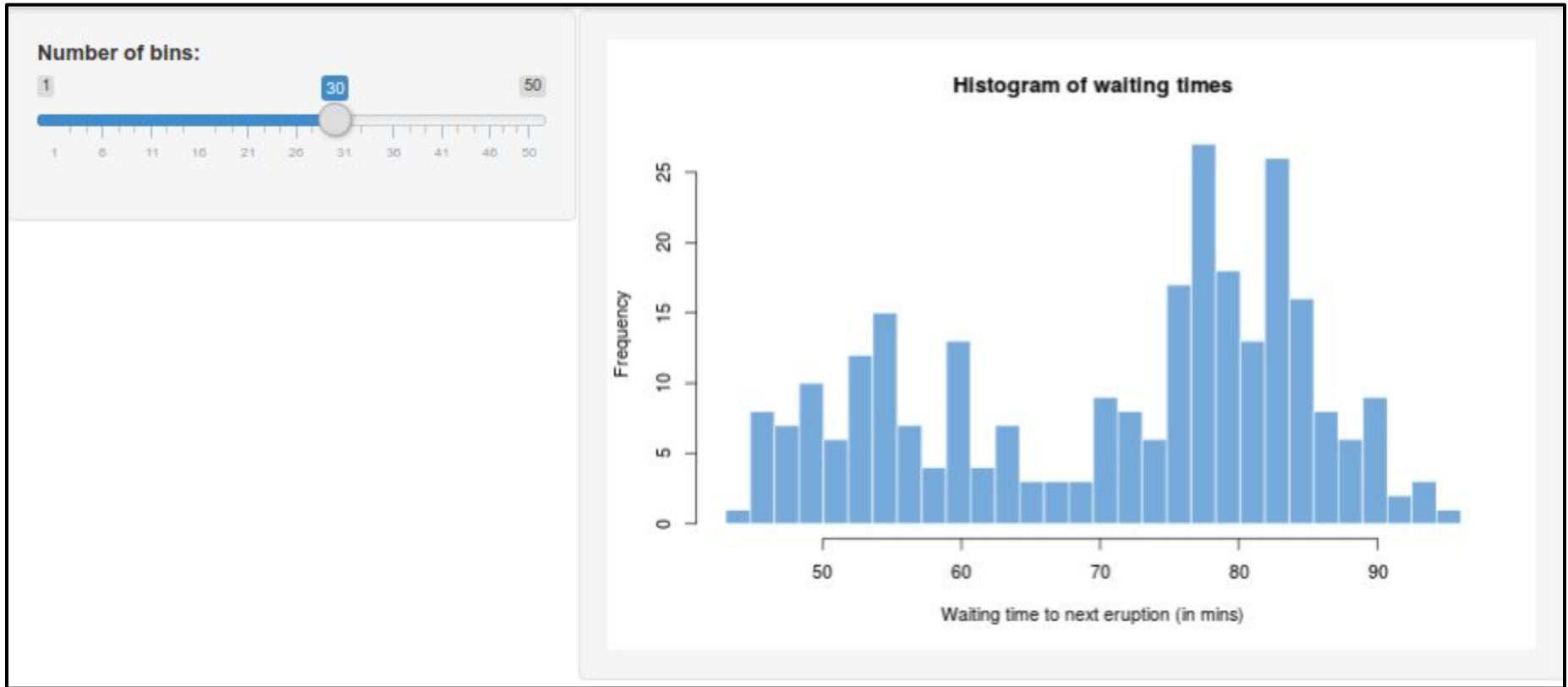
```
# Ctrl + V
```

```
$ R;
```

```
> library('shiny');
```

```
> runApp();
```

ShinyWYSIWYG - Enjoy



ShinyWYSIWYG - Enjoy

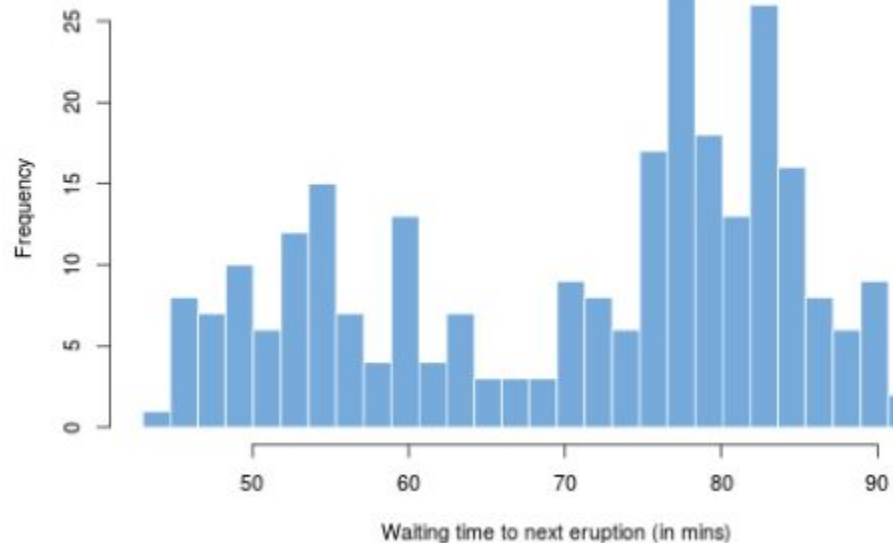
```
shiny::runExample('01_hello')
```

Hello Shiny!

Number of bins:



Histogram of waiting times



Limitations

- UI: Generates *absolutePanels*

Limitations

- UI: Generates *absolutePanels*
- Server: Uses *observeEvent*

Questions?

github.com/jcrodriguez1989/shinyWYSIWYG

— Lic. Juan Cruz Rodriguez
Ing. Claudio Vargas Rojas
Dr. Elmer A. Fernández

—  /jcrodriguez1989
 @IACCancu

