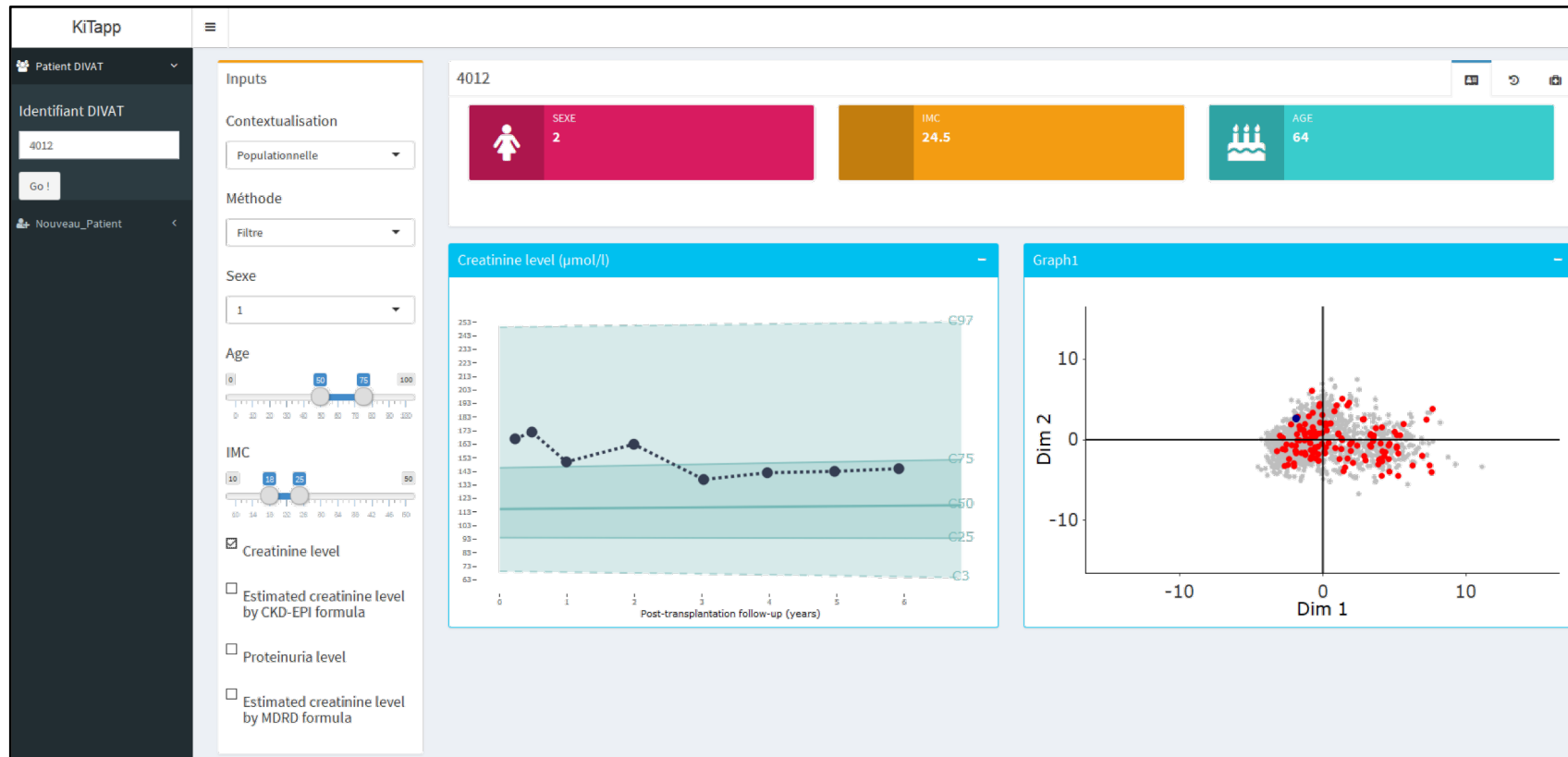


How to build a *Shiny* app



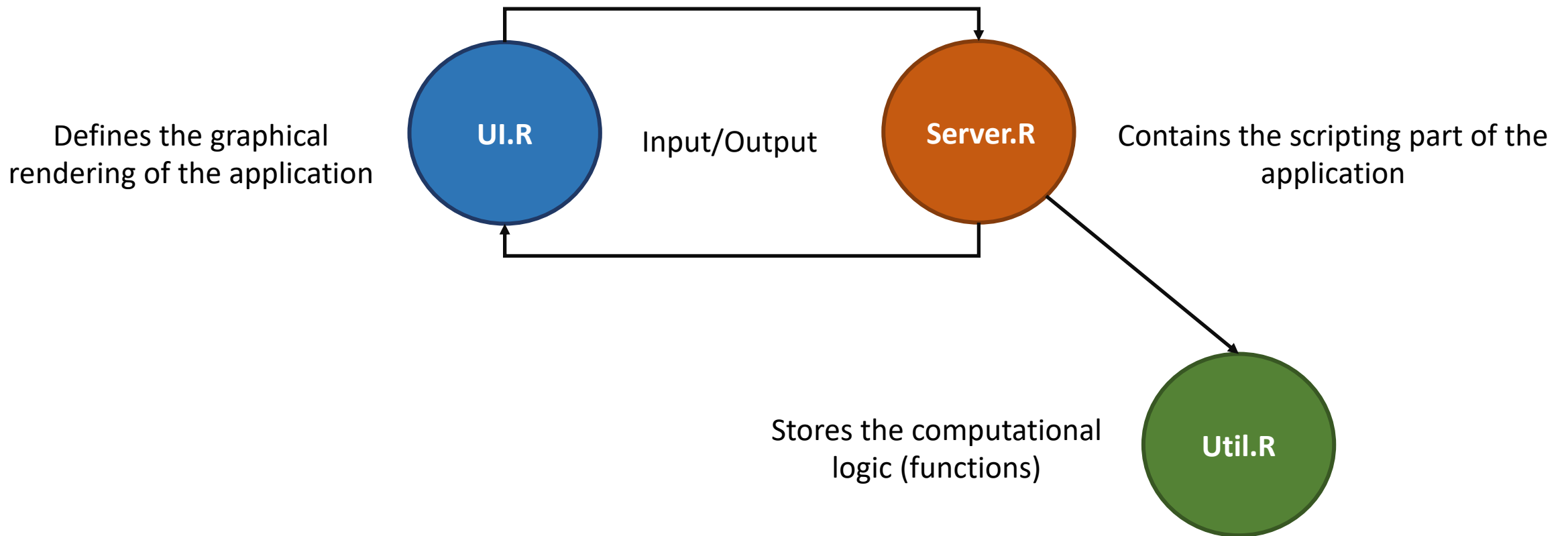
What's Shiny ?

- Open source R package from Rstudio
- Makes easy to build interactive web applications with R.



Shiny architecture

app.R



Shiny architecture : ui.R

- ui.R

```
library(shiny)
```

Loads the shiny package

```
shinyUI(fluidPage(
```

Defines UI.R as the shiny interface

```
  titlePanel("Tutorial Genetic Code"),
```

Sets a title for the app

```
  tabsetPanel(
```

Divides the app into independent panels

```
    # DNA sequence manipulation (length, GC-content, etc.)
```

```
    tabPanel("DNA sequence",
```

Defines a panel

```
      br(),
```

```
      h3("1 Load DNA sequence(s)",
```

```
      fileInput("dna.file", "Fasta file (e.g. with the mysterious sequence)"),
```

```
      br(),
```

```
      h3("2 DNA content per sequence"),
```

```
      dataTableOutput("dna.content")
```

```
    ),
```

```
    # DNA sequence translation
```

```
    tabPanel("Protein sequence", "contents"),
```

```
    # Protein sequence alignment
```

```
    tabPanel("Sequence alignment", "contents")
```

```
  )
```

```
)
```

http://127.0.0.1:3771 | Open in Browser

Publish

Basic widgets

Buttons

Action

Submit

Single checkbox

☒ Choice A

Checkbox group

☒ Choice 1

☐ Choice 2

☐ Choice 3

Date input

2014-01-01

Date range

2017-06-21 to 2017-06-21

File input

Browse... No file selected

Help text

Note: help text isn't a true widget, but it provides an easy way to add text to accompany other widgets.

Numeric input

1

Radio buttons

☒ Choice 1

☐ Choice 2

☐ Choice 3

Select box

Choice 1

Sliders

0 50 100

0 25 75 100

Text input

Enter text...

Shiny architecture : `server.R`

• `server.R`

```
library(shiny)
library(sequinr)
library(Biostrings)
```

Loads packages

```
# 'util' file for genetic code logic and strings cleaning
```

```
source("util.R")
```

Uses functions from `util.R`

```
# server logic to read fasta files
```

```
shinyServer(function(input, output) {
```

Defines the file as the app's server and takes input and output as arguments

```
  output$dna.content <- renderDataTable({

    # save inFile for internal logic
    inFile <- input$dna.file

    if (is.null(inFile))
      return(NULL)

    dna.sequences <- read.fasta(inFile$datapath, seqtype = "DNA", forceDNAtoLower = T)

    dna.content <- compute.dna.metrics(dna.sequences)

    return(dna.content)
  })
})
```

Processes the user input and provides output using functions from `util.R`

```
})
```

Shiny architecture : util.R

- util.R (store the non-server logic there)

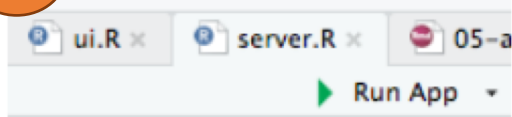
```
compute.dna.metrics <- function(dna.sequences){  
  # Bonus exercise: combine the 6 sapply calls  
  dna.content <- data.frame(names=names(dna.sequences))  
  
  dna.content$a <- sapply(dna.sequences, compute, 'a')  
  dna.content$c <- sapply(dna.sequences, compute, 'c')  
  dna.content$g <- sapply(dna.sequences, compute, 'g')  
  dna.content$t <- sapply(dna.sequences, compute, 't')  
  
  dna.content$length <- sapply(dna.sequences, length)  
  dna.content$GC.content <- sapply(dna.sequences, GC)  
  
  return(dna.content)  
}
```

Contains functions that will be used to process the user entries

```
compute <- function(seq, letter){  
  # Exercise 4.1 with the Exercise 2.1 function  
}  
  
codon.table <- read.table(file = "../TP-Student/Code_shortcut/Genetic_code.txt",  
  col.names = c("codon", "aa", "letter"),  
  stringsAsFactors = F)  
  
codon.table$codon <- tolower(codon.table$codon) # use only lower case characters  
rownames(codon.table) <- codon.table$codon # use codons as table keys  
  
dna2peptide <- function(sequence, frame=0, sens='F'){  
  # Exercise 6.4  
}
```

Run your Shiny app

1



2

```
library(shiny)
runApp("my_app")
```

3

```
# Run the app ----
shinyApp(ui = ui, server = server)
```

Output

Tutorial Genetic Code

DNA sequence Protein sequence Sequence alignment

1 Load DNA sequence(s)

Fasta file (e.g. with the mysterious sequence)

Browse... Mysterious_seq.txt

Upload complete

2 DNA content per sequence

Show 25 entries Search:

names	a	c	g	t	length	GC.content
seq	1378	1052	1175	1405	5010	0.444511

names a c g t length GC.content

Showing 1 to 1 of 1 entries Previous 1 Next

Console Terminal x

~/Projet_de_these_Rokhaya/2018_2019/Cours_centrale/R/R_S5/TP_R/tp-ecn/ ↗

STOP BUTTON !

Let's Start !!