

# SHINY

SELO

1/20/2021

```
if (!require("pacman")) install.packages("pacman")
```

```
## Loading required package: pacman
```

```
## Warning: package 'pacman' was built under R version 4.1.2
```

```
pacman::p_load(twitteR, wordcloud, tm, tidyr, tidytext, syuzhet, ngram, NLP, RColorBrewer, RTextTools, c
```

```
library(shiny)
```

```
## Warning: package 'shiny' was built under R version 4.1.2
```

```
library(here)
```

```
## Warning: package 'here' was built under R version 4.1.2
```

```
## here() starts at E:/CODING/R/project_akhir_prakDS
```

```
library(vroom)
```

```
library(dplyr)
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:twitter':
```

```
##
```

```
##      id, location
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
##      filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(plotly)
```

```
## Warning: package 'plotly' was built under R version 4.1.2
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##   last_plot
```

```
## The following object is masked from 'package:stats':
##
##   filter
```

```
## The following object is masked from 'package:graphics':
##
##   layout
```

```
library(syuzhet)
library(twitterR)
library(ROAuth)
```

```
## Warning: package 'ROAuth' was built under R version 4.1.2
```

```
library(tm)
library(rtweet)
```

```
## Warning: package 'rtweet' was built under R version 4.1.2
```

```
##
## Attaching package: 'rtweet'
```

```
## The following object is masked from 'package:syuzhet':
##
##   get_tokens
```

```
## The following object is masked from 'package:twitterR':
##
##   lookup_statuses
```

```
library(wordcloud)

twitter<- vroom(here("data_bersih.csv"))
```

```
## New names:
## * ' ' -> ...1
```

```
## Registered S3 method overwritten by 'cli':
##   method      from
##   print.tree tree
## Rows: 1500 Columns: 2

## -- Column specification -----
## Delimiter: ","
## chr (1): text
## dbl (1): ...1

##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
tweet<- twitter$text
ui <- fluidPage(
  titlePanel("SENTIMENT ANALISIS VAKSIN SINOVAC"),
  mainPanel(

    tabsetPanel(type = "tabs",
      tabPanel("Emotion", plotOutput("emotion")),
      # Plot
      tabPanel("Data dalam bahasa indonesia", DT::dataTableOutput('tbl')),
      # Output Data Dalam Tabel
      tabPanel("Wordcloud", plotOutput("Wordcloud")),
      tabPanel("Frekuensi", plotOutput("frekuensi"))
    )
  )
)

# SERVER
server <- function(input, output) {

  ####Output Data
  output$tbl = DT::renderDataTable({
    DT::datatable(twitter, options = list(lengthChange = FALSE))
  })

  ####Output Emotion
  output$emotion <- renderPlot({sentiment_dataset<-read.csv("data_bersih.csv",stringsAsFactors = FALSE)

  review <-as.character(sentiment_dataset$text)

  get_nrc_sentiment('happy')
  get_nrc_sentiment('excitement')
  s<-get_nrc_sentiment(review)

  review_combine<-cbind(sentiment_dataset$text,s)
  par(mar=rep(3,4))
  barplot(colSums(s),col=rainbow(10),ylab='count',main='Sentiment Score For vaksin sinovac')
}, height=400)

  ####Output Wordcloud
```

```

output$Wordcloud <- renderPlot({yogyakarta<-readRDS('data_mentah.rds')
jogja=twListToDF(yogyakarta)

#cleaning data
#hanya ambil tweet saja
komen<-jogja$text
komenc<-Corpus(VectorSource(komen))

#hapus tanda baca, link url, huruf aneh, dan emoji
removeURL <-function(x) gsub("http^[[:space:]]*", "", x)
twitclean <-tm_map(komenc,removeURL)

removeRT<-function(y) gsub("RT", "", y)
twitclean<-tm_map(twitclean,removeRT)

removeUN<-function(z) gsub("@\\w+", "", z)
twitclean<-tm_map(twitclean,removeUN)

remove.all <- function(xy) gsub("[^[:alpha:][:space:]]*", "", xy)
twitclean <- tm_map(twitclean,remove.all)

twitclean<-tm_map(twitclean, removePunctuation)
twitclean<-tm_map(twitclean, tolower)

removeprambanan<-function(x) gsub("vaksin sinovac", "", x)
jogja_new<-tm_map(twitclean,removeprambanan)

#membuat nilai untuk masing-masing kata
{
  dtm<-TermDocumentMatrix(jogja_new)
  m<-as.matrix(dtm)
  v<-sort(rowSums(m),decreasing = TRUE)
  jogja_new<-data.frame(word=names(v),freq=v)
}
head(jogja_new,n=10)

set.seed(1234) # for reproducibility
wordcloud(words = jogja_new$word,
  freq = jogja_new$freq,
  min.freq = 1,
  max.words =200,
  random.order=FALSE,
  rot.per=0.35,
  colors=brewer.pal(8, "Dark2"))

})

####Output frekuensi
output$frekuensi <- renderPlot({yogyakarta<-readRDS('data_mentah.rds')
jogja=twListToDF(yogyakarta)

##visualisasi time series
ts_plot(jogja, "3 hour") +

```

```

ggplot2::theme_minimal() +
ggplot2::theme(plot.title = ggplot2::element_text(face = "bold")) +
ggplot2::labs(
  x = NULL, y = NULL,
  title = "Frekuensi tweet tentang vaksin sinovac selama 9 hari kebelakang",
  subtitle = "Twitter status (tweet) counts aggregated using three-hour intervals",
  caption = "\nSource: Data collected from Twitter's REST API via rtweet"
)
}, height=400)
}
shinyApp(ui = ui, server = server)

```