First I will load the required packages.

```
library(tidyverse)
library(gutenbergr)
library(tidytext)
```

Since Pride and Prejudice is currently the most downloaded book on gutenberg over the past 30 days and I happen to like it myself, I'll use that as my example book. It is easy to download the full text with the `gutenbergr` package.

```
book_text <- gutenberg_download(1342)

data("stop_words") # load stop words

book_text %>%
  unnest_tokens(word, text) %>% # turn the text into a single column of
words
  mutate(word = str_extract(string = word, pattern = "[[:alpha:]]+"))
%>% # remove any non-alphanumeric characters
  select(word) %>% # get rid of the extra columns
  unique() %>% # get rid of duplicate words
  anti_join(stop_words, by = "word") %>% # get rid of boring "stop"
words
  drop_na() %>% # drop anything that didn't make it through cleanly
  unlist() %>% # turn the column into a vector that sample() knows what
to do with
  sample(4) # chose four words at random
##        word273       word4600        word632       word4122
##       "choose"      "content" "circumstance"      "mingled"
```

Well that was super easy, wasn't it? The only thing that isn't easy with this setup is looking up a book to use. But that can readily be solved with a shiny app. You can see the code below (at the time this post was knit—the current code will always be on github here), and the live app is at https://jameson-marriott.shinyapps.io/Password_App/.

```
library(shiny)
library(shinythemes)
library(gutenbergr)
library(dplyr)
library(tidyr)
library(tidytext)
library(stringr)
library(rclipboard)

# get all the titles for the drop-down menu
titles <- gutenberg_works(only_text = TRUE, distinct = TRUE) %>%
    select(title) %>%
    drop_na()

# load the stop words so that we don't have to reload it later
data("stop_words")
```

```
ui <- fluidPage(theme = shinytheme("cerulean"),

                rclipboardSetup(), # what it sounds like

    verticalLayout(
        fluidRow(
            column(width = 8, offset = 1,
                   titlePanel(title = "XKCD-Inspired, Gutenberg-Sourced
Passwords"),
                   p("This web-app lets you generate passwords inspired
by ",
                      a(href = "https://xkcd.com/936/", "this xkcd comic."),
                      br(),
                      "First select a book from ",
                      a(href = "https://www.gutenberg.org/", "Project
Gutenberg"),
                      " and then chose the number of words you want to
use from that book for your password.")
                   ),
        ),
        fluidRow(
            column(width = 6, offset = 1,
                   selectizeInput(inputId = "book_title",
                                  label = "Book Title",
                                  choices = c("Chose one" = "",
titles), # removes the default selection, but needs error handling for
the down-stream items
                                  selected = NULL),
                                  #choices = titles,
                                  #selected = "Pride and Prejudice"),
                   p(textOutput("book_length")),
                   sliderInput("number_of_words",
                               "Number of words to chose",
                               min = 1,
                               max = 10,
                               value = 4))
        ),

        # Show the password
        fluidRow(
            column(width = 6, offset = 1,
                   tags$hr(),
                   textOutput("password", container = tags$strong)
            ),
        ),
        # Show the password without spaces
        fluidRow(
            column(width = 6, offset = 1,
                   uiOutput("password_no_spaces"))
        )
    )
```

```
)

server <- function(input, output) {

    # get the book
    gutenberg_book <- reactive({
        validate(
            need(input$book_title != "", "Please chose a book.")
        )
        gutenberg_works(title == input$book_title) %>% # get the
gutenberg id
            gutenberg_download() %>%
            unnest_tokens(word, text) %>% # turn the text into a single
column of words
            mutate(word = str_extract(string = word, pattern =
"[[:alpha:]]+")) %>% # remove any non-alphanumeric characters.
            select(word) %>% # get rid of the extra columns
            unique() %>% # get rid of duplicate words
            anti_join(stop_words, by = "word") %>% # get rid of boring,
"stop" words
            drop_na() %>% # drop anything that didn't make it through
cleanly
            unlist()
    })

    # Report the number of unique words in the book
    output$book_length <- renderText({
        length <- gutenberg_book() %>%
            length() %>%
            format(big.mark = ",") # Add some nice formating

        paste0("There are ", length, " unique words in this book
(including diffent forms of the same word).")
    })

    # Generate the actual password from the book
    password <- reactive({
        validate(
            need(input$book_title != "", "")
        )
        gutenberg_book() %>%
            sample(input$number_of_words) %>% # chose words at random
            paste0() # drop the names
    })

    # Output the password for the UI
    output$password <- renderText({
        password()
    })

    # Make the button to copy the password to the clipboard
    output$password_no_spaces <- renderUI({
```

```
        rclipButton("clip_button", paste0("Copy \"",
str_flatten(password()), "\""), str_flatten(password()))
    })
}

# Run the application
shinyApp(ui = ui, server = server)
```