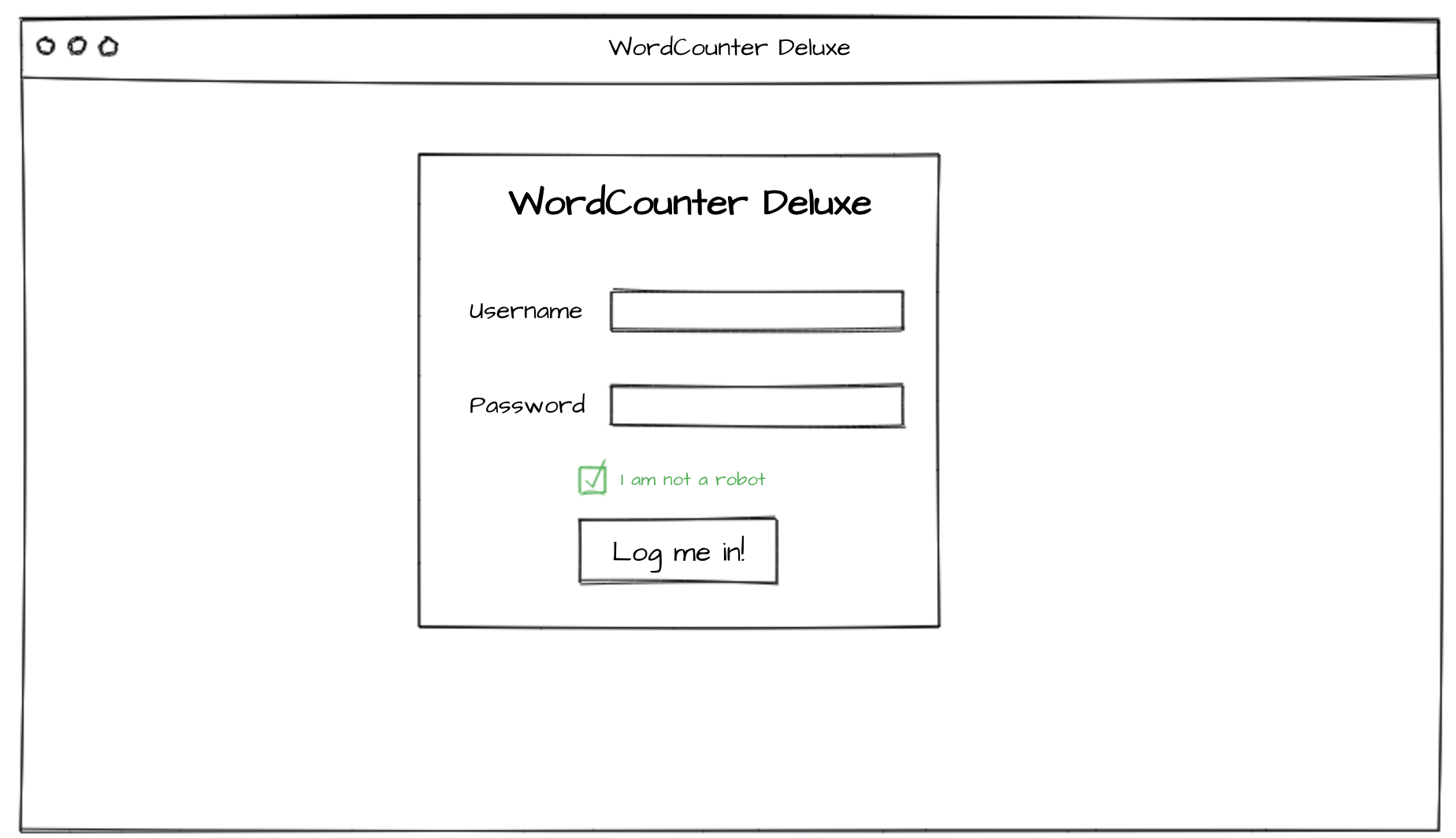
Hi Yuval.

Congrats on passing the first step in the Hyver interviewing processes. As we discussed, the second phase in this process is the programing exrecise this document covers. The purpose of this exrecise is to assess you programing, design, and learning abilities.

In this excresize, we will create a simple website called “WordCounter Deluxe” which enables users to log in and count the number of occurences of their favorite words in their favorites websites!

The website flow is as follows:

1. Login page:



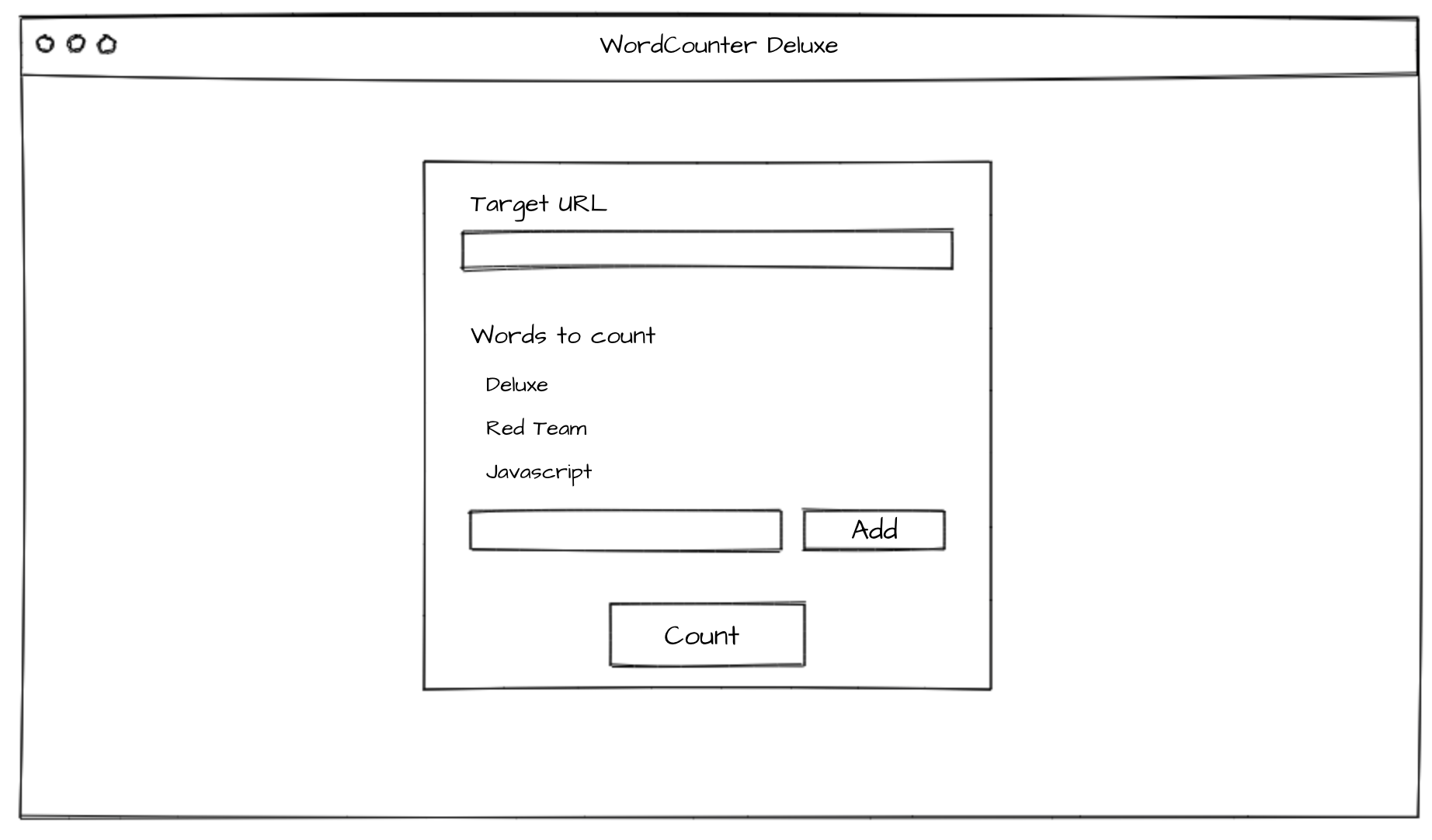
The login page will contain the following elements:

* 1. A title with the name of the website
  2. An input for inserting a username
  3. An input for inserting a password
  4. A reCaptcha I’m not a robot widget
  5. A login button

When the user will click on the login button, a login should take place. The login should fail if the username or password are wrong, or if the user did not pass the captch challenge. If the login fails, an error message should be displayed.

If the login succsudes, the user will continue to:

1. Start count page:



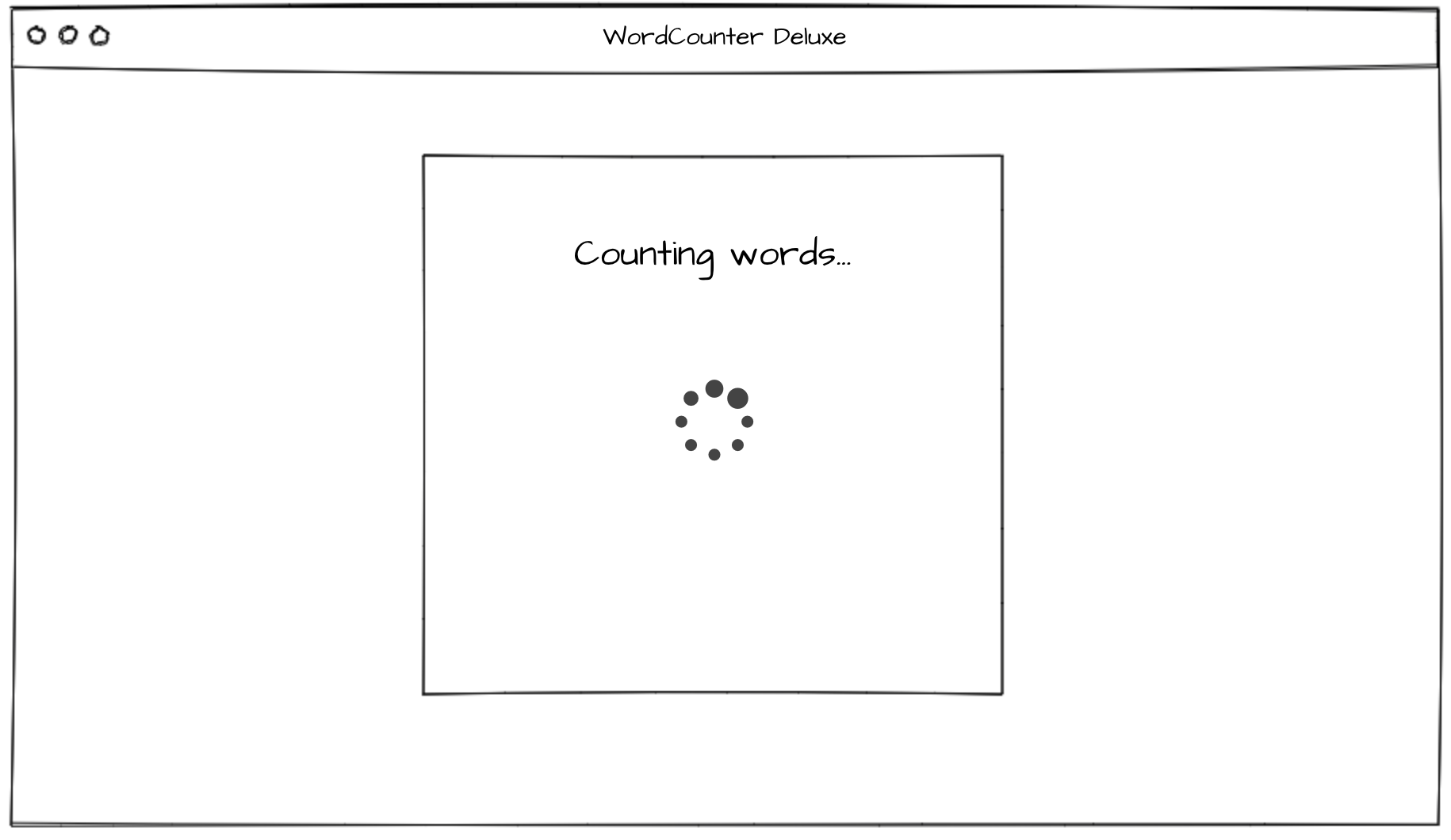
The start count page will contain the following elements:

* 1. An input (and label) in which the target URL is to be entered (Example: [www.hyver-security.com)](http://www.hyver-security.com))
  2. A list of the words already added to the count (In the example above the user has already added the words “deluxe”, “Red team” and “JavaScript”
  3. An input and an add button to add new words to the list. When a word is added it should immediately appear in list b
  4. A count button.

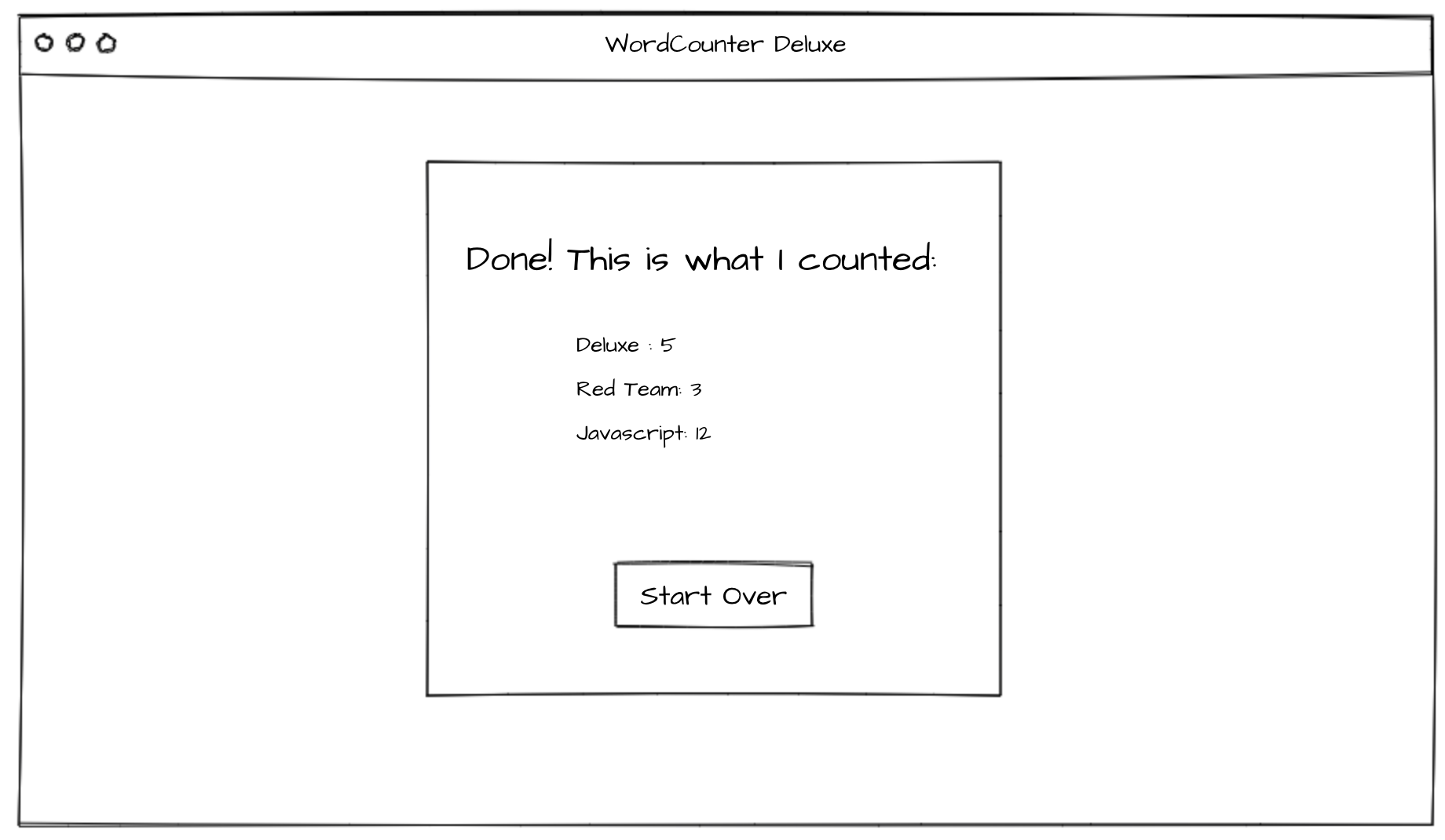
When the count button is clicked, the following should take place:

1. A request will be sent to the server
2. The server will fetch the content of the target url
3. The server will count the number of occurrences of the words added and return the number of times each of the words appeared in the target URL.
4. You can assume that the target URL does not require authentication.

While the server is counting the words, a loading spinner should be shown:



After the count is done, the results should be displayed as follows, with a button that will return the user to screen 2.



Instructions

1. Implement, using your choice of a programming language and web server the web site described above.
2. Only authenticated users should be able to reach the post login pages and invoke the counting activity.
3. Implement the captcha mechanism using Google’s reCaptcha API: https://www.google.com/recaptcha/intro/
4. For each user, store the search history in a persistence storage solution of your choice. (DB, Files, whatever you find best suiting)
5. It is preferable, but not essential, for the assignment to run on Linux.
6. Nice visual design for the website is not a must, but its defiantly a bonus. However, please attempt to follow the layout provided.
7. Provide a “hyber\_ex.tar” file which will contain the following:
   1. All the files needed to run the webserver
   2. A script which will output all the words that were ever searched by any user on any website, and how many times they were searched.
   3. Clear installation instructions

Good luck!

The Hyver dev team.