Boston University
Electrical & Computer Engineering
EC463 Capstone Senior Design Project

First Prototype Testing Plan

Crowdsourcing platform for rating and improving research code: Reproducibility, Reusability, Readability (RE3)

Submitted to

Ana Trisovic
anatrisovic@g.harvard.edu

by

Team #5
RE3

Team Members

Andreas Francisco De Melo Oliveira andoliv@bu.edu
Ethan Hung ehung@bu.edu
Jyotsna Penumaka jyotsnap@bu.edu
Layan Bahaidarah layanb@bu.edu
Lukas Rosario lukasr@bu.edu

Summary of Equipment and Setup

Hardware:

- Computer/Laptop

Software:

- Front End
    - React JS web application
    - Tailwind CSS
    - Firebase JavaScript SDK
- Backend:
    - Google Firebase
        - Firestore
        - Storage
        - Authentication
    - Python scripts to:
        - Upload all snippets to google storage
        - Read in user snippet ratings into json

Set Up:

Since our project is just software based all of the set up required is cloning the git repository with the following command:

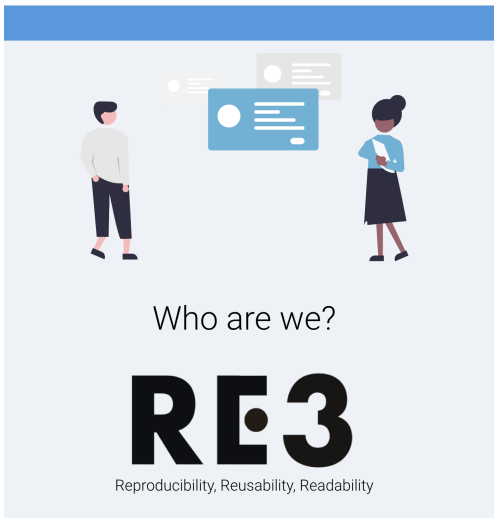
'git clone https://github.com/BostonUniversitySeniorDesign/21-05-Re3.git'

Following that command simply go to the re3-client folder inside the cloned repository, and run 'npm install', followed by 'npm start' in the terminal (make sure you have Node.js installed). The browser should open the website for our application on localhost:3000 and we can start testing it. Also, a python script is run beforehand to populate the storage with code snippets that will be displayed on the web page for users to evaluate (in terms of readability).

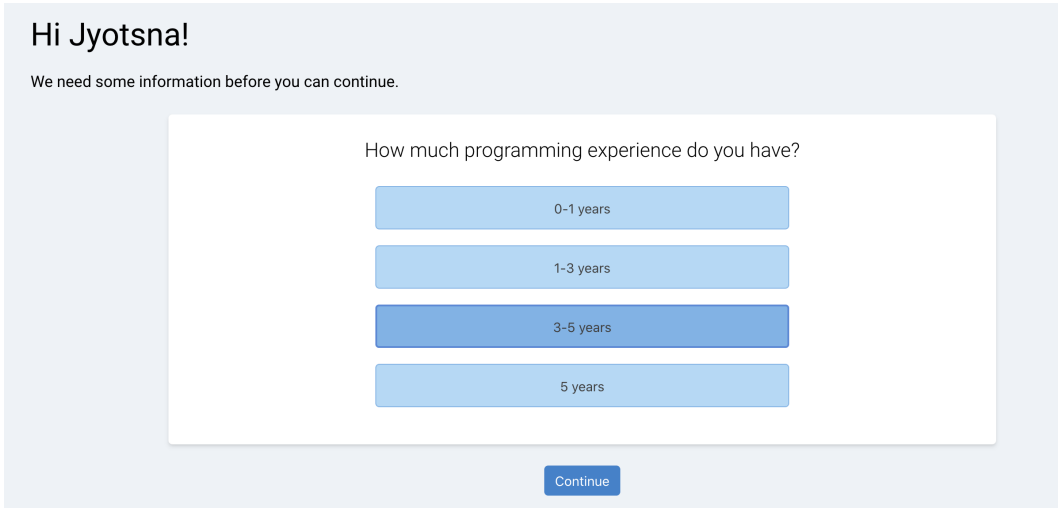
Detailed Measurements Taken:

The criteria for successful running and output is as follows:

1. User was directed to the *Home Page* when the project was opened in the web browser

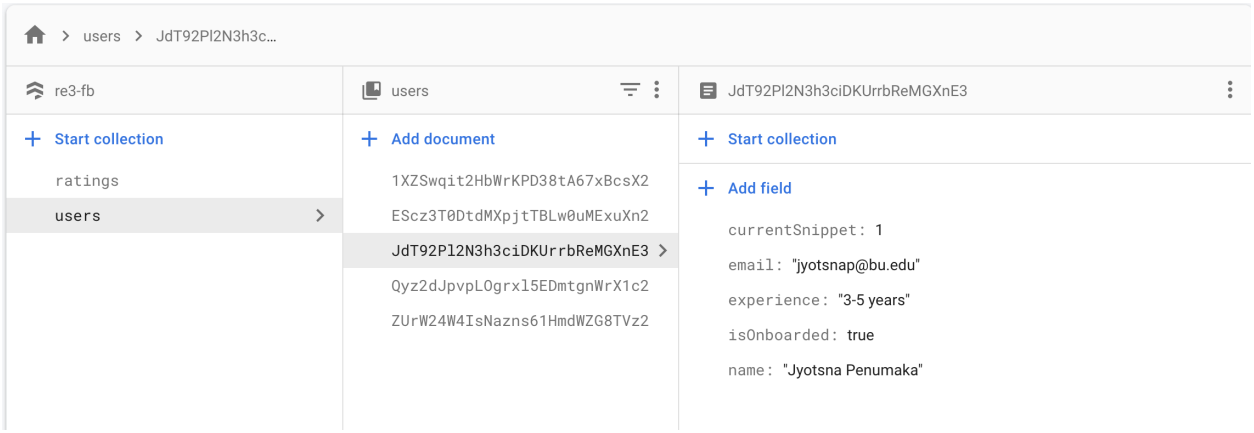


2. On clicking the *Login Button* the user was directed to the login page, since this was the first time the user logged in, he was redirected to the *Onboarding Page.*

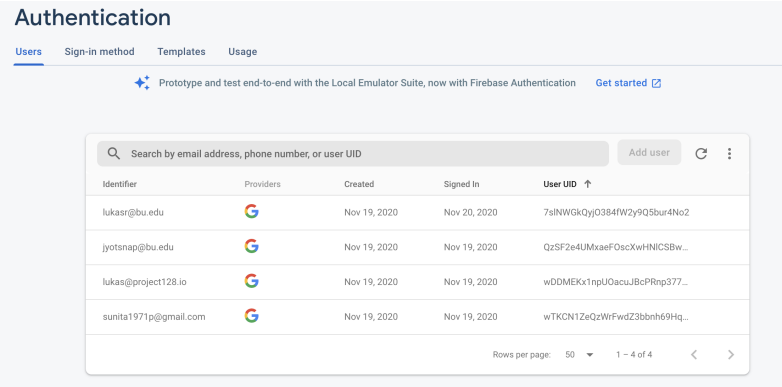


   a. On clicking the *continue button* on the *Onboarding Page* the following information was stored in the firestore under:

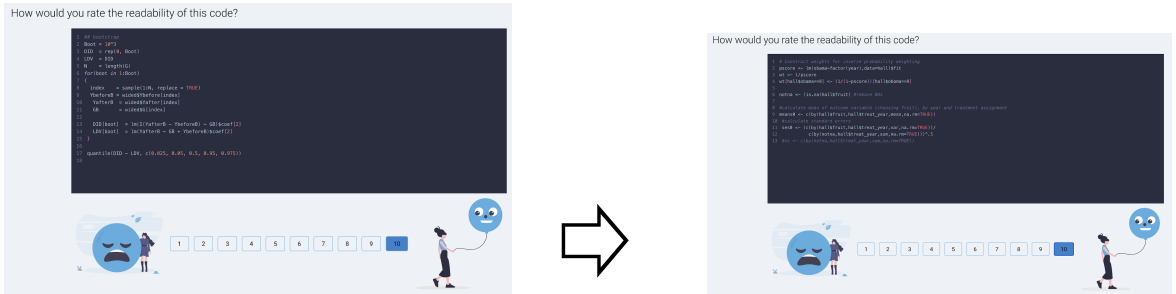
users → uid → { currentSnippet, email, isOnboarded, experience, name }



3. If the user logins again after signing out, he was redirected to the *Rating Page* instead of the *Onboarding Page*
4. User was registered in google firebase authentication (uid, name and email)

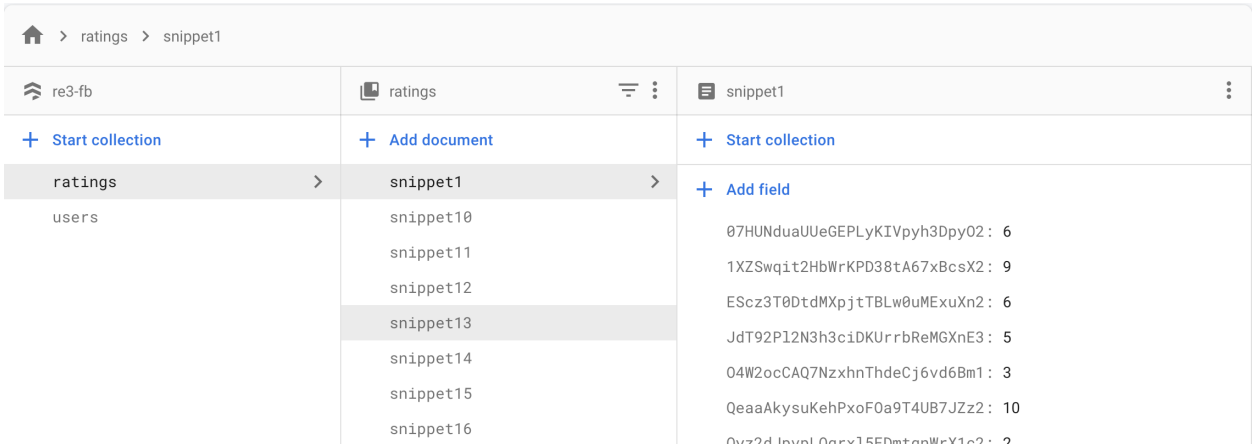


5. The text box displaying the code snippet should transition to the next snippet when the user clicks on a rating

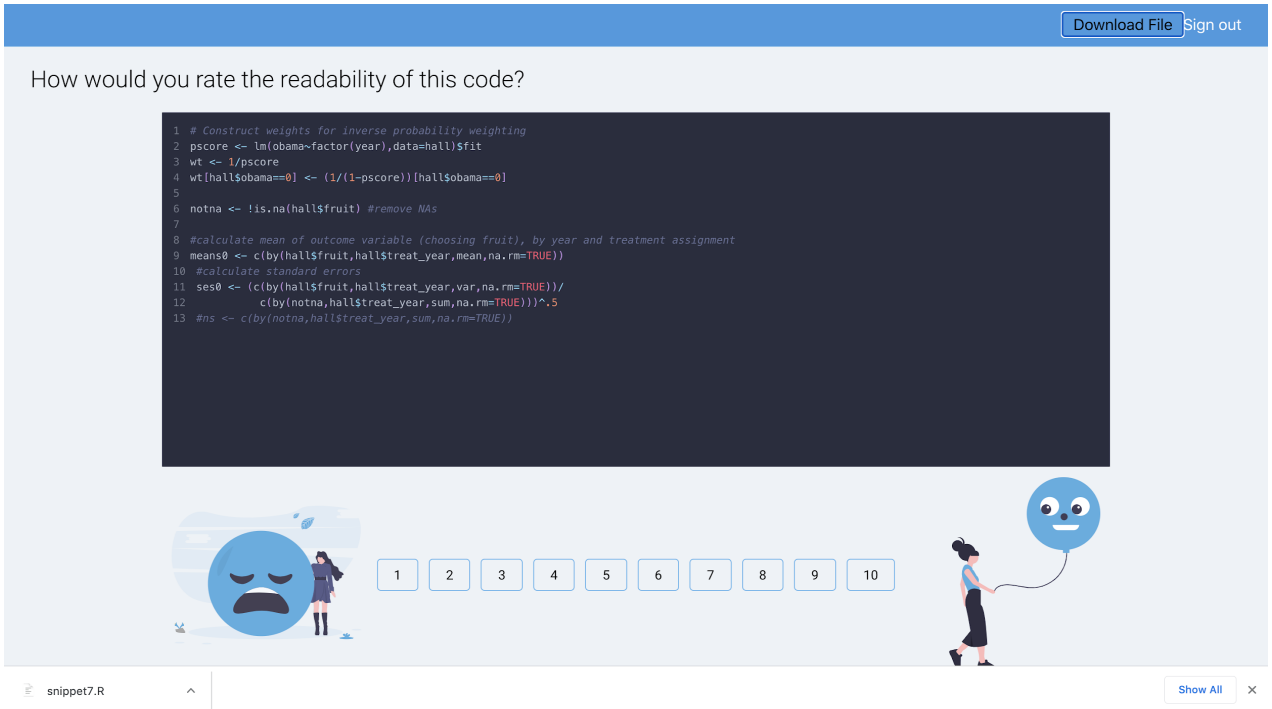


6. Once the user clicked on a rating (1 - 10) the rating for the snippet was updated on firestore under:

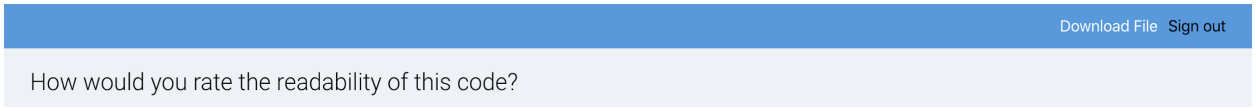
   ratings → snippet# → user_uid : rating

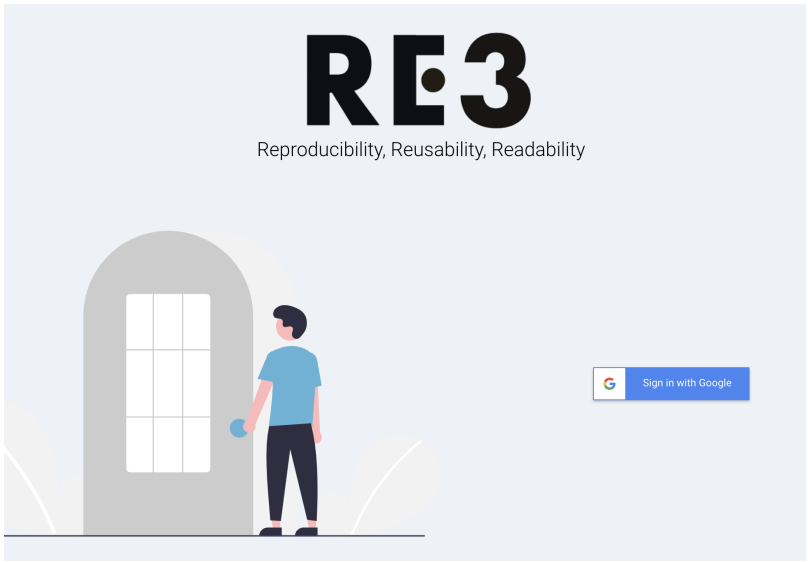


7. The *Download File button* should download the current snippet on the user's computer.

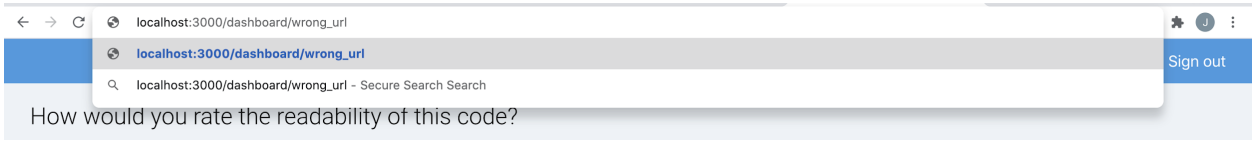


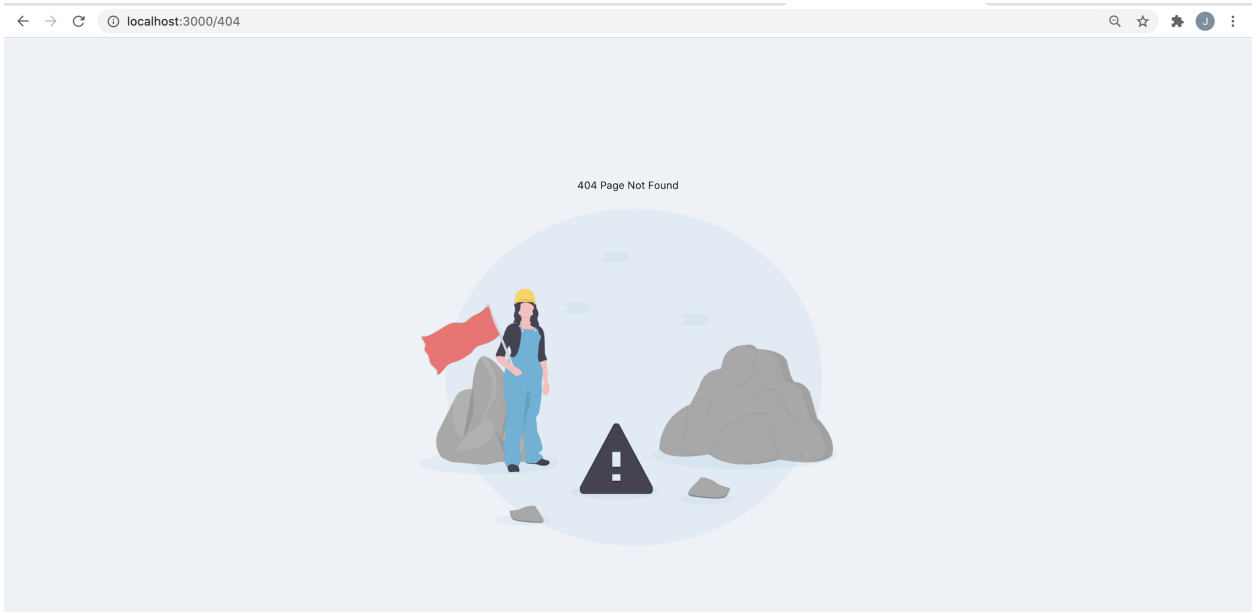
8. When the user clicks on the *Sign out button* they should be redirected to the *Login Page*





9. If the user enters an invalid url, they should be directed to the *NotFound Page*.





Conclusions:

Overall, prototype testing went well. The group demonstrated our current progress to Professor Osama and received helpful feedback for the future of our project. Following prototype testing, a few conclusions were made by the group. Firstly, a few aspects could be added to our rating platform that would make the webpage more user friendly. Another conclusion was that before the website goes online for users to rate snippets, the group had to look into Firebase security rules to prevent any malicious behavior. Also, since our React web application has been running locally on our machines for this testing period, the group has to look into finding a public domain for hosting our web application. In terms of bugs, an issue that has to be combated by the group is a bug when logging into 2 separate google accounts. Lastly, the group needs to look into how to administer this code readability survey to a 100 or more students. These future changes are listed below in bullet form.

Future Changes:

- Add UI improvements to rating system:
    - Transition pages to redirect the user to a different page after he has finished rating all snippets.
    - A snippet counter allowing the user to track his progress out of 100.
    - A previous button allows the user to go back and change his rating after submitting.
- Change to more secure security rules for Google firestore and storage.
    - Prevent unauthorised users from uploading files to storage.
    - Only allow authenticated users to add snippet ratings in firestore.

- Decide on website domain so we can start deploying the survey
- A bug was found that if a user was logged into two google accounts then the current snippet the user was supposed to rate was not updated correctly
- Look into administering survey to 100 or more people
    - ECE students
    - CS students
    - Dataverse workers