Software Implementation and Testing Document

For

Group 5

Version 1.0

Authors:

Trevor Fagan

Christopher Que

Dori Palmer

Miguel Sarenas

Wilfredo Huertas

1. Programming Languages (5 points)

For this project we are currently using the MERN stack, so a lot of our project is done in Javascript and Typescript. We are using HTML5 and CSS3 on the frontend. In the backend we are just using Javascript. To query our database we are using SQL.

2. Platforms, APIs, Databases, and other technologies used (5 points)

When it comes time to deploy, we are going to use vercel for the project. Currently we are using the Auth0 for user authentication and login through Google, the Plaid API for connecting to user's bank accounts, Bootstrap for styling, and React for the overall site. Auth0 is a great because it saved us from having to completely code out user authentication on our own and allows Google login which is convenient. We used the Plaid API because it is one of the most popular banking APIs and allowed us to efficiently and conveniently connect to dummy data (and hopefully real data once we get confirmation). Tailwind CSS is great because it allows us to easily make the site responsive and have the styling options we like. Finally, we used React because the optimization for SEO is great and it is a great modern framework that we should all have experience in using.

3. Execution-based Functional Testing (10 points) *Describe how/if you performed functional testing for your project (i.e., tested for the functional requirements listed in your RD).*

We performed functional testing by going through our list and testing the features on our website. To test the buttons on the sidebar, we clicked through all the buttons and made sure it directed to the right page, which they do. We also tested to make sure that the link to set up gmail worked correctly. We tested the boxes on the dashboard by clicking them to also make sure they direct the user to the correct page. For example, the transaction income box when clicked will redirect the user to the income page. The log out button was tested by clicking on it and making sure the user is logged out completely. We tested the connect your bank by making sure it directs you to the appropriate spot to connect your bank account, the actual bank account link isn't functional yet.

4. Execution-based Non-Functional Testing (10 points) *Describe how/if you performed non-functional testing for your project (i.e., tested for the nonfunctional requirements listed in your RD.*

For non-functional testing, we mainly had to make sure that our APIs performed properly with correct time constraints. We verified that it took less than 5 seconds to connect to the Auth0 API and return the data to the screen. Similarly, we did the same thing for the Plaid API and made sure that we got a response from the API so that we can fetch from our database quickly.

5. Non-Execution-based Testing (10 points) *Describe how/if you performed non-execution-based testing (such as code reviews/inspections/walkthroughs).*

At this point, we haven't really done much non-execution-based testing simply because we work on the project as a group. We code together and in that sense we perform this kind of testing in real time between each other.