Software Implementation and Testing Document

For

Group 8

Version 1.1

Authors:

Nicholas Pena
Yelena Trunina

1. Programming Languages (5 points)

Python will be used to handle the backend functionality.
Python has a clean syntax and a large library of modules and packages.

HTML/CSS/JavaScript will be used for the front-end design and layout
Provide a way to create visually appealing and responsive web pages; easy to learn.

SQL will be used for the database design.
Highly optimized and provides a way to store data in a structured way.

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

Flask will be used to handle the backend functionality.
Jinja will be used for template rendering.
MySQL will be used as the database management system.

3. Execution-based Functional Testing (10 points)

Home Page:

- *Verify that the description of the web app is displayed on the page.*

- *Click the "Sign up" button and ensure that it takes you to the sign-up page.*
- *Click the "Sign in" button and ensure that it takes you to the login page.*

Login Page:

- *Enter a valid username and password and click the "Login" button. Ensure that it logs you into the application.*
- *Enter an invalid username or password and click the "Login In" button. Ensure that it displays an error message.*

Sign Up Page:

- *Enter valid information in all required fields and click the "Sign up" button. Ensure that it creates a new user account.*
- *Enter invalid information in any of the required fields and click the "Sign up" button. Ensure that it displays an error message.*

Dashboard Page:

- *Click the "Create budget" button and ensure that it takes you to the budget creation page.*
- *Select a budget type, enter a budget name, start date, and end date, and click the "Create" button.*
- *Ensure that it creates a new budget and redirects you to the dashboard page.*
- *Click on the newly created budget and ensure that it takes you to the budget page.*

Budget Page:

- *Add a new transaction and ensure that it is successfully added to the budget.*
- *Edit an existing transaction and ensure that the changes are reflected in the budget.*
- *Delete an existing transaction and ensure that it is successfully removed from the budget.*

4. Execution-based Non-Functional Testing (10 points)

For each page,

- *Verify that the sign-up process is secure, and that user data is encrypted and stored safely.*
- *Verify that the page loads within a reasonable time frame.*
- *Verify that pages are responsive.*

5. Non-Execution-based Testing (10 points)

For all the pages,

- *Review the content on the page to ensure that it is clear, concise, and provides users with a good understanding of the purpose and functionality of the page.*
- *Review the user interface design to ensure that it is consistent with industry standards and user expectations.*
- *Review the code for any security vulnerabilities or potential performance issues.*