Software Requirements and Design Document

For

Group <8>

Version 1.0

Authors:

AJ Tello
Alaina Foo
Evgeniya (Ginny) Kalashnikova
Nicholas Pena
Yelena Trunina

1. Overview (5 points)

Creating and adhering to a budget while traveling is essential for a successful trip; however, it can be a daunting task to plan, manage expenses, and divide costs amongst multiple people, especially when traveling in a large group. Our proposed web application aims to make this process easier by providing a collaborative, user-friendly platform for creating and managing a travel budget, tracking transactions, and analyzing spending.

Features:

- *Budget creation: users will be able to create a budget for their trip by specifying the total amount they plan to spend and allocating funds to different categories (e.g. transportation, food, activities).*
- *Transaction tracking: users will be able to log their transactions (e.g. Airbnb, gas, flights) and see how they are impacting their budget in real time.*
- *Spending analysis: the application will provide users with reports and charts that show how they are spending their money, helping them identify areas where they may be overspending.*

2. Functional Requirements (10 points)

Home Page (Medium Priority, Medium size)
- *The home page should provide a clear description of what the web application is about, including its purpose and functionality.*
- *The home page should provide easy access to the 'sign-up' and 'sign-in' pages through clearly labeled buttons.*

Login Page (High Priority, Small size)
- *User should be able to click 'Login' button after entering Username and Password and be taken to Dashboard page.*
- *The login page should provide clear error messages if the user enters incorrect login information.*

Sign Up Page (Medium Priority, Medium size)
- *The sign-up page should allow users to enter their personal information and create a new account.*
- *The sign-up page should provide clear error messages if the user enters incomplete information.*

Dashboard Page (Medium Priority, Extra-Large size)
- *The dashboard page should allow users to view and manage their budgets.*
- *The dashboard allow users to create a new budget by entering budget information, such as budget type, budget name, start date, and end date.*

Budget Page (Low Priority, Extra-Large size)
- *The budget page should allow users to add, edit, and delete transactions associated with a specific budget.*
- *The budget page should display key transaction information, such as date, amount, and description.*
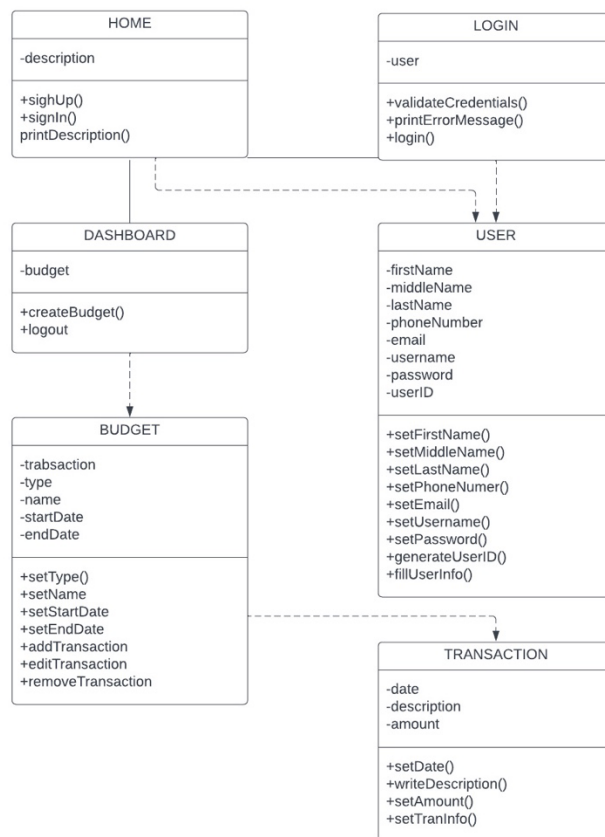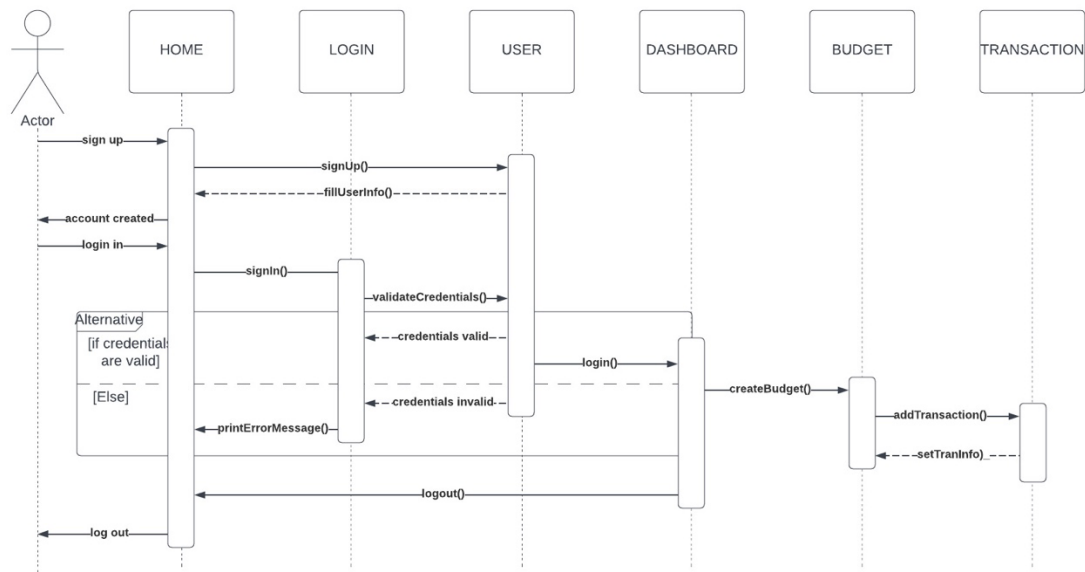
3. Non-functional Requirements (10 points)

- *Data is stored in the database.*
- *User data is encrypted and stored safely.*
- *Pages loads within a reasonable time frame.*
- *Pages are responsive.*

4. Use Case Diagram (10 points)
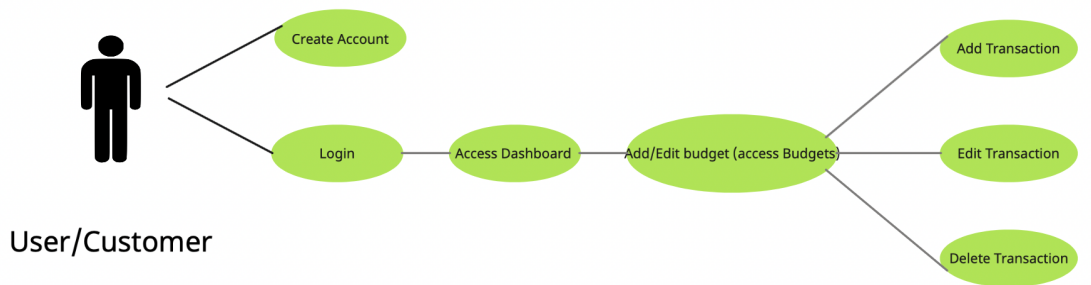
5. Class Diagram and/or Sequence Diagrams (15 points)

6. Operating Environment (5 points)

The software can operate effectively as long as the user has a modern web browser that is up-to-date, is running on an operating system that is updated to a reasonably new version, and has a reliable internet connection.

7. Assumptions and Dependencies (5 points)

The presupposed factors entail the non-utilization of encryption for the storage of user data and the lack of implementation of security features. Additionally, it is presumed that the user will not engage in data overloading that could potentially result in database corruption or deliberately attempt to crash the application. The project currently has the following dependencies:

- Python: The project requires Python to be installed on the system as it serves as the primary language used for development.

- Jinja: Jinja is a templating engine used for rendering HTML or other markup languages in Python-based web applications. This project requires the Jinja library to be installed as a Python module.

- Flask: Flask is a lightweight web framework used to develop web applications in Python. This project requires Flask to be installed as a Python module.

- MySQL: MySQL is the chosen relational database management system used with our project. The project would require the MySQL connector to be installed as a Python module to interact with the database.