Team Name: CS Enjoyers

Project Title: Budget Buddy

Deliverable: PM 2

Use cases:

- **Case 1: Setting Up a New Budget**
    - **Goal:** User wants to set a new budget for a specific duration and amount
    - **Preconditions:** User must have an account and be logged in. User must also have at least one bank account or card connected to their account
    - **Main Flow:**
        - User selects the option to create a new budget
        - User enters the details of the budget, including category, amount, and duration/time frame
        - The system validates the information and saves the new budget to the account
        - The user receives confirmation from the system
    - **Alternative Flows:**
        - If the user enters and invalid amount (ex. Negative number, number bigger than monthly limit, etc.), the system prompts for the correct input
        - If no bank account or card is connected, the system prompts user to connect an account before budgeting
    - **Post Conditions:** The new budget is active, and tracking begins immediately for the user
- **Case 2: Viewing Budget Status**
    - **Goal**: User wants to view the current status of their budgets
    - **Preconditions:** User has an account and is logged in
    - **Main Flow:**
        - User navigates to the budget overview section
        - The system displays all active budgets with their current status, including spent amount, remaining balance, etc.
        - User selects a budget to view more information about the budget and detailed transactions
    - **Sub Flow:**
        - Users can filter the view by budget category, duration, limit, etc.
    - **Alternative Flows:**

        - If the user has no active budgets, the system will prompt them to start budgeting so that they can view progress
    - **Post Conditions**: User gains detailed insight to their budget statuses
- **Case 3: Receiving Budget Notifications**
    - **Goal**: User receives notifications to stay on track with their budgets
    - **Preconditions**: User has an account and has enabled notifications within their settings
    - **Main Flow**:
        - The system checks each active budget's status and calculates the remaining balance after each purchase.
        - If a budget exceeds X% of its limit, the system sends a notification to the user
        - The user views notification.
    - **Sub Flows**:
        - User can customize notification settings, such as the percentage threshold
        - User can adjust number of notifications they receive
    - **Alternative Flows**:
        - If notifications are disabled, no notifications are sent to the user
    - **Post Conditions**: User is aware of their budget status and can adjust spending accordingly
- **Case 4: Updating a Budget**
    - **Goal**: User wants to modify the details of an existing budget
    - **Preconditions**: User must have an account, be logged in, and have an existing budget
    - **Main Flow**:
        - User selects an existing budget to update
        - User edits the budget’s details such as amount, category, duration.
        - The system validates and saves the updated budget
        - The user receives a confirmation notification and their budget continue
    - **Alternative Flows**:
        - If user tries to set an invalid budget amount, the system prompts them for a valid input
    - **Post Conditions:** The budget is updated with new details and the user can continue spending within their set budget
- **Case 5: Connecting a New Bank Account or Card**
    - **Goal**: User wants to add a new bank account or card to the app

- **Preconditions**: User must have an account and be logged in
- **Main Flow:**
  - User selects the option to add a new bank account or card
  - User provides the necessary authentication details required by their bank and our system
  - The system securely connects with the bank's APO and adds the account or card
  - User receives a confirmation that the account or card is connected
- **Alternative Flows:**
  - If authentication fails, the user is prompted to re-enter details or contact their bank for assistance
- **Post Conditions:** The new account or card is connected and available for the user to budget with

Kanban

Priority List:

1. Establishing a deadline to work towards
   a. A deadline to work towards is crucial to get our product / app out into the real world. Without a deadline our team would have nothing to work towards and would eventually slow down how fast everyone works. With the deadline, we can work towards it and finish quicker.
2. Establishing clear communications between all members/teams.
   a. We need to establish clear communication between all team members and teams because it is essential that our various teams communicate to each other about issues that might affect each team. For example, if the front end is implementing something that would need the backends help like a login system for example they would need to let the backend know how the login system works and what features it has.
3. Meet with backend developers to discuss plans for the backend structure.
   a. This is essential to creating the app since we need to always know what the backend team has been doing/ planned so we try to get all perspectives when it comes to developing the app in terms of the backend
4. Meet with frontend developers to discuss plans for the frontend structure.
   a. This is also essential to creating the app since we need to always know what the front-end team has been doing/ planned so we try to get all perspectives when it comes to developing the app in terms of the front end. After forming

a plan with the front end, we can successfully get a vision of how the app will look.

5. Contact various banks.
    a. Having contacted banks we can establish a list of banks that are onboard with our app and will be able to work around the information our app will need for each user and their budget plan. We need to ensure the major banks are willing to work with us to move forward.
6. Having a working prototype for test candidates to use.
    a. Having a working prototype is essential for being able to collect sufficient data from test subjects. This will be above our test candidate priorities, as having a working prototype is the most important when it comes to collecting feedback and data from test subjects.
7. Find test candidates to link banks with the plan to get sample data.
    a. To collect correct sample data from the test candidates, we need to make sure that the test candidates can link their banks with our app for the prototype to work. This also includes receiving correct information from our test subjects.
8. Meet with backend/frontend to try to incorporate sample data from findings.
    a. Our backend and frontend teams must meet to discuss any developments in their progress. This establishes clear communication and provides transparency to the whole team, which will allow us to be more efficient and successful when working towards our end goal.
9. Update previously written code if necessary
    a. Considering that we are using an adaptive and iterative process, it is important for us to go back and look through our code as we progress forward. We need to be able to adapt our code to any new findings, which will make our app greater and more flexible when developing.
10. Have weekly meetings to discuss progress towards deadlines.
    a. Having clear communication and transparency is extremely important when working as a team, so having weekly meetings that give updates on each team member's progress is essential for having this transparency.

The reason for having deadlines as our top priority is because without establishing a deadline, we have no timetable or timeline to work with. This makes communication a lot more difficult, as well as making planning a lot more difficult. Without having deadlines, working as a group towards an end goal would be almost impossible, as it is especially important to understand what should be done and when.

The reason for having weekly meetings as our least important priority is because while important, if we have clear communication between group members, we can still work towards our end goal. Having weekly meetings is a priority, but the others listed are more essential when it comes to developing our final product.