Project Analyse

Title:

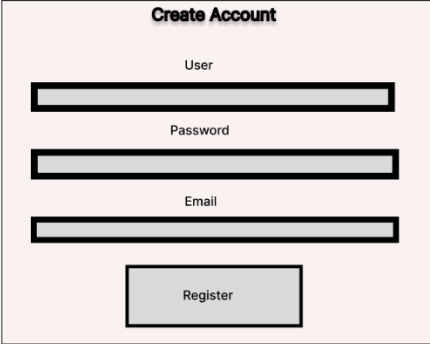BudgetBuddy

Description:

BudgetBuddy is an application that helps users in managing their personal financial budgets more effectively. In this application, users are provided with the option to create different spendings and measure expenses through entering these figures by hand or through entering preexisting bank statements. The application offers insights into the user's spending on a monthly basis and also provides bulges based on previous records. The users are free to place financial controls, and the application gives advice on saving methods when the users are close to the financial goals.

User Story 1:

Create Account:
As a new user, I'm able to create an account so that I can manage my personal finances.
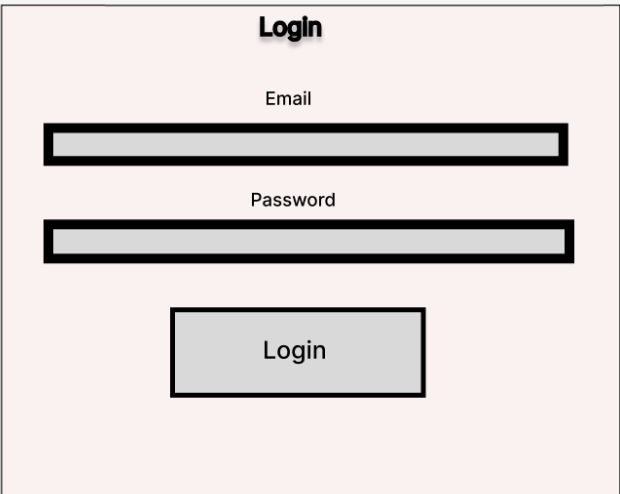
Wireframe:

Acceptance Criteria:
- User can create a new account with unique login credentials.
- An error message is displayed if the email address is already in use.

User Story 2:

Login to Account: As a user, I'm able to log into my account so that I have access to my finances.
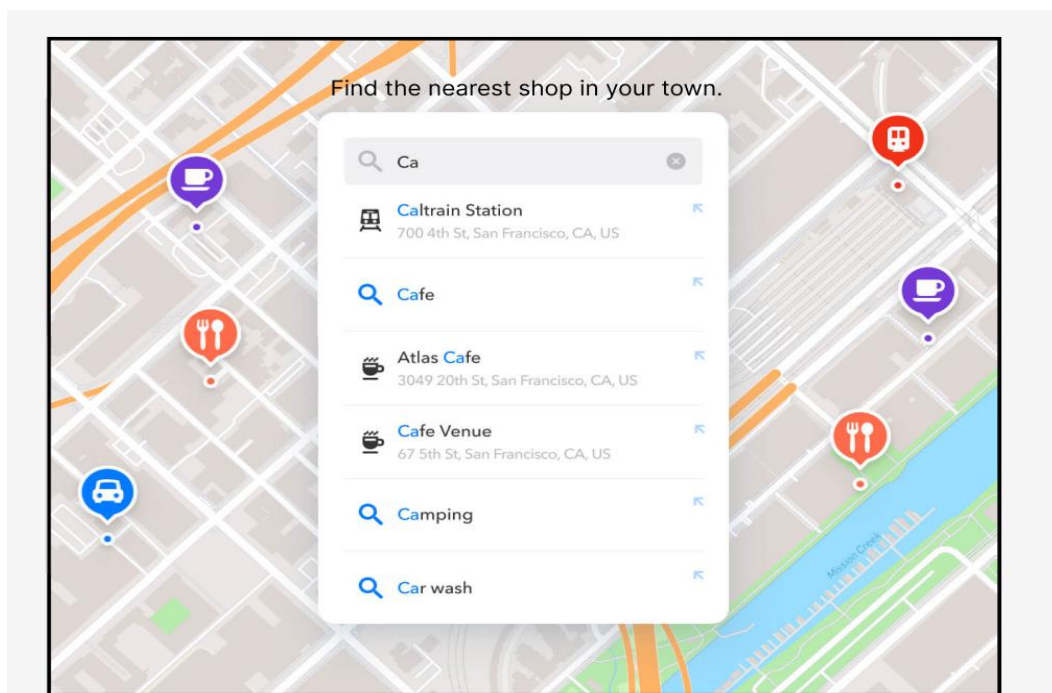
Wireframe:

Acceptance Criteria:

- User can log in with the correct email and password.
- An error message is displayed if the login credentials are invalid.

User Story 3:

View Nearby Shops on Homepage: *As a user, I want to see a list of nearby shops and a map on the homepage so that I can easily find places to shop around me.*

Wireframe:
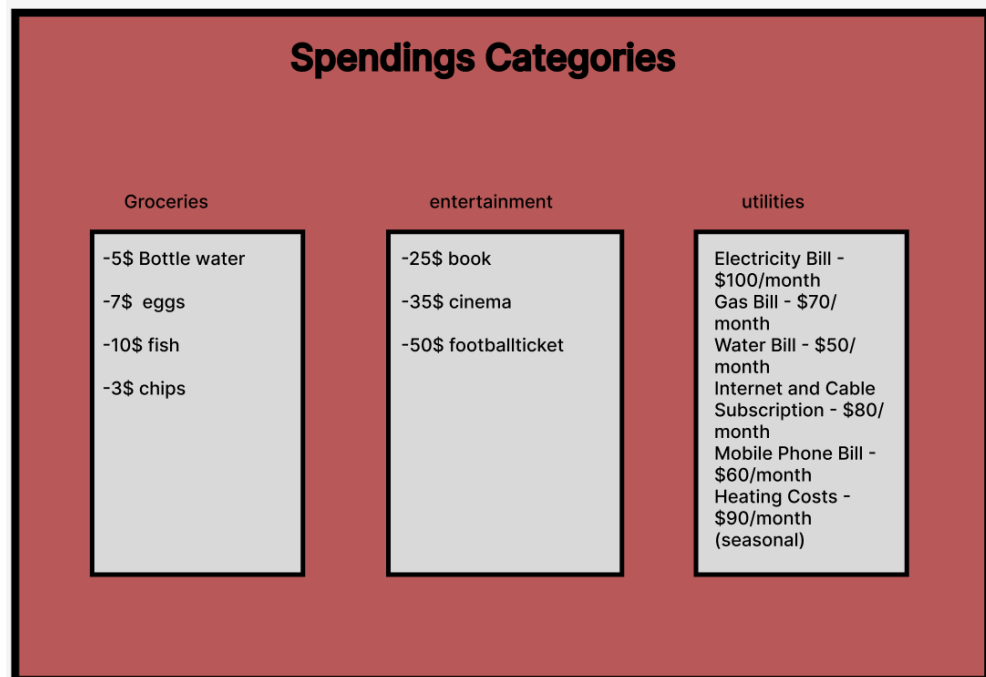

Acceptance Criteria:

- The homepage displays a list of nearby shops, including the shop name, type (e.g., grocery store, clothing store), and distance from the user.

- The homepage includes an interactive map that shows the locations of these shops.

- The user's location is used to show the closest shops.

User Story 4:

Create Spending Categories: As a user, I'm able to create custom spending categories (e.g., groceries, entertainment, utilities) so that I can organize my expenses and track where my money is going.

Wireframe:

Acceptance Criteria:
- User can create, edit, and delete custom categories.
- A confirmation message is displayed after creating or editing a category.
- Each category shows the total spent amount and remaining budget.
- Error message appears if the category name is empty.

User Story 5:

Log Expenses: As a user, I'm able to log my expenses in different categories so that I can keep track of my spending habits.
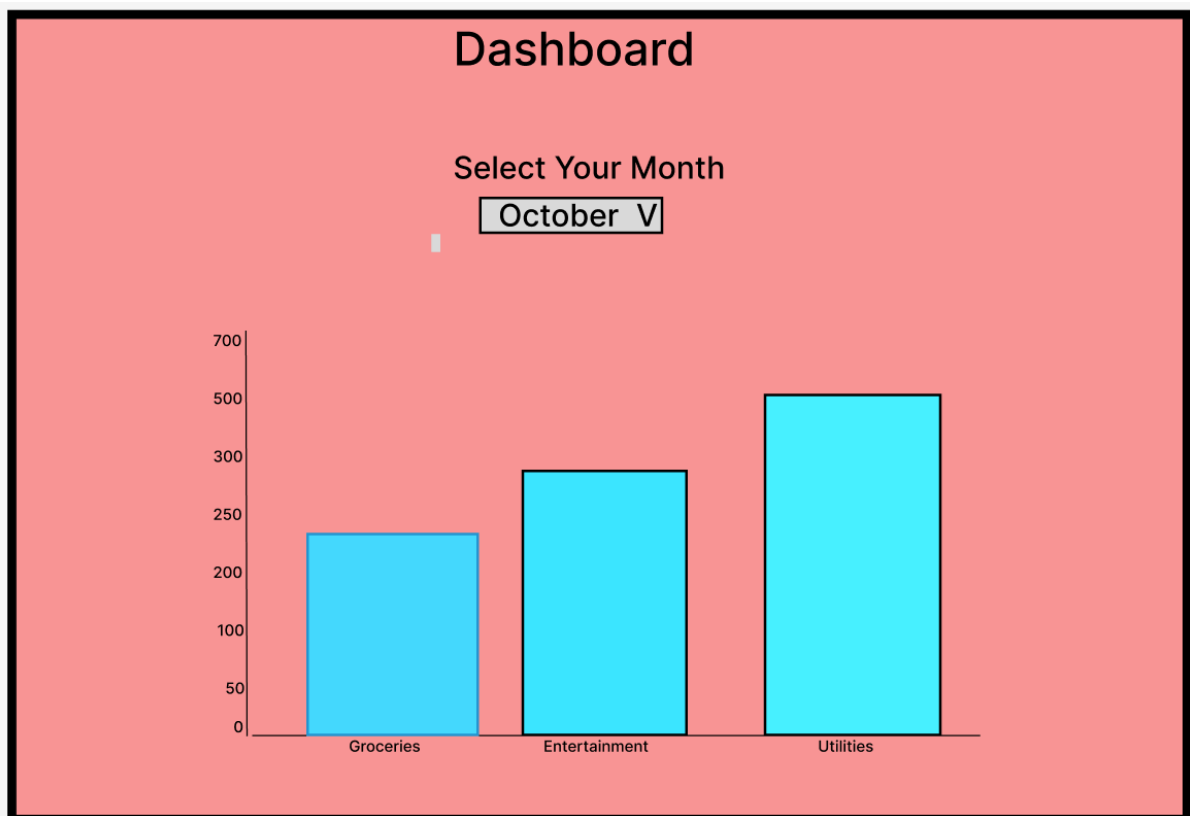
Wireframe:

Acceptance Criteria:
- User can manually log expenses with the amount, date, and description.
- Expense data is reflected immediately in the total spent for that category.
- A success notification after saving an expense.
- Error handling for invalid inputs (e.g., negative numbers, empty fields).

User Story 6:

View Monthly Spending Report: As a user, I'm able to view a monthly report of my spending so that I can analyze my financial behavior and make informed budgeting decisions.
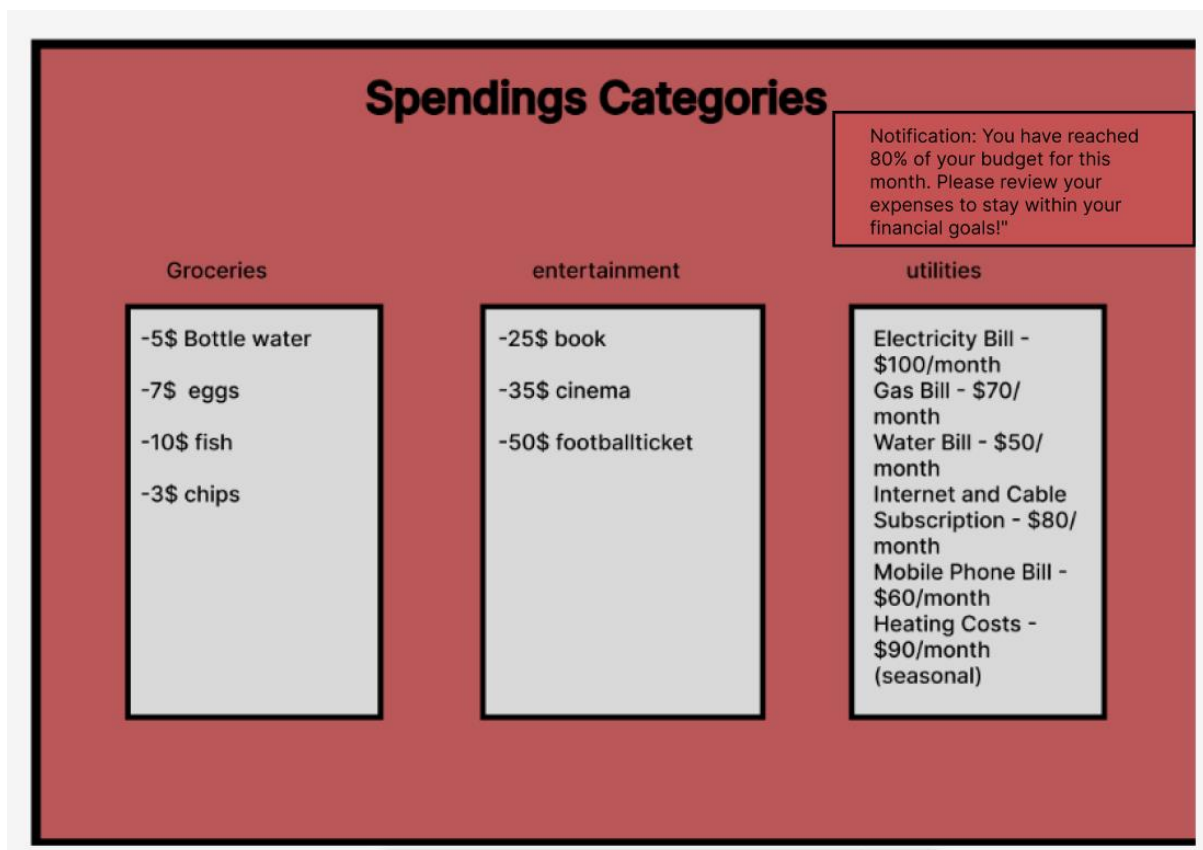
Wireframe:

Acceptance Criteria:

- The app generates a monthly report summarizing total spending and category-wise breakdown.
- Users can view reports for previous months.
- Users receive a notification or alert if they exceed the set budget for any category.
- Report includes visual representation (charts or graphs) and summary text.

User Story 7:

Set Budget Alerts: *As a user, I want to receive real-time alerts when I exceed or approach my spending limits for specific categories so that I can manage my budget more effectively.*
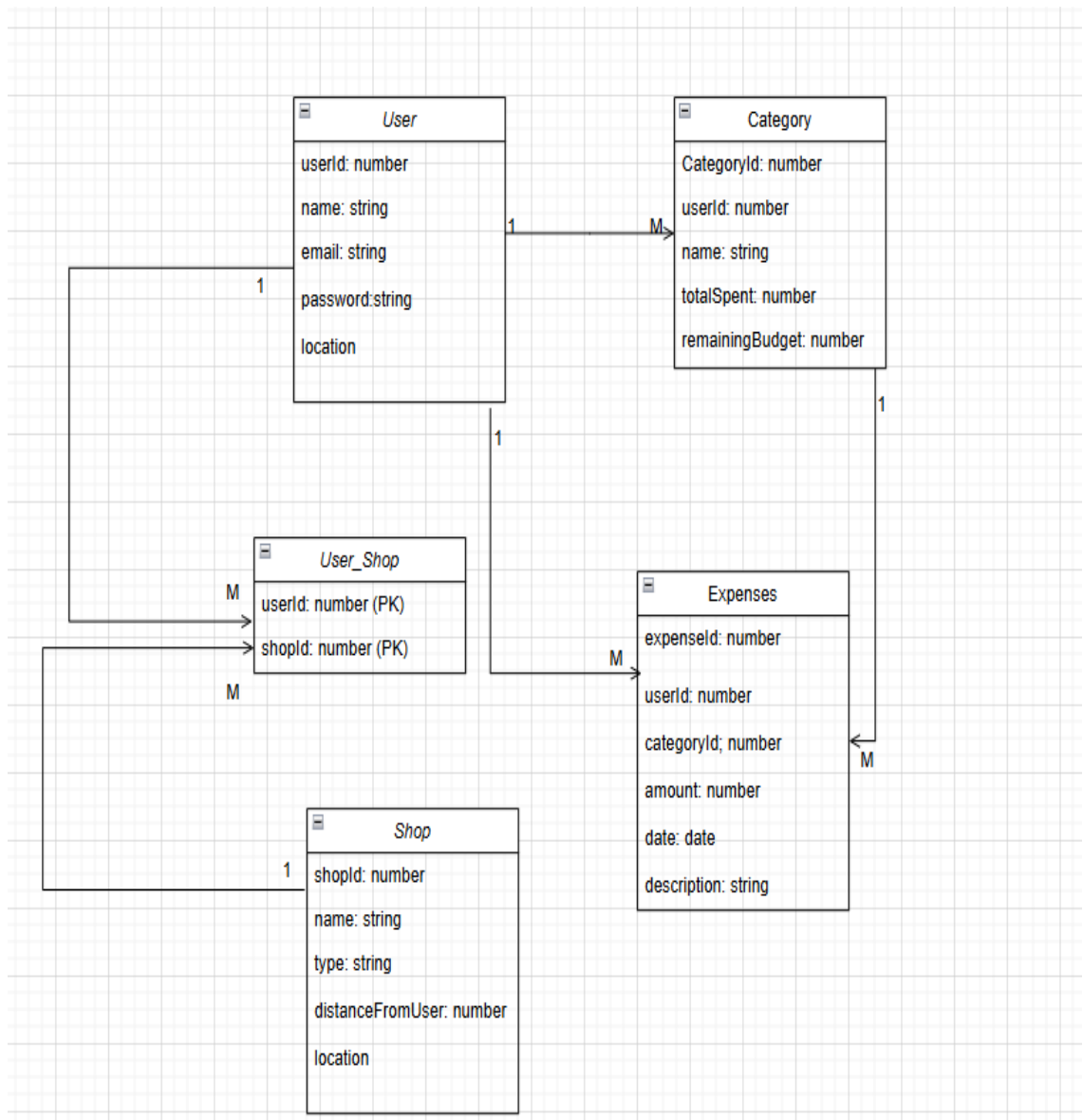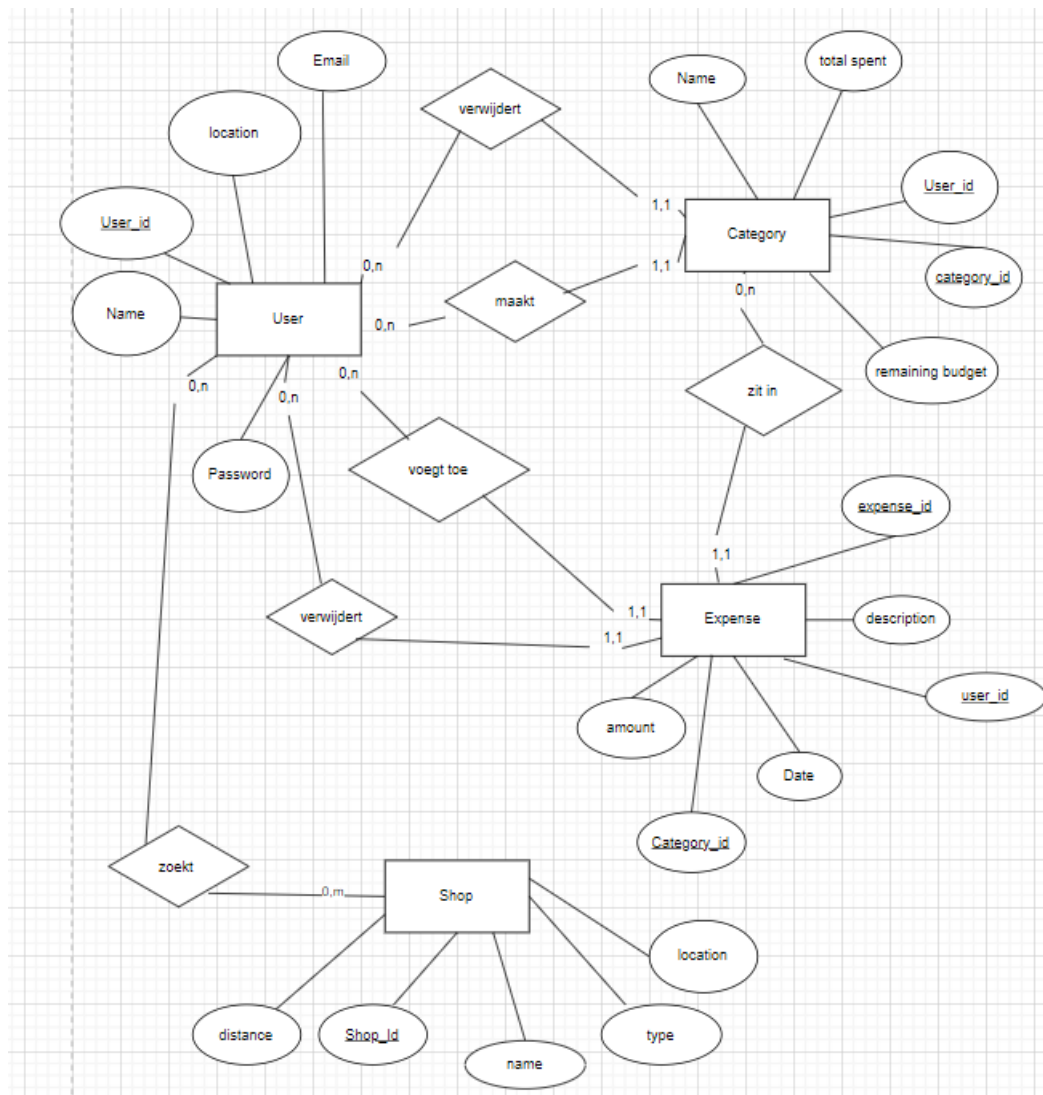
Wireframe:

Acceptance Criteria:

- Users can set a spending limit for each category (e.g., groceries, entertainment).
- The app sends real-time notifications when spending exceeds the threshold or limit.

Domain Model

User
- userId: number
- name: string
- email: string
- password:string
- location

Category
- CategoryId: number
- userId: number
- name: string
- totalSpent: number
- remainingBudget: number

User_Shop
- userId: number (PK)
- shopId: number (PK)

Shop
- shopId: number
- name: string
- type: string
- distanceFromUser: number
- location

Expenses
- expenseId: number
- userId: number
- categoryId; number
- amount: number
- date: date
- description: string

User 1 — M Category

User 1 — M User_Shop

Shop M — 1

User_Shop

User 1 — M Expenses

Category 1 — M Expenses

Conceptual Model

Logical Model