

**Project Proposal: Money Minder**

Ontario Tech University

Course: CSCI 4100U - Mobile Devices

Professor: Dr. Michael Miljanovic

Date: October 20, 2023

## Table of Contents

|   |          |
|---|----------|
| Project Overview.....                       | 2        |
| Group Information & Responsibility.....     | 2        |
| <u>Features and Functionality.....</u>      | <u>2</u> |
| List of Features.....                       | 2        |
| Applying Basic Functional Requirements..... | 3        |
| Extra Challenge.....                        | 4        |
| <u>UML Diagram and Description.....</u>     | <u>5</u> |
| <u>UML Diagram.....</u>                     | <u>5</u> |
| <u>UML Description.....</u>                 | <u>5</u> |
| Mock-up of User Interface.....              | 7        |

## Project Overview

The purpose of this project is to develop a robust and user-friendly mobile application to track expenses, manage budget and analyze spending patterns to gain insights on their personal finances. This app aims to provide users with a comprehensive tool to organize and manage their daily expenses.

## Group Information & Responsibility

- **Aanisha Nawaz (100788588):**
  - **Expenses by Categories:** Build the expense tracking feature, allowing users to categorize their expenses. Implement the ability to create, update, and delete expense categories.
  - **Spending Habit:** Generate insights into users' spending habits, offering statistics and reports through graphs.
  - **Share Functionality:** Enable users to share their financial insights in various formats like Excel, CSV, or via email.
  - **Financial Insights:** Design and implement the graphical representation of financial data for easy visualization.
- **Jahanvi Pareshkumar Mathukia (100832165):**
  - **Stock Market API:** Integrate external APIs to provide real-time stock market data and financial information.
  - **Receipt Scanner:** Develop a feature that allows users to scan and upload receipts for expense tracking and organization.
  - **Design the User Interface:** Responsible for the overall design, look, and feel of the app, ensuring it is user-friendly and intuitive.
  - **Ensure App Cohesiveness:** Make sure that all features and components of the app work together seamlessly and offer a consistent user experience.
- **Jessica Patel (100685837):**
  - **Login/Signup (Recaptcha):** Responsible for developing and implementing the user authentication system, including ensuring the security of user login/signup processes with Recaptcha.
  - **Profile Management:** Create and manage user profiles, allowing users to update their personal information.
  - **Budget:** Implement the budget management feature, enabling users to set, track, and manage their weekly budgets.
  - **Goal Setting:** Develop the goal-setting functionality, where users can set financial goals and track their progress.

- **Mansi Patel (100805283):**
  - **Database (Local and Cloud):** Manage the application's data storage, including local and cloud databases, ensuring data integrity and accessibility.
  - **Multi-User Support:** Implement the capability for multiple users to access and manage their financial data within the app securely.
  - **Design the User Interface:** Responsible for the overall design, look, and feel of the app, ensuring it is user-friendly and intuitive.
  - **Ensure App Cohesiveness:** Make sure that all features and components of the app work together seamlessly and offer a consistent user experience.
- **Extra Responsibilities for Everyone:**
  - **Bill & Location Reminders:** Collaboratively work on the feature that enables users to set reminders for bill payments and location-based reminders for financial tasks.
  - **Budget Reminders (Over Limit):** Implement a notification system that alerts users when they exceed their budget limits.

## Features and Functionality

### List of Features

- Expense Tracking by Categories:
  - The primary goal of this application is to allow users to track their expenses by different categories (e.g.: University fees, Rent, Mobile bill, etc.) to provide clear and organized information about their daily expenses.
- Income Tracking by Categories:
  - This feature will enable users to track their income by aggregating all the investments to provide a comprehensive view of their financial status.
- Budgeting:
  - This app will provide users the ability to create and manage their budget by different categories. It will allow users to set financial goals against their budget and will provide notifications and alerts to help users stay within their budget.
- Financial Insights and Analysis:
  - This feature will provide users with detailed insights into their spending habits and trends through graphical representation.

- Bill Reminders:
  - This feature will notify users about their upcoming bills, payment due dates, and other financial commitments to ensure they are on top of their financial obligations and avoid any late fee charges.
- Financial Goal Tracking:
  - This app will provide users with functionality to set and monitor short and long-term finance goals (e.g.: mortgage of the house, pay off student debt, etc.).
- Expense Sharing:
  - This app will provide users with a feature to share their expenses and financial insights with trusted individuals, such as family members. Users will also be able to track family member's expenses by switching member accounts, helping families meet their financial goals together.
- User-Friendly Interface and Navigation:
  - This app will ensure users can easily access all the features and will allow users easy navigation to different screens.
- Security:
  - This app will ensure the privacy of their financial information. Each user will have their own email and password information to access their account.
- Accessibility:
  - Users will be able to add/modify the expense details and will be able to access their budgets 24/7.

## Applying Basic Functional Requirements

The application will adapt the course's required functionalities to ensure high-quality development.

- Dialogs and pickers:
  - Money Minder will include dialogs and pickers to let its users select the date to upload their purchases by category if they are updating their spending history from the past. Additionally, users can select the date (including a range of dates) to view their spending history in the form of lists, charts, and graphs of their spending habits during the desired time frame by category. Lastly, users will be able to add/modify their expense categories as needed. A similar implication will be done for income tracking, goal, and budget setting.
- Multiple screens and navigation:

- Money Minder will let its users easily navigate through multiple screens to access individual features. Features with individual screens include a view of daily spending, access to long-term spending with graphics, uploading recent spending, etc.
- Snack bars:
  - Snack bars will be implemented in the application to provide its users with confirmation after updating their daily budget and uploading recent spending.
- Notifications:
  - Money Minder will have notification options to remind its users about upcoming bills.
- Local storage (SQLite):
  - Money Minder will have daily local storage for the user to instantly access any time of the day to view their daily spending habit.
- Cloud storage (Firebase or other):
  - This financial mobile app will store all spending and budgeting data in the cloud (Firestore) by refreshing all device-stored data every 24 hours. This will ensure the application is not taking up too much storage on the user's device while also letting the user access their spending data through charts and graphs whenever needed.
- HTTP requests:
  - The mobile app will use HTTP requests to get information about stock markets so its user to easily keep track of their investment.

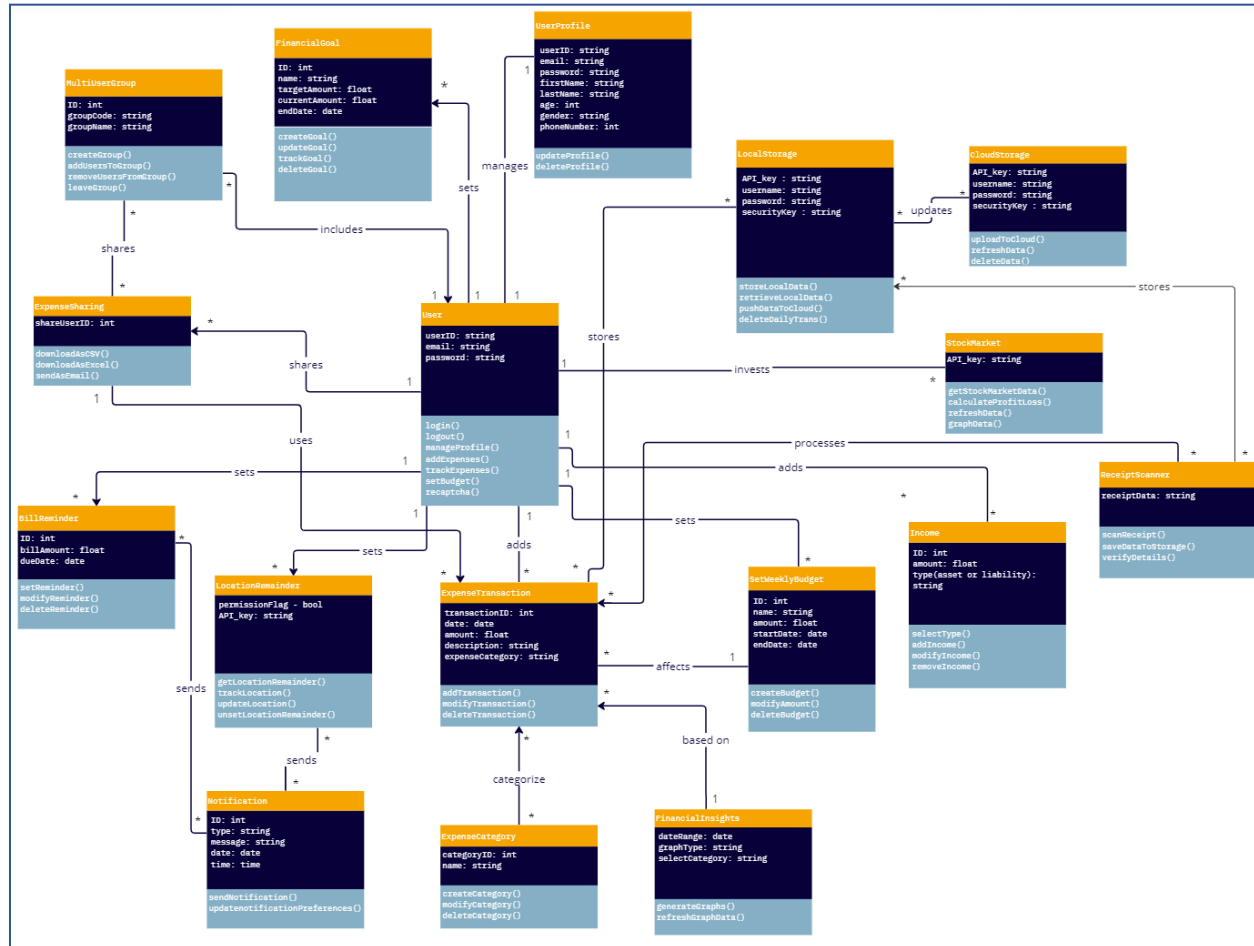
### Extra Challenge

There will be three additional features added to Money Minder depending on the level of complexity and the timeframe of this project.

- Receipt scanning with camera:
  - Money Minder will have a camera feature in the application to make uploading recent spending easy. The camera function will scan receipts and update the spending history based on the date and time, categorizing the spending as well.
- Spending upload reminder based on user location:
  - The application will adapt geolocation so that users will get notifications after leaving a store or restaurant, reminding them to update their daily spending. This will ensure that its users are staying on top of their spending habits and accurately meeting their financial goals.
- Multiple Users Support:
  - Multiple users can join the same group with a unique code and track the expenses together (e.g.: couples can share their expenses in the same group, etc.).

# UML Diagram and Description

## UML Diagram



## UML Description

Here's a brief explanation of each class represented in the UML diagram with their functionality:

- **User:**
  - This class represents a user of the application. Each user needs to enter their email and password to log in to the app. Users can log in, log out, manage their profile, set budgets, add their expenses, track their expenses, and authorize through recaptcha. This is the most important class as this is the entry point for any user to be able to use the application.
- **UserProfile:**

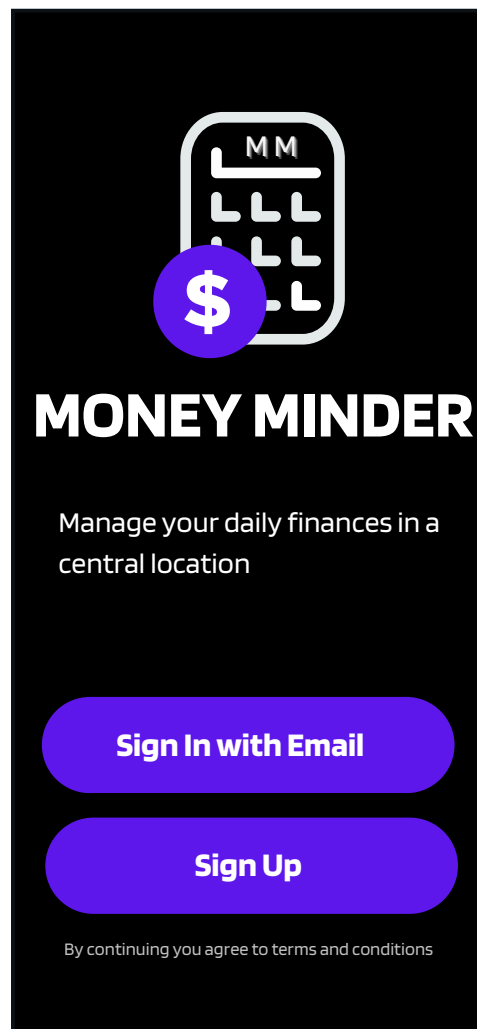
- This class stores additional information about the user like name, age, and contact details, including email and password, enhancing the user's profile. Every user can manage their user profile.
- **FinancialGoal:**
  - In this class, the user can set a new financial goal by giving it a name (eg: savings goal for 2023), entering the target amount that the user wants to save, the current amount, and the last date before which the user wants to achieve this target.
- **MultiUserGroup:**
  - The Purpose of this class is, that each user can create a group in the app by adding other people like their family members, etc., They can create a group by giving the group a name, and entering a unique group code. In addition to creating a group, users can add users to the group, remove users from a group, and leave the group.
- **ExpenseTransaction:**
  - The user can add their daily expenses through this class. To add the expense transaction, the user will enter the date, amount, description, and category of the expense.
- **ExpenseCategory:**
  - This class has different types of categories, which categorize the expense transactions entered by the user (ex.: mobile bill, rent, etc.). The user can create new categories, update categories, and delete categories.
- **FinancialInsights:**
  - Purpose of the class is to provide financial insights to a user for their overall spending habits through different types of charts and graphs. This class will provide a graphical representation of the user's expenses, income, and financial goals.
- **ExpenseSharing:**
  - This class uses the expenseTransaction class to get the list of transactions that the user and multi-user group has in the database and will be able to download that data as CSV, or Excel files or can share via email.
- **Notification:**
  - This is a generic class that will send all types of notifications with a message, date, and time.
- **BillReminder:**
  - The user can set a bill reminder to get notification before the due date. The user can also modify and delete the remainder. The bill reminder sends notifications through the Notification class.
- **LocationReminder:**
  - The user can set up location reminders to get notified whenever the user visits any shopping places to add their expenses to the app. The user can also update and




unset the location reminder. The reminders will be sent through the Notification class.

- **SetWeeklyBudget:**
  - Purpose of this class to create a weekly budget with a name, amount, and start and end date. The user can create, modify, and delete as many weekly budgets as they like.
- **Income:**
  - Purpose of the Income class is to add their net income to determine their financial goals and weekly budgets. They need to enter the amount, and type (asset or liability). The user can modify and delete their income if they wish to.
- **ReceiptScanner:**
  - The user can also add their expenses by scanning the receipts through their camera. The scanner will process the expense transactions and will add to them based on the category. All the receipt scanner data will be stored in the local storage.
- **StockMarket:**
  - Purpose of this class is to fetch the current stock market prices and allow the user to track the profits and losses on their investments.
- **LocalStorage:**
  - This class will store all the daily expense transactions and receipt scanner uploads in the local storage for a day. Every night, the local storage data will be uploaded to the cloud storage. Local storage will also push all the other data like, the goals, incomes, budget, etc.,
- **CloudStorage:**
  - Every 24 hours, all the data from the local storage will be pushed to save it in the cloud for a longer time period. This data will be used to provide financial insights.

## Mock-up of User Interface




This is the first landing page. The user is asked to sign in or sign up with email and password.



# Welcome Back!

Enter your email and password to access your account.

Email


Password  

☐ Remember Me [Forgot Password?](#)

[Log In](#)

Don't have an account? [Sign Up](#)

The user can enter their email and password over here to enter the application.




# Create Account


Let's create an account

Email

Full Name

Username


Password    
Must contain a number and least of 6 characters

Confirm Password    
Must contain a number and least of 6 characters

**Sign Up**

Have an account ? **Log In**

This is a sign up page. The user can create a new account here.



# Oh, no! I forgot

Enter your email, and we'll send you a link to  
change a new password




Email

Forgot Password

Don't have an account?


[Sign Up](#)

This is the forgot password page



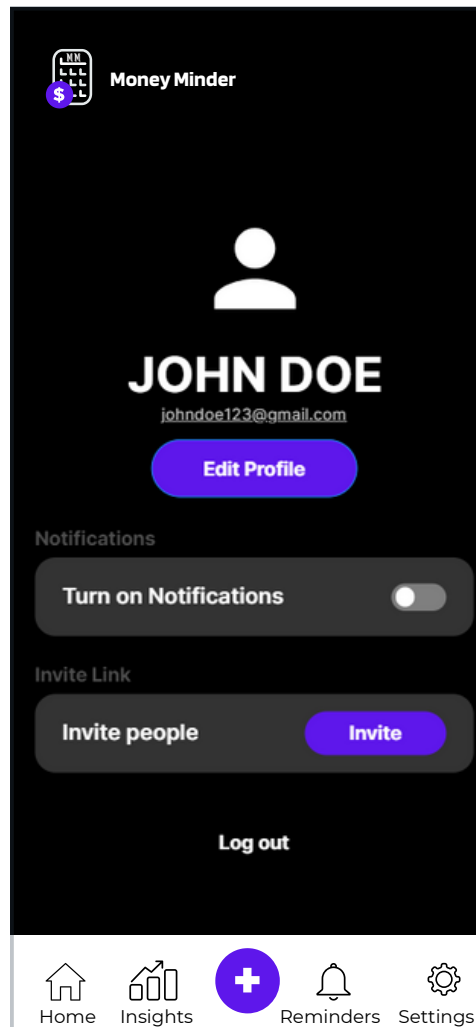
## Verification Code

Enter your verification code that we sent through your e-mail or phone number

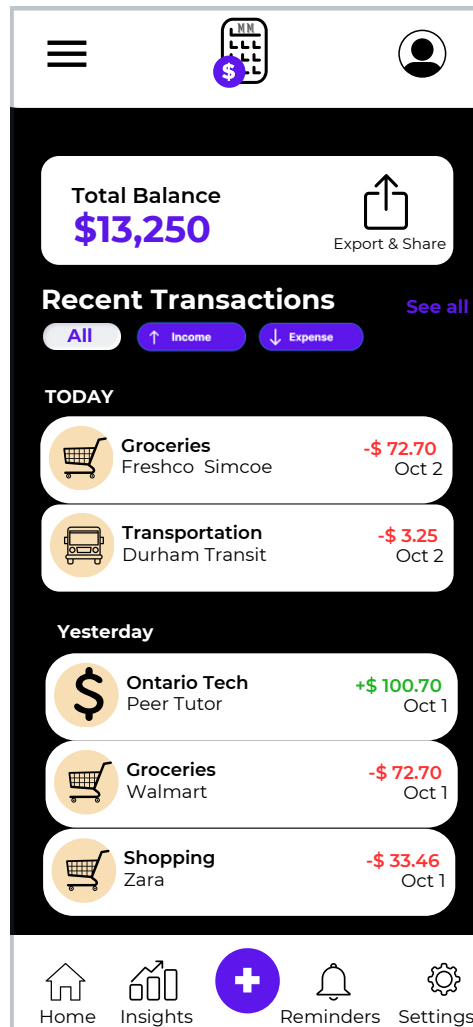
 3:12

Submit

The user will get verification code on their email, after that they can set their new password (from forgot password page).

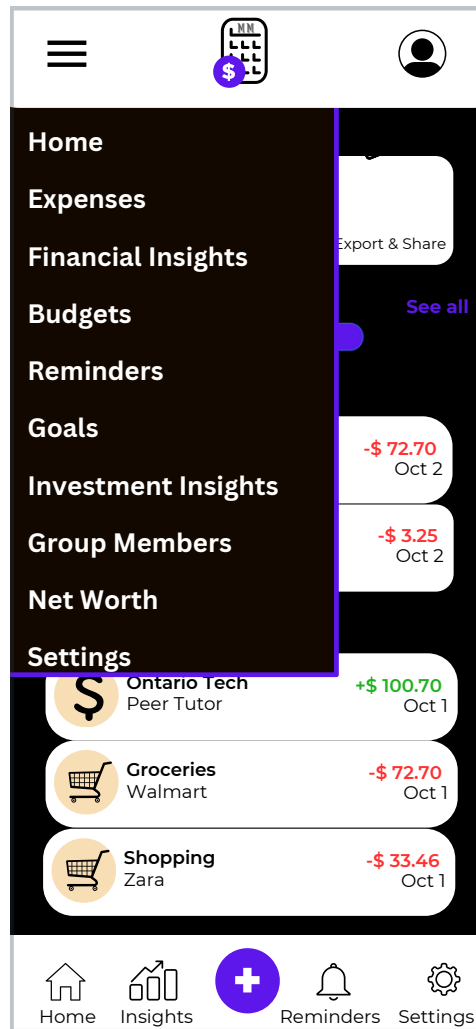


This is the settings screen. The user can update their profile and preferences here.

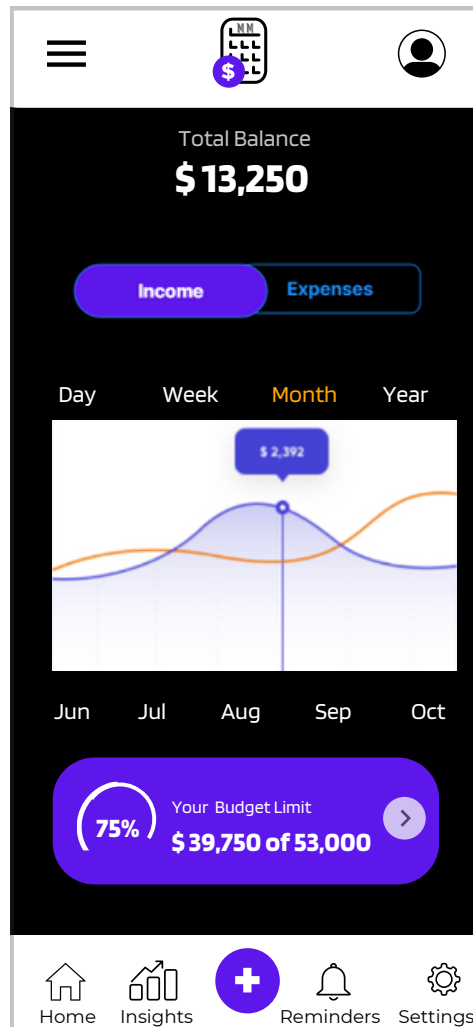


The main screen after user sign in, where they can add their transactions.

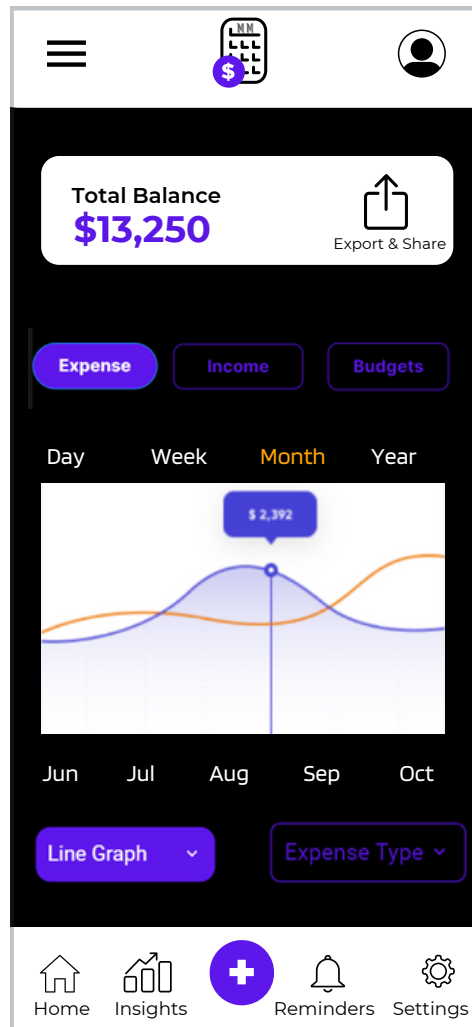




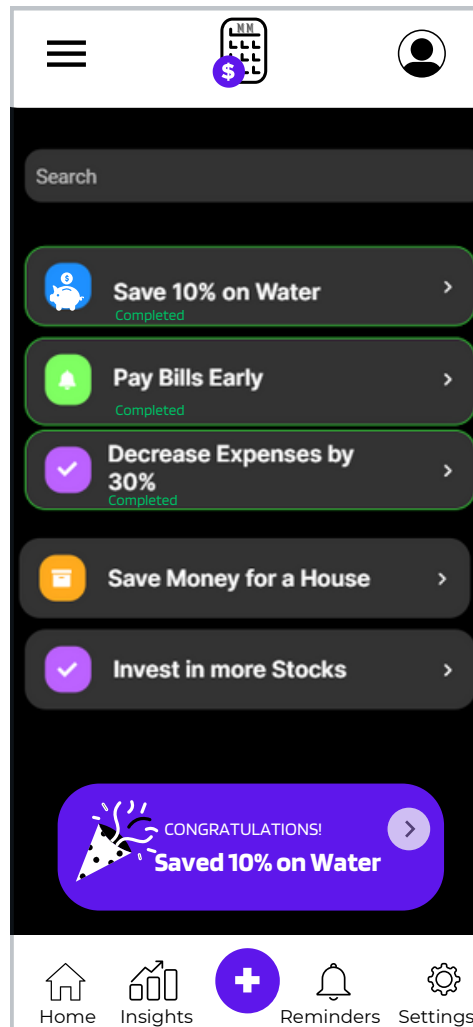
The side menu bar gives all required options.



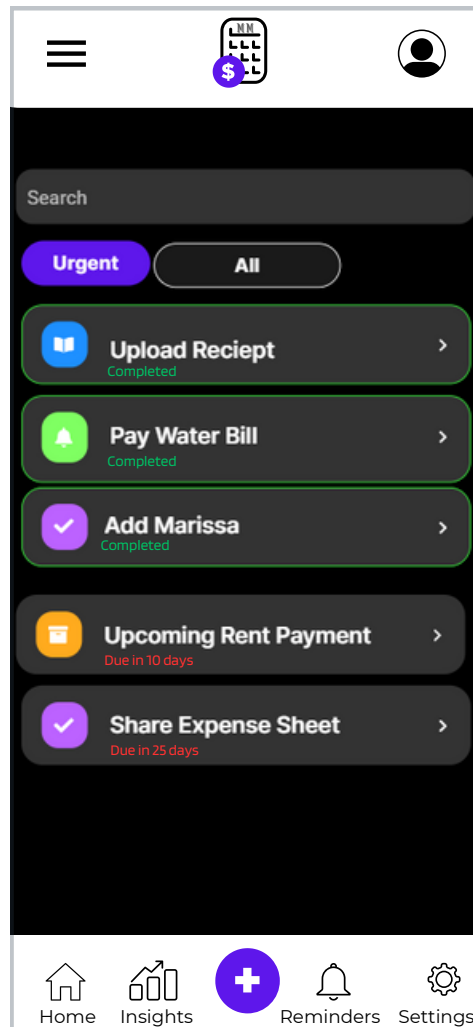
This screen shows financial insights for recent transactions.



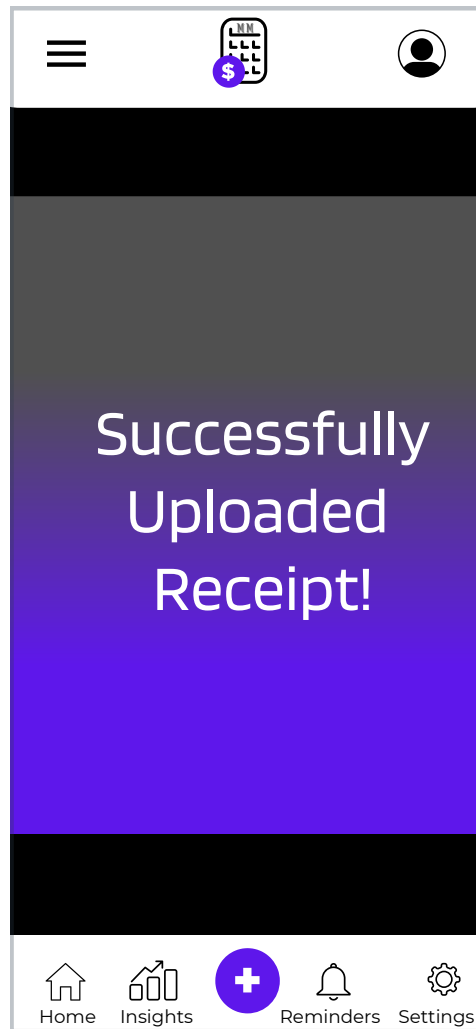
This is screen for financial insights where user can see the graphs for their income, expenses, and budgets.



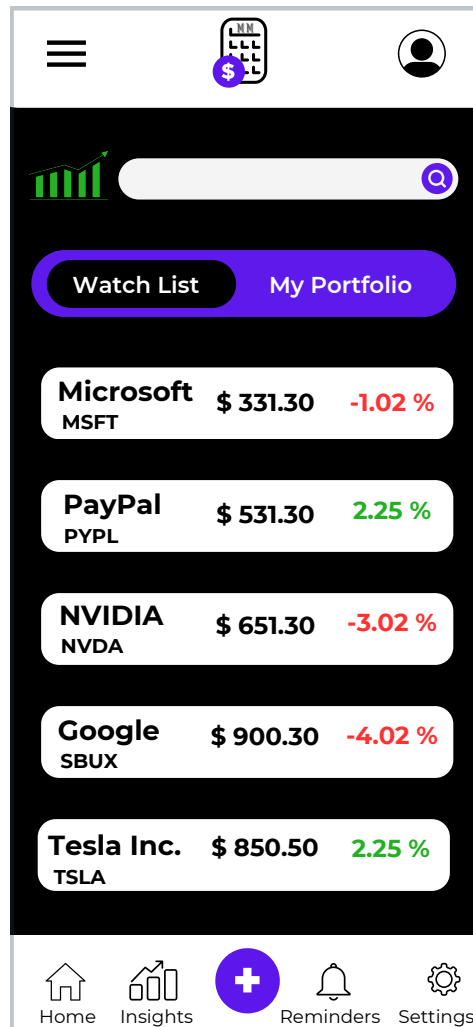
This is the goals screen where user can set different goals to achieve.



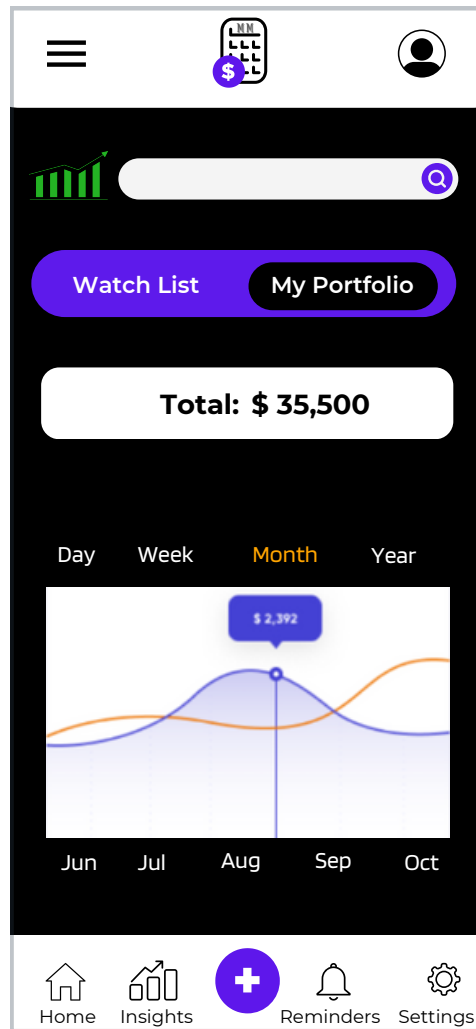
This is the reminders screen where user can set different reminders



This is conformation screen which user will be able to see after the successfully uploading the receipt.

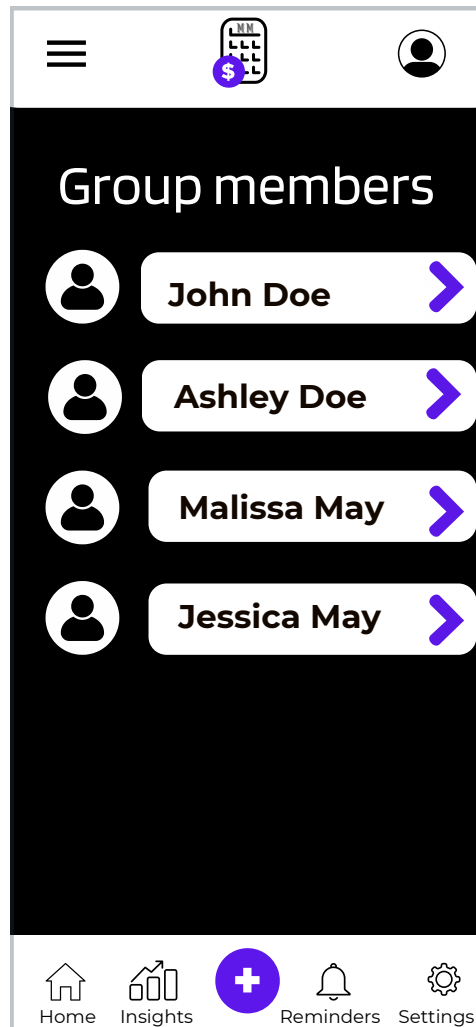


This is the screen for Investment insights where the live stock prices will be displayed.

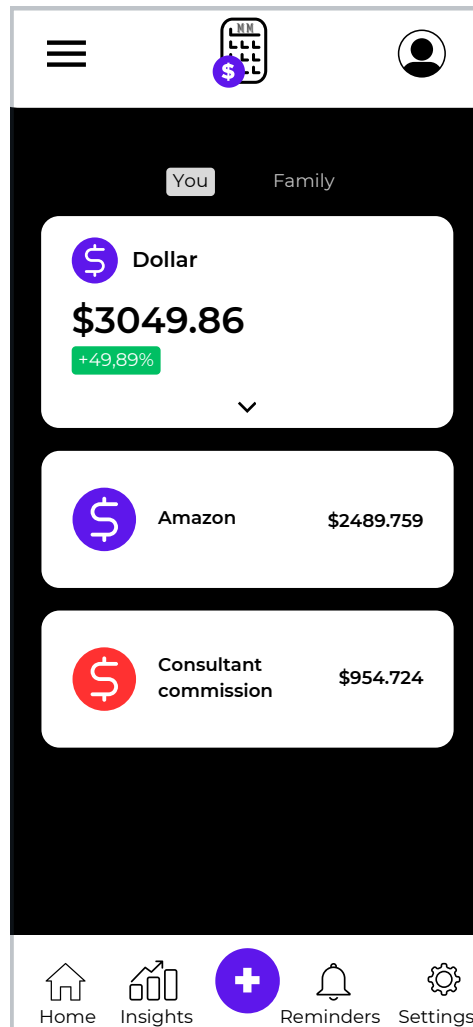


This screen shows investment portfolio of the user.

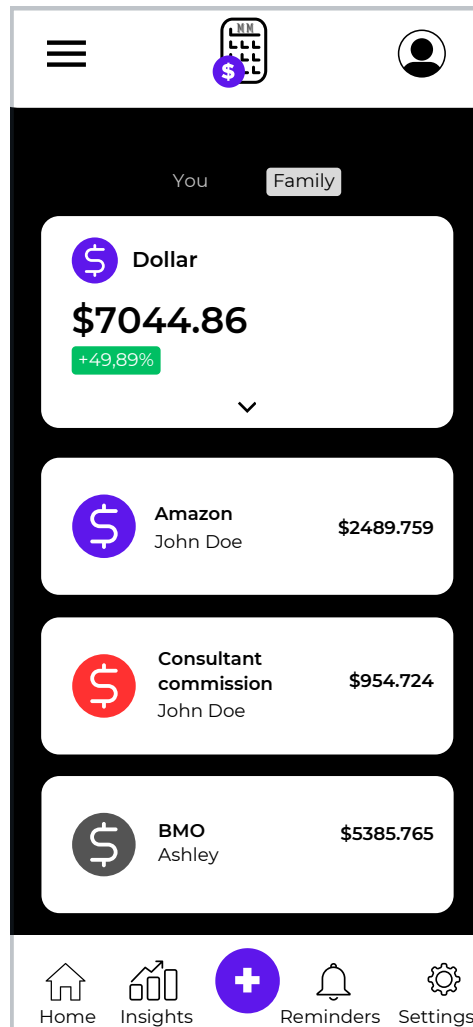




This is the screen for Group members.



This is the screen for Income/NetWorth



This is the screen for Income/NetWorth of all the family members