Software Requirements Specification

Group 8

Expense Monitoring System

Software Engineering
Fall 2023

CS673 A1

BOSTON UNIVERSITY

Abhishek Gupta
Siddharth Krishnakumar
Julia Peterson
Jiuzhou Lu
Haolong Yang

1. Introduction

1.1 Purpose

The purpose of this SRS document is to provide a detailed description of the functionalities and requirements of the Expense Monitoring System. The Expense Monitoring system provides users with a comprehensive and interactive platform to monitor, track, categorize, and analyze their expenses by setting a budget limit monthly or yearly. This also helps the user study and track their spending patterns and revisit their spending on a monthly or yearly basis.

1.2 Scope

The system aims to provide users with an interactive platform to monitor, track, categorize, and analyze their personal and group expenses while facilitating budgeting and bill-splitting functionalities.
The scope of the project includes:

- Users would be able to register or log in through a secured platform and manage their accounts.
- Users would be able to input, edit, categorize, and view their expenses and expense summary.
- Users would be able to create a group track and split the expenses and budget of a group.
- Users would be able to visualize an overall analytics dashboard containing their expenses based on categories.
- Users would be able to able to access the application using a mobile device or through a browser as well.

2. System Features and Functionalities / Functional Requirements

2.1 User Registration and Authentication

2.1.1 Users shall be able to register with the system using an alphanumeric user ID and passwords.

2.1.2 User authentication shall be in place to ensure that only authorized users can access the system.

2.2 User Roles and Permissions

2.2.1 The system shall support multiple user roles, such as administrators and regular use.

2.2.2 Each role shall have specific permissions and access levels within the system.

2.3 Individual Expenses Track

2.3.1 Users shall be able to input expenses, including date, amount, description, and category.

2.3.2 The system shall support the addition of receipts or attachments for expense verification.

2.4 Group Expenses Track

2.4.1 Users shall be able to create groups to track the expenses and budgeting of each group.

2.4.2 Users shall be able to invite other users to join the group.

2.4.3 Users as a member of a group shall be able to input expenses, including date, amount, description, and category of the group.

2.4.4 The system shall support the expense split function.

2.4.5 The system shall display the split expenses to the individual expense section of each user in the group.

2.4.6 The user shall be able to select splitting methods, including equal sharing and custom percentages.

2.5 Bill Splitting

2.5.1 Feature to divide bills with platform users.

2.5.2 Support for equal and custom percentage splits.

2.5.3 Settlement summary for clearing shared expenses.

2.5.4 Inclusion of individual's share in their standalone expenses.

2.6 Report Services

2.6.1 Users shall be able to see a detailed summary of expenses monthly.

2.6.2 The system shall support a dashboard for the visualization of expenses.

2.7 Notification Services

2.7.1 The system shall send email notifications to all group members when a group member adds expenses to a group.

2.7.2 The system shall send an email notification to all group members when a new member joins the group.

2.8 Data Security and Privacy

2.8.1 The system shall employ some kind of encryption for data transmission and storage.

2.8.2 Regular data backups and disaster recovery procedures shall be in place.

2.8.3 Implementation of robust security measures to maintain user data confidentiality.

2.9 Dashboard and Report Services

2.9.1 Users shall have the ability to export their expense data for offline use or reporting purposes.

2.9.2 Analytics dashboard for enhanced visual insights.

2.9.3 Detailed expense summary report for users.

3 Non-Functional Requirements

3.1 Performance

3.1.1 The system shall be responsive, supporting concurrent user access.

3.1.2 The system should respond to user interactions within 2 seconds on average.

3.2 Availability

3.2.1 The system must be available 24/7, with scheduled downtime communicated in advance for maintenance.

3.2.2 Availability should be at least 99% uptime.

3.3 Scalability

3.3.1 The system architecture shall be designed to accommodate future growth in users and data volume.

3.3.2 Scalability testing should be performed to ensure smooth expansion.

3.4 Security

3.4.1 The system shall employ some kind of encryption for data transmission and storage.

3.4.2 The system must implement strong password policies and provide account lockout after a specified number of failed login attempts.

3.4.3 Access to sensitive data and functions should be restricted based on user roles and permissions.

3.5 Data Backup and Recovery

3.5.1 Regular automated backups of the database shall be performed.

3.5.2 A disaster recovery plan should be in place to ensure data can be restored in case of system failures.

3.6 Compatibility

3.6.1 The system should be compatible with commonly used web browsers (e.g., Chrome, Edge, Safari, Firefox).

3.7 Load Handling

3.7.1 The system should be able to handle a load of at least 10000 concurrent users during peak usage periods.

4 User Stories

4.1 User Authentication

4.1.1 As a new user, after opening the application I should be able to create an account or sign up and login to access the expense monitoring system application.

4.1.2 As a user, I should also be able to recover my account, if my password is lost.

4.2 Home Page Access

4.2.1 As a user, I should be able to access the home page once I log in to the expense monitoring application.

4.2.2 As a user, I should be able to navigate inside the homepage application to view personal expenses, group expenses, and my overall dashboard and edit my profile settings anytime.

4.3 Personal expenses access

4.3.1 As a user, I should be able to go into my personal expense tab and be able to input my expenses, and be able to categorize the expenses into different categories. E.g. (Entertainment, Academics, Groceries, Miscellaneous).

4.3.2 As a user, I should also be able to set a budget limit to track my sending and be able to get a notification alert when I am about to cross the budget set.

4.4 Group expenses access

4.4.1 As a user, I should be able to go into the group's tab, create groups, view the number of groups that I am a part of, and be able to input expenses under different categories. I should also be able to split them between the groups according to the number of people involved in the expense.

4.4.2 As a user, I should also be able to invite people to join the expenses group through a personalized link.

4.5 Bill Splitting

4.5.1 As a user who paid for a shared expense, I want to split the bill with others so that everyone pays their fair share.

4.5.2 As a participant in a bill split, I want to see the breakdown of what I owe and to whom.

4.5.3 As a user, I want to settle shared expenses to clear any dues.

4.6 Dashboard Access

4.6.1 As a user, I should be able to navigate to my personal dashboard page and view my overall numbers, and graphical representation of my expenses and be able to export my expenses as a file based on any number of months or years.

4.7 Notification Service

4.7.1 As a group member, I want to receive an email notification when a new expense is added to our group, keeping me updated.

4.7.2 As a user, I want to get an email notification when I'm invited to a group so that I can join and participate.

4.8 Data Privacy and Security

4.8.1 As a user, I want my data to be stored securely, ensuring my personal and financial information is protected.

4.8.2 As a user, I want to know that my interactions with the platform are encrypted and secure.

5 Diagrams

5.1 Sequence Diagram:

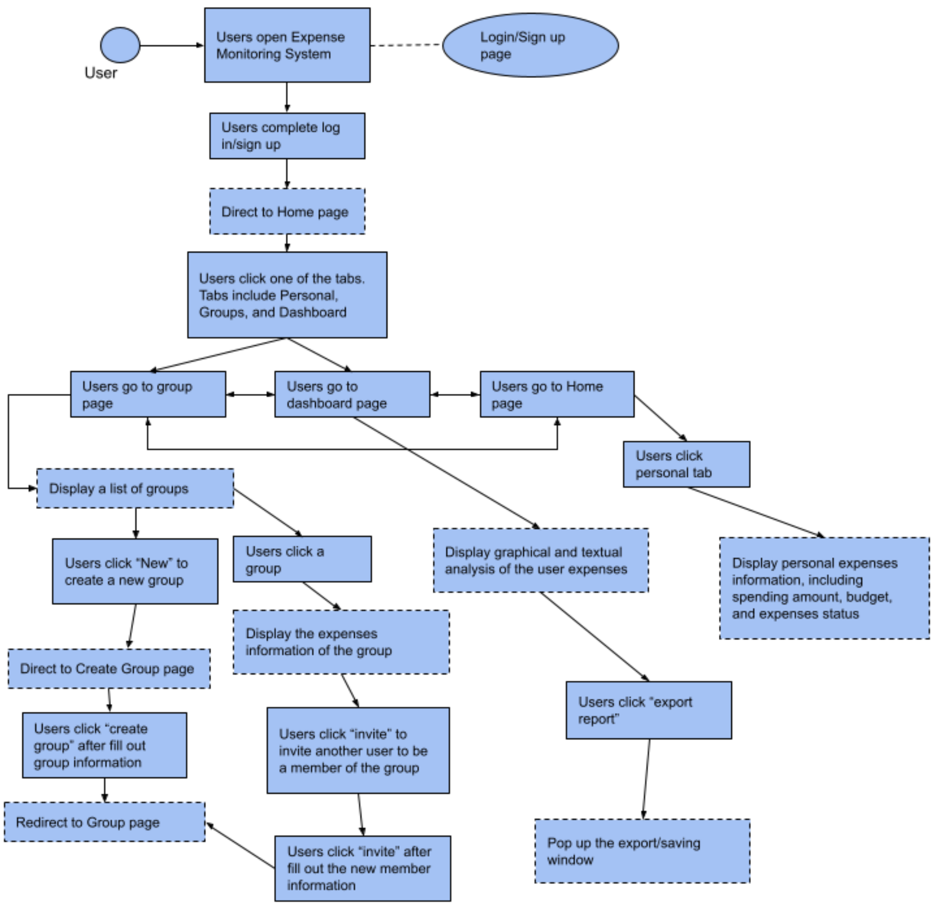

Fig: Sequence diagram.

5.2 Use Case Diagram:

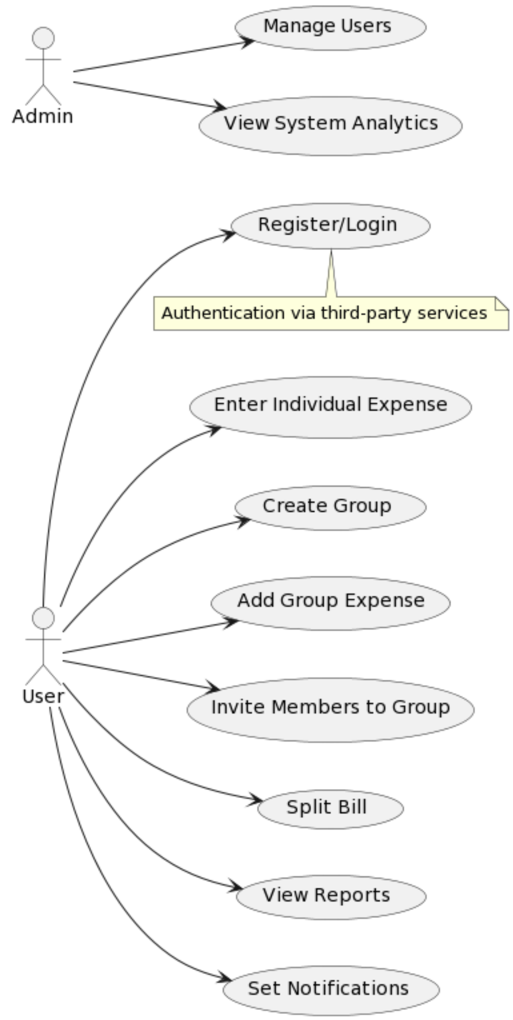

Fig: Use case diagram.

5.3 Data Flow Diagram

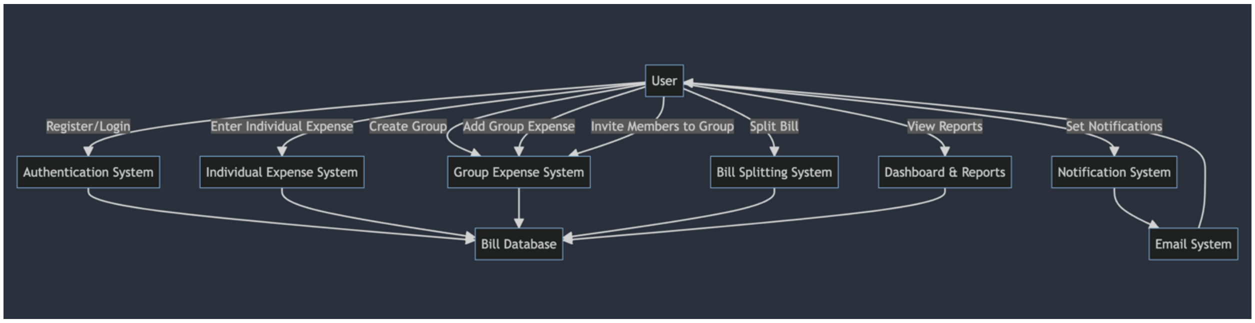

Fig: Data Flow Diagram.

5.4 High-level API Design:

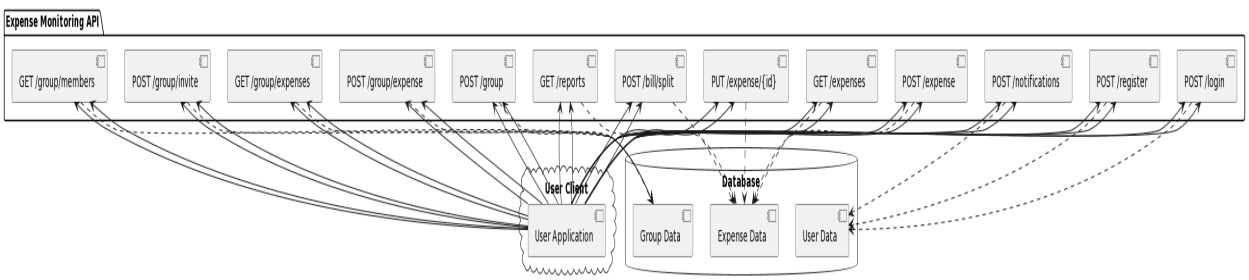

Fig: API Design.

5.5 High-level UX Design

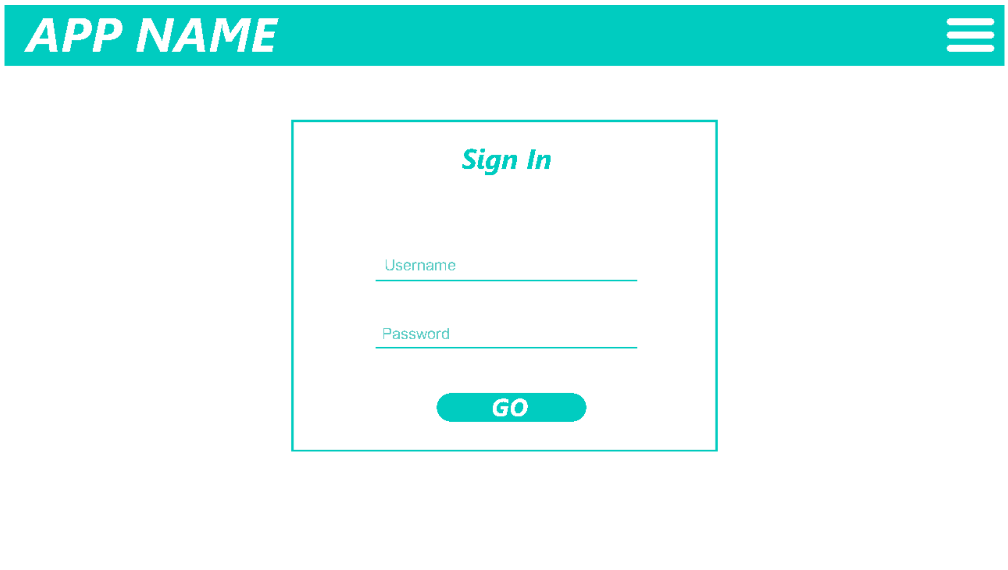

Fig: Login Page

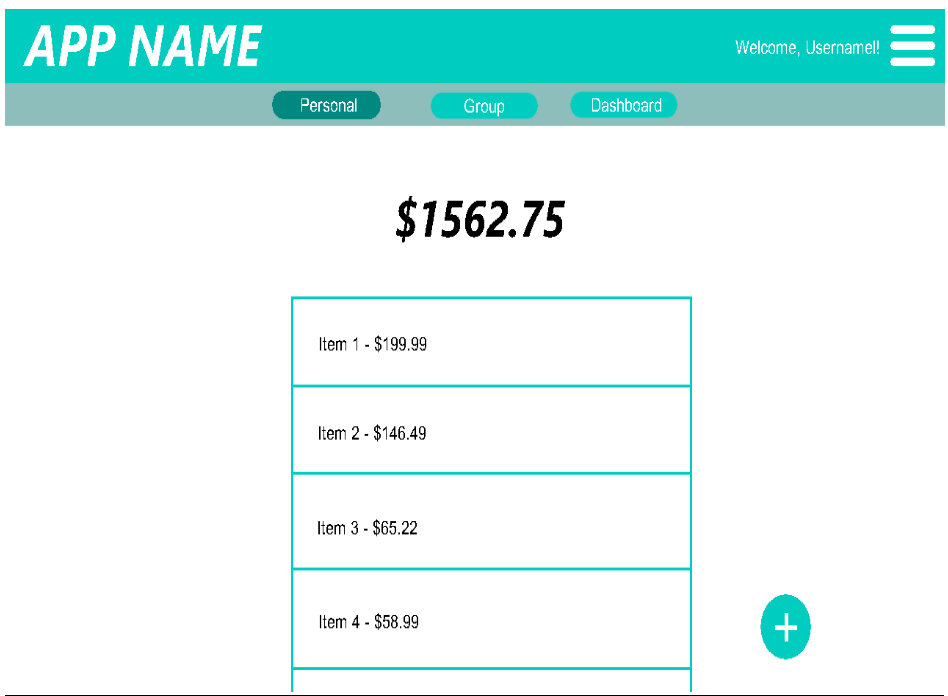

Fig: Personal Tab

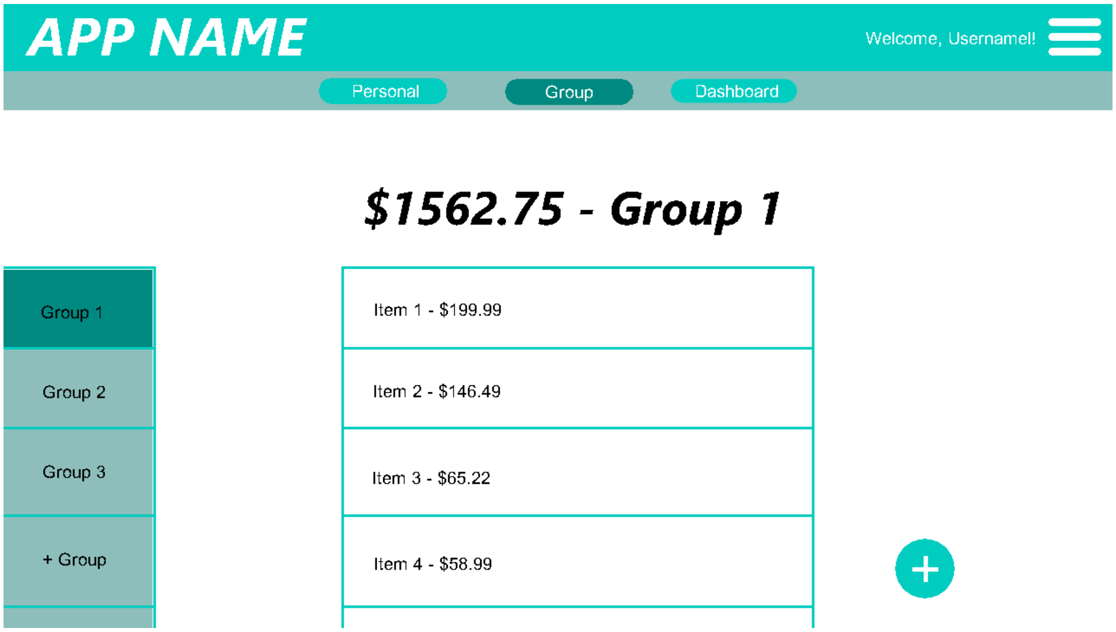

Fig: Group Tab

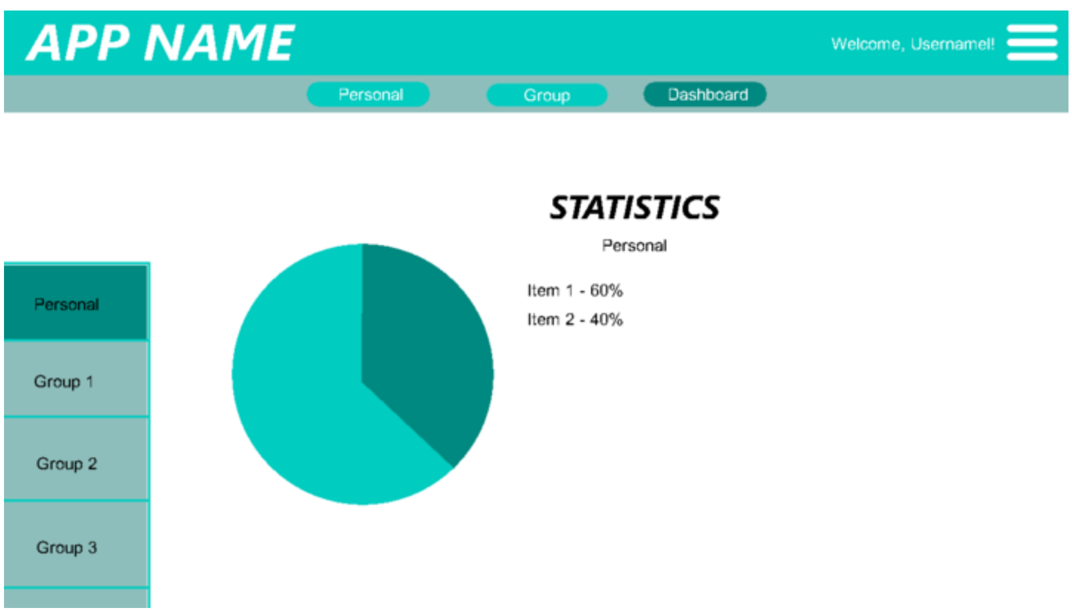

Fig: Dashboard

6 External Interface Requirements

6.1 User Interfaces

6.1.1 The application runs on the latest version of Chrome browser on Windows and MacOS.
6.1.2 The application contains 6 pages:
6.1.2.1 Login page: support sign-in and sign-up functionality.
6.1.2.2 Home page: display menu, user personal spending amount, user budget, and expenses status.
6.1.2.3 Groups page: display group expense information, including group spending. group budget: group expenses status, etc.
6.1.2.4 Create Group page: allow users to create groups, and fill out group information.
6.1.2.5 Invite Page: invite other users, fill out other user information, etc.
6.1.2.6 Dashboard page: display graphical and textual analysis of the user expenses.

6.2 Software Interfaces

6.2.1 Database: MongoDB as NoSQL database and MySQL/Postgres as SQL database. (May vary depending on technical changes, performance, and complexities).

6.2.2 Operating System: Windows, Mac, Chromebook, Linux etc.

6.3 Communication Interfaces

6.3.1 E-mails and embedded forms of communication will be used.

7 References

- https://asana.com/resources/software-requirement-document-template
- https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document
- https://www.bmc.com/blogs/software-requirements-specification-how-to-write-srs-with-examples/