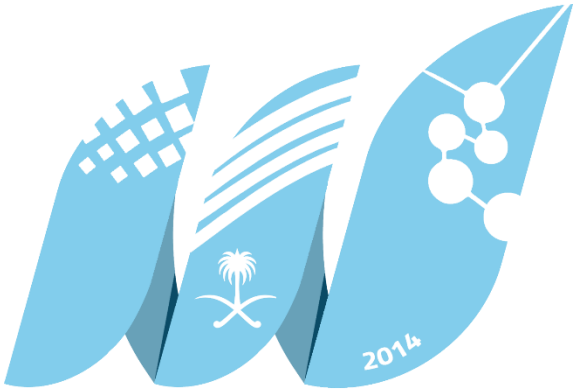


جامعة جدة

University of Jeddah

College of Computer Science and Engineering

Department of Software Engineering

Software Process Models

CCSW 315

Sprint #0 - Wallet Guard

Team Members	1. Lama Munir Noor | 2211796 2. Lama Fathi Akbar | 2212609 3. Joud Jalal Batarfi | 2210353 4. Raghad Al-saiari | 2210207 5. Ranad Waleed Aljhdali | 2211806 6. Hajar habiburahman | 2211350
Section	E2

Contents

Project Initiation

Project Description

The **Wallet Guard** is a decentralized application developed to assist individuals in managing their monthly expenses and budgets with greater efficiency. The app enables users to input recurring expenditures, such as transportation, fuel, subscriptions, and other monthly costs, and categorize them. It monitors spending throughout the month, providing insights into the user's adherence to their planned budget. At the end of each month, the app generates a comprehensive analytics report, detailing total expenses, identifying areas of overspending. This tool is ideal for individuals seeking to improve their financial habits and maintain control over their spending.

Problem Definition

Many individuals face difficulties in managing their finances effectively, often losing track of where their money is being spent. Without clear visibility into their monthly expenses, staying within a planned budget can be challenging, leading to overspending and financial pressure. Traditional methods, such as spreadsheets, are time-consuming and rely on users consistently updating them. There is a clear need for a user-friendly solution that provides budget management to promote financial discipline and control.

The Solution

The **Wallet Guard** offers a streamlined and efficient solution to these financial challenges by providing a platform that:

- Allows users to input and categorize their monthly expenses.
- Enables users to set spending limits for different categories.
- Allow users to add expenses, alerting users when they are nearing their budget limits.
- Generates detailed monthly analytics reports that summarize total spending.

In summary, this application empowers users to maintain control over their finances, improve their budgeting habits, and make informed financial decisions.

The Scope

The scope of the **Wallet Guard** includes:

1. **Expense Input and Categorization:** Users can enter and categorize their monthly expenses (e.g., transportation, groceries, subscriptions).
2. **Budget Planning:** The application allows users to establish monthly spending limits for each expense category.
3. **Monthly Reports**: The app generates detailed reports at the end of each month, offering insights into total spending and areas that require budgetary improvements.
4. **Notifications:** The system sends provides alerts when users approach their budget limits.

Out of Scope:

5. **Automated Expense Entry:** The app will not be linked to bank accounts or other external systems for automatic expense entry.
6. **Investment Management:** The focus of the app is solely on expense tracking, without support for tracking investments.

The Users

Individuals Seeking to Manage Their Personal Finances:

- Users who need a comprehensive tool to track both their expenses and savings in order to maintain financial control.
- Individuals aiming to improve their budgeting habits and achieve their savings goals.

People Focused on Achieving Savings Goals:

- Users who are looking for a straightforward way to monitor their progress towards specific savings targets while managing daily expenses.

Product Backlog

Functional Requirements

1. **Expense Input**
   1.1. The system shall allow users to manually input expenses.
   1.2. For each expense entry, the system shall require the following information:
      - Name
      - Amount
      - Date
      - Occurance (once, daily, weekly, monthly, annualy)
      - Description
   1.3. The system shall automatically create expense entries for recurring transactions on their due dates.
2. **Budget Management**
   2.1. The system shall allow users to set an overall monthly budget.
   2.2. The system shall provide an interface for users to easily adjust their monthly budget.
   2.3. The system shall allow users to set individual spending limits for each category.
   2.4. The system shall track spending against the overall budget and category-specific budgets.
3. **Category Management**
   3.1. The system shall allow users to create custom expense categories.
   3.2. The system shall provide a set of default categories for users to choose from.
   3.3. The system shall allow users to edit or delete custom categories.
   3.4. The system shall not allow deletion of categories that have associated expenses.
4. **Expense Categorization**
   4.1. The system shall allow users to assign each expense to a specific category.
5. **Expense Editing and Deletion**
   5.1. The system shall allow users to edit any field of an existing expense entry.
   5.2. The system shall allow users to delete any expense entry.
   5.3. The system shall maintain a log of all edits and deletions for auditing purposes.
6. **Reporting and Visualization**
   6.1. The system shall provide visual representations of budget status and spending patterns.

6.2. The system shall generate pie charts showing spending distribution by category.
6.3. The system shall generate bar graphs comparing actual spending to budgeted amounts.
6.4. The system shall generate line graphs showing spending trends over time.
6.5. The system shall allow users to select custom date ranges for all reports.

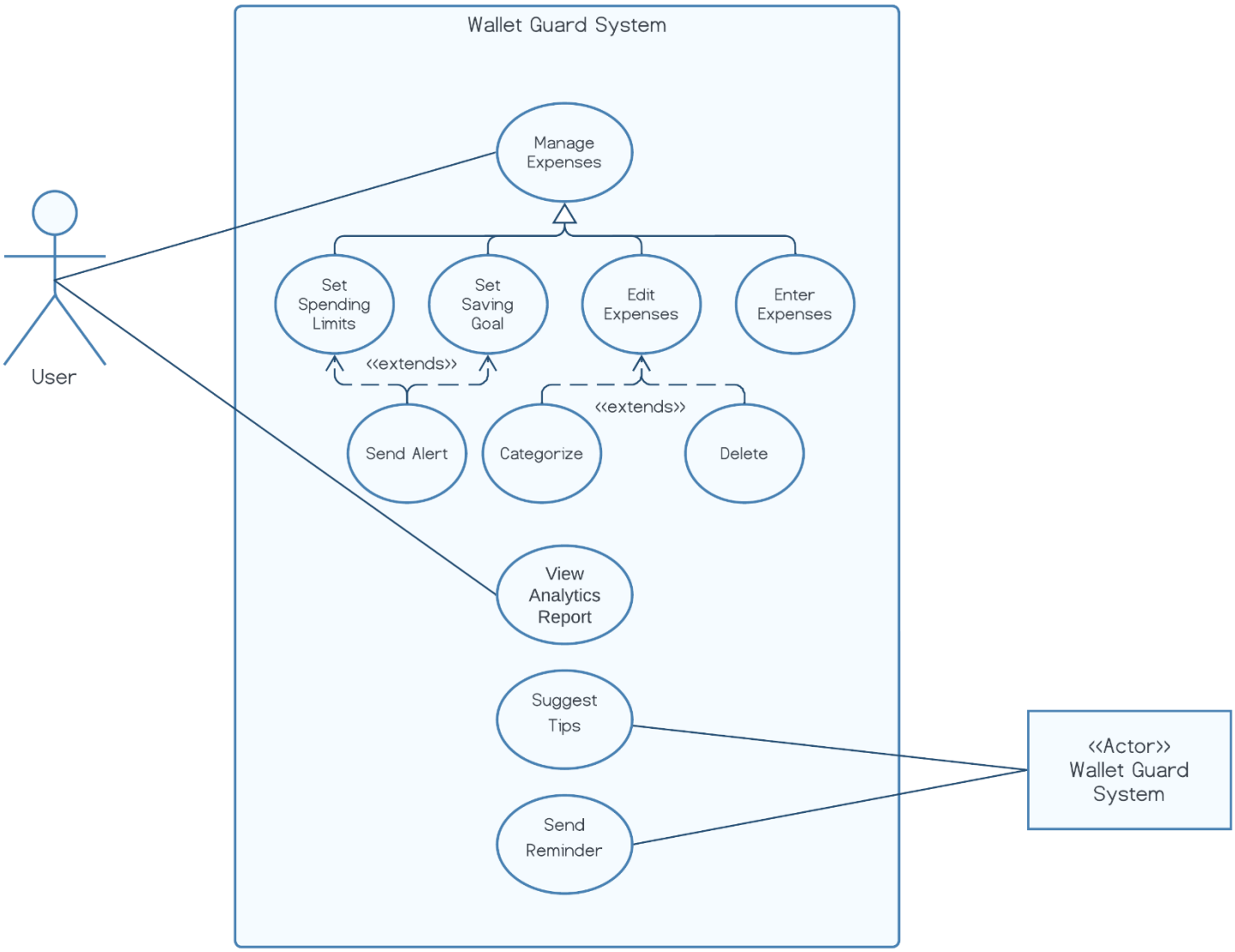
7. **Alerts and Notifications**
   7.1. The system shall send alerts when a user reaches 80% of their overall monthly budget.
   7.2. The system shall send alerts when a user exceeds their overall monthly budget.
   7.3. The system shall send alerts when a user reaches 80% of any category-specific budget.
   7.4. The system shall send alerts when a user exceeds any category-specific budget.
   7.5. The system shall allow users to customize the threshold percentages for alerts.

Non-functional Requirements

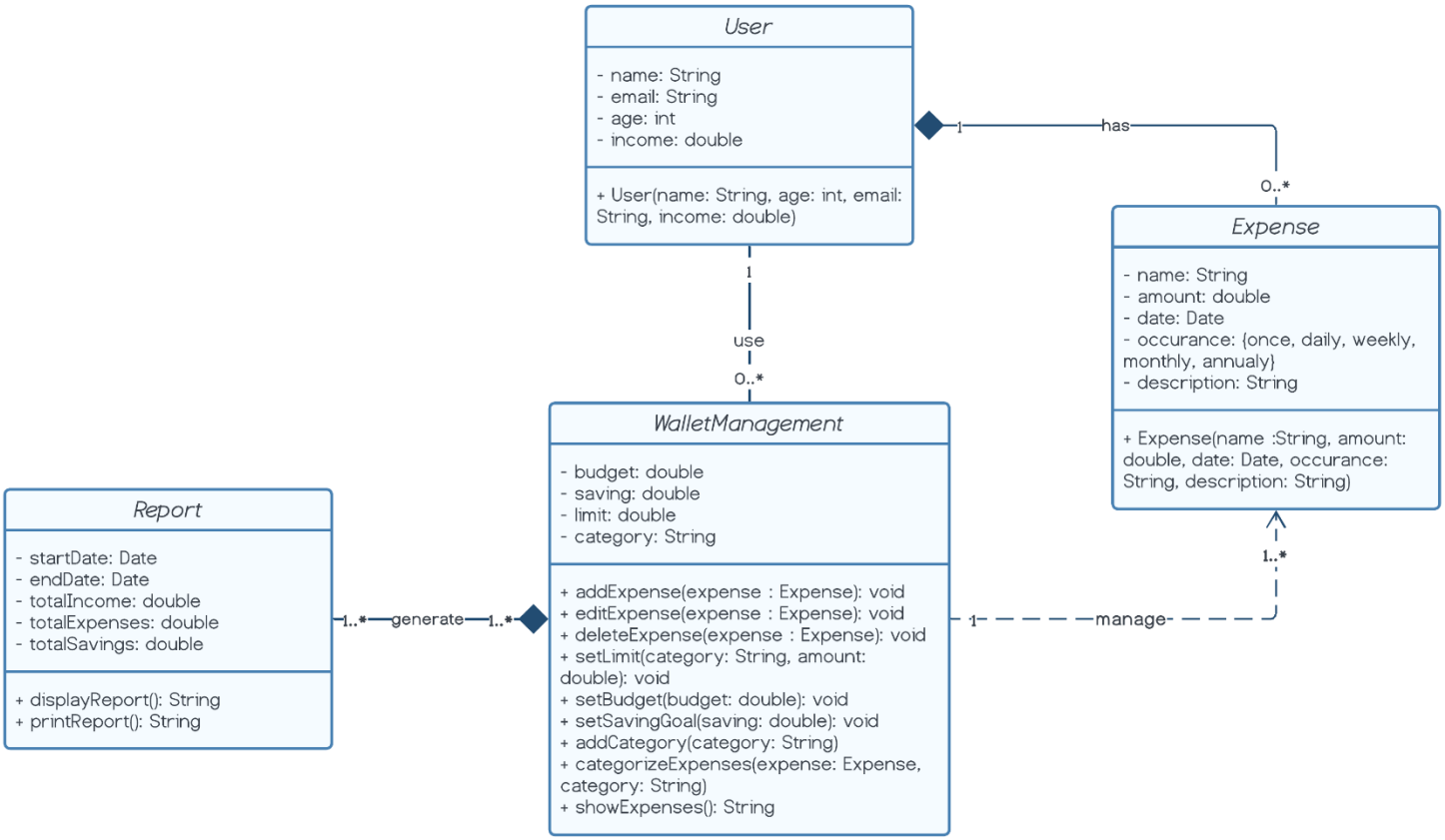
1. The application shall run natively on desktop operating systems without requiring an internet connection or external server dependencies.
2. The system shall be packaged as a self-contained executable JAR file, incorporating all necessary libraries and resources.
3. Users shall be able to run the application directly from the JAR file without requiring a complex installation process.
4. All user data, including wallet information and expenses history, shall be stored locally on the user's computer.
5. The application shall utilize JavaFX to provide a graphical user interface accessible on the user's desktop.
6. The system should have an intuitive and user-friendly interface to ensure ease of use.
7. The system shall operate without requiring user login, allowing immediate access to functionality upon launching the application.
8. All core functionalities of Wallet Guard shall be fully operational without requiring an active internet connection.
9. The application shall be compatible with major desktop operating systems that support Java Runtime Environment (JRE).
10. The system must be able to handles common errors
11. The system must ensure that all user’s inputs are validated to prevent incorrect or incomplete data entry.

System modelling

Use Case Diagram



Class Diagram



Tools

Integrated Development Enviroment (IDE)

Visual Studio Code, Netbeans, and Eclipse



Agile Planning Tool

Jira

Modeling App

Lucidchart

Git Tools

GitBash and Github


Messaging App

Slack