**Software Requirements and Specifications**

Personal Finance Management App

Version <1.0>

Group Name: B7

Chu Chun Yin 1155194319 1155194319@link.cuhk.edu.hk

Ko Kam San 1155194979 1155194979@link.cuhk.edu.hk

Lai Tsz Hin 1155194965 1155194965@link.cuhk.edu.hk

Chan Yiu Cheung 1155193060 1155193060@link.cuhk.edu.hk

Instructor: Dr. Lam Tak Kei

Course: CSCI3100

Department: Computer Science and Engineering

Printing Date: 10/2/2025

Contents

[1 Introduction 3](#_Toc1223647549)

[1.1 Document Purpose 3](#_Toc1552157820)

[1.2 Product Scope 3](#_Toc1510383715)

[1.3 Intended Audience and Document Overview 3](#_Toc1235461114)

[1.4 Definitions, Acronyms and Abbreviations 4](#_Toc1087068628)

[1.5 Document Conventions 4](#_Toc1212642510)

[1.6 References and Acknowledgments 4](#_Toc1006209609)

[2 Overall Description 4](#_Toc1358990256)

[2.1 Product Overview 4](#_Toc497695824)

[2.2 Product Functionality 5](#_Toc1407910115)

[2.3 Design and Implementation Constraints 5](#_Toc2033164604)

[2.4 Assumptions and Dependencies 6](#_Toc1283021877)

[3 Functional Requirements 6](#_Toc1137334902)

[3.1 User Authentication 6](#_Toc1732767039)

[3.2 Expense Tracking 7](#_Toc1107086832)

[3.3 Budget Management 7](#_Toc1514797779)

[3.4 Financial Reports 7](#_Toc302596568)

[3.5 Budget Alerts 7](#_Toc448878375)

[3.6 Data Export 7](#_Toc1180360302)

[3.7 Multi-Platform Support 8](#_Toc2020504651)

[4 Non-functional Requirements 8](#_Toc1110488463)

[4.1 Performance Requirements 8](#_Toc1433128843)

[4.2 Safety and Security Requirements 8](#_Toc1755781272)

[4.3 Software Quality Attributes 8](#_Toc1328693477)

# 1 Introduction

## 1.1 Document Purpose

This document provides comprehensive software requirements and specifications for the Personal Finance Management Web. The version of this web application is 1.0. This document includes detailed requirements such as functional and non-functional requirements, and the description of this web application. This SRS is a clear and detailed guideline for our intended audience such as developers, testers, project managers and the client to understand, design, and implement the app.

## 1.2 Product Scope

The Personal Finance Management Web is a web application which designed to help users track their expenses, set budgets, and monitor their financial health. There are the key features:

* Expense Tracking: users can record, categorize, and manage their daily expenses.
* Budget Management: users can set, modify, and review budgets with real-time alerts to prevent the spending thresholds from approaching predefined limits.
* Data Visualization: user can have better financial insights by generating charts and trend reports.
* Notifications: users will receive alerts via email and SMS.

The benefits of these features are that users can improve their financial awareness and management. Furthermore, it gives users the ability to plan and adjust their spending habits. The last is that it provides data-driven insights via charts and trend reports.

## 1.3 Intended Audience and Document Overview

The intended audience of this document:

* **Developers:** To understand the function and architecture of system.
* **Project Managers:** For planning, scheduling, and resource management.
* **Testers:** To create test cases and validate the functions.
* **Client/Stakeholder:** For review and confirmation of required functionalities and objectives.

The SRS is organized into four parts. The first part is that Introduction provides the purpose, scope, and definitions. The second part is that the Overall Description gives a major part of the product and its constraints. The third part is that the GroupSpecific Requirements detail the interfaces, functionalities, and use cases. The four part is that the remaining sections cover non-functional requirements.

## 1.4 Definitions, Acronyms and Abbreviations

* **API:** Application Programming Interface
* **PDF:** Portable Document Format
* **PFM Web:** Personal Finance Management Web application
* **PR**: Performance Requirements
* **SQA**: Software Quality Attributes
* **SRS:** Software Requirements Specification
* **SSR**: Safety and Security Requirements

## 1.5 Document Conventions

* **Formatting:** This document is written in Times New Roman, size 12 for body text. Section headers are bolded and numbered according to IEEE standards.
* **Numbering:** Non-Functional Requirements are numbered sequentially (e.g., PR1, SSR1).
* **Terminology:** Terms are defined in Section 1.4 for consistency.

## 1.6 References and Acknowledgments

* IEEE Std 830-1998 – IEEE Recommended Practice for Software Requirements Specifications.
* GMU-SRS-template.pdf – Provided project template.
* Online resources and industry guidelines for personal finance management web applications.

# 2 Overall Description

## 2.1 Product Overview

The web application is a new, self-contained product designed to help users manage their finances effectively. The app will provide users with tools to track income, expenses, savings, and investments, offering a comprehensive view of their financial health. The product is aimed at individuals who want to take control of their finances, whether they are budgeting for personal expenses, saving for future goals, or managing investments.

The web application will be accessible via web browsers and will interact with external financial institutions through APIs to fetch transaction data. It will also integrate with third-party services for features like credit score monitoring and financial advice. The system will be designed to ensure data security and privacy, adhering to industry standards for encryption and user authentication.

## 2.2 Product Functionality

The major functions of the application include:

* User Authentication
  + Secure authentication with encryption.
  + Multi-factor authentication including third-party SSOs.
* Financial Tracking
  + Record real-time real-life transactions.
  + Categorize transactions into sensible groups.
* Budget Management
  + Set up budget goal and financial plan for the future.
* Reporting and Analytics
  + Generate financial reports.
  + Visualize spending patterns with charts and graphs.
  + Provide insights and recommendations.
* Budget Alerts
  + Notify users when limits are reached.
  + Incorporate budget goal setup.
* Data Export
  + Export all personal financial data into a compatible format.

## 2.3 Design and Implementation Constraints

The development of the Personal Finance Management Web App will be subject to the following constraints:

Hardware Limitations

* The app must be responsive and performant on both desktop and mobile devices.
* Ensure efficient memory usage and fast load times.

Technologies and Tools

* Use of the COMET method for software design.
* UML modeling language for system design and documentation.
* Front-end development using React.js.
* Back-end development using Node.js and Express.
* Database management using PostgreSQL.

Security Considerations

* Implement strong encryption for data at rest and in transit.
* Ensure compliance with GDPR and other relevant data protection regulations.

Design Conventions and Programming Standards

* Follow RESTful API design principles.
* Adhere to coding standards and best practices for JavaScript and SQL.

## 2.4 Assumptions and Dependencies

The project is based on the following assumptions and dependencies:

Assumptions:

* Users will have access to a stable internet connection.
* External APIs for financial data will be available and reliable.
* Users will provide accurate and up-to-date financial information.
* The app will be used primarily by individuals, not businesses.

Dependencies:

* Availability of third-party services for credit score monitoring and financial advice.
* Use of open-source libraries and frameworks for development.
* Reliance on cloud services for hosting and data storage.

These assumptions and dependencies are critical to the design and functionality of the app. Any changes or issues with these factors could significantly impact the project's success.

# 3 Functional Requirements

## 3.1 User Authentication

Requirements: Users are able to create a personal account by entering their personal information, such as email and phone number for authentication in order to log in securely. Also, it supports login social media (Google, Facebook, Apple ID).

Specifications: Registration with social login via OAuth 2.0. Also, session management can use JSON Web Tokens (JWT) .

## 3.2 Expense Tracking

Requirements: Users are able to add, edit, and delete personal expenses they have input. Those expenses can be categorized based on food, transportation, utilities. Thus, users can track their expenses easily.

Specifications: Add the amount, category, date etc. Edit can modify the previous record. Deleting can permanently remove the record the user input for the expenses.

## 3.3 Budget Management

Requirements: Users are able to set a budget for each categorized expense weekly or monthly. The app will keep track of the expenses to see whether it is below the budget set.

Specifications: The set budget must be a positive number. Then the app will calculate the amount of expenses and display a visual progress bar for each category to show the result.

## 3.4 Financial Reports

Requirements: The app can demonstrate financial reports to users by using pie charts, plotted graphs, tables etc. Also, it provides insight into expenses based on the reports to remind users of increasing their financial health.

Specifications: The app will generate a report every period using graphs or charts. The insight will give the comment of the summary of the expenses this period.

## 3.5 Budget Alerts

Requirements: The app will send notifications to users when the budget management system detects the expenses are close to or exceed users’ budget. Users can choose the frequency of alerts such as weekly or monthly.

Specifications: Users set a threshold and receive notifications for the mobile app when exceed the threshold using Firebase Cloud Messaging (FCM). Users can reset their budget for the next month.

## 3.6 Data Export

Requirements: Users are able to export financial data or financial reports into CSV or PDF format and store into their device.

Specifications: Users can select specific date ranges, categories, or types of transactions to include in the export. Also, the exported files will follow the naming convention: Financial\_Report\_DD/MM/YY.

## 3.7 Multi-Platform Support

Requirements: The app will be available on mobile (iOS, Android) and web platforms.

Specifications: Both iOS and Android can be built using React Native, and the web are using React.js . For UIUX, HTML CSS, JS or Python would be used.

# 4 Non-functional Requirements

## 4.1 Performance Requirements

PR1: The system shall respond to user inputs within 3 seconds under normal load

PR2: The data synchronization between users’ devices and database shall occurr within 5 seconds

PF3: The user authentication should be finished within 3 seconds

PF4: The dashboard charts and financial reports shall render witthin 5 seconds

## 4.2 Safety and Security Requirements

SSR1: HTTPS shall be used to secure all data transmissions

SSR2: All sensitive data including financial records and user accounts must be encrypted in transit and at rest

SSR3: System should be automatically logged out after a period of inactivity

SSR4: Implementing regular application maintenance and testing for security, loopholes identification and fixing

## 4.3 Software Quality Attributes

SQA1: The user interface must be intuitive, with clear labels, icons and instructions

SQA2: Providing clear and timely feedback to the users based on their status and action

SQA3: The web application must be responsive and adapt for different devices and screen sizes in order to provide a consistent user experience

SQA4: Implementing regular backup mechanisms to prevent data loss and corruption

SQA5: Implementing error handling system to provide error message to users and prevent system crashes

SQA6: The architecture should be modular, allowing easy updates, integration of additional features and replacement without affecting the system

SQA7: Conducting code reviews to ensure code readability and maintainability

SQA8: The system can handle large volumes of data without impacting performance