System Analysis and Design

Final Project

2024

Presented by:

Group BG

1. Executive Summary

Universities financial software is primarily designed to manage finances for students in institutions. It has features that help students track income and expenses, create financial statements, and provide visibility on charts. The main goal of this software is to provide students with a simple tool that makes budgeting easier.

2. Introduction

2.1 Project Background

Financial management is an important skill, especially for university students who often face challenges balancing expenses with limited income. The proposed software seeks to provide needed solutions, facilitating financial management and planning.

2.2 Objectives:

Develop user-friendly financial management software.

Enable students to track income and expenses.

Provide data visualization through charts.

Set up reminders for financial tasks.

2.3 Scope:

This project covers the full software development lifecycle, from initial concept through to deployment. It also includes user documentation and support.

3. System Requirement Specification (SRS)

1. Introduction

1.1 Purpose:

The financial management software is designed specifically for university students to help them manage their personal finances. The software allows users to track their income and expenses, set financial reminders, and visualize their financial data through charts. The primary goal is to provide students with an intuitive tool that simplifies financial management.

1.2 Scope:

This software includes functionalities such as user registration and login, income and expense tracking, financial reminders, and data visualization via a dashboard and charts. The system is built using C#, Windows Forms, and a MySQL database, making it suitable for offline usage.

1.3 Definitions, Acronyms, and Abbreviations:

- **GUI:** Graphical User Interface
- **CRUD:** Create, Read, Update, Delete
- **SRS:** System Requirement Specification

1.4 References:

- Dotnet-bot. (n.d.). *.NET framework documentation*. Microsoft Learn: Build skills that open doors in your career. https://learn.microsoft.com/en-us/dotnet/framework/
- YouTube. https://youtu.be/OnzpBGsLgiE

2. System Overview

2.1 System Functionality: The software provides the following functionalities:

- **User Registration and Login:** Secure authentication system allowing users to create accounts and log in.
- **Income and Expense Management:** Users can add, Read, update, and delete income and expense entries.
- **Reminders:** Users can set reminders for financial events or deadlines.
- **Dashboard:** Displays an overview of the user’s financial status.
- **Charts:** Visual representation of financial data, including income and expense breakdowns.

2.2 User Characteristics:

The target users are university students with basic computer literacy. They require a simple and intuitive interface to manage their finances efficiently.

2.3 Assumptions and Dependencies:

- The software assumes that users have access to a Windows-based system.
- The software depends on MySQL for data storage and .NET Framework for execution.

3. Functional Requirements

3.1 User Registration and Login:

- User should be able to create an account by providing a username and password.
- Users should be able to log in using their credentials.

3.2 Income/Expense Management:

- User should be able to add income entries with details like amount, source, and date.
- User should be able to add expense entries with details like amount, category, and date.
- User should be able to edit or delete existing income/expense entries.

3.3 Reminder System:

- User should be able to set reminders for specific financial tasks or deadlines.
- Reminders can be edited or deleted as needed.

3.4 Dashboard and Charts:

- User should be able to generate charts according to their expenses and incomes.
- User should be able to view income vs. expenses over a selected period.

3.5 Logout Functionality:

- User should be able to securely log out of the application.

4. Non-Functional Requirements

4.1 Performance:

- The software should load the dashboard and charts within 2 seconds.
- All database queries should be executed within one second.

4.2 Usability:

- The interface should be intuitive.

4.3 Reliability:

- The software must be capable of running offline and store data reliably without corruption.

4.4 Security:

- All sensitive data, such as passwords, must be encrypted.
- User sessions should time out after a period of inactivity to ensure security.

5. System Architecture

5.1 Overview of Architecture:

- The system follows a client-server architecture, with the Windows Forms application as the client and MySQL as the backend database.

5.2 Database Design:

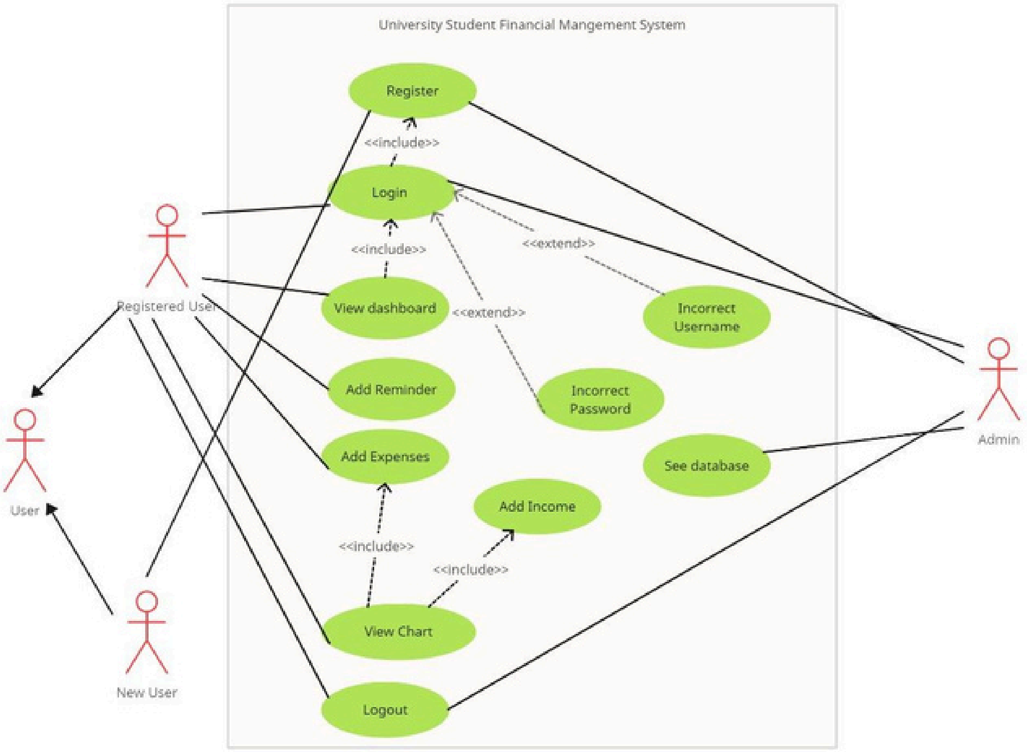
- **Users Table:** Stores user details (e.g., username, email, hashed password).
- **Income Table:** Stores income entries linked to users.
- **Expenses Table:** Stores expense entries linked to users.
- **Reminders Table:** Stores reminders set by users.

5.3 Technology Stack:

- **Frontend:** C# (Windows Forms)
- **Backend:** MySQL
- **Security:** BCrypt for password hashing

6. System Models

6.1 Use Case Diagrams:



6.2 User Description:

Name of the use case: view dashboard

Actors: user (primary actors)
System (secondary actors)

Entry conditions:

- User should register to the system.
- User should log in to the system.

Flow of events:

- When the user logs in to the system the user will be able to select / click the dashboard from the navigation bar.
- When the user clicks the dashboard button the system will display the dashboard after fetching the data from the back end.
- User will navigate to the dashboard of the system.

- User can view the dashboard, update the dashboard as preferred.
- The system will update the dashboard if the user makes any changes.

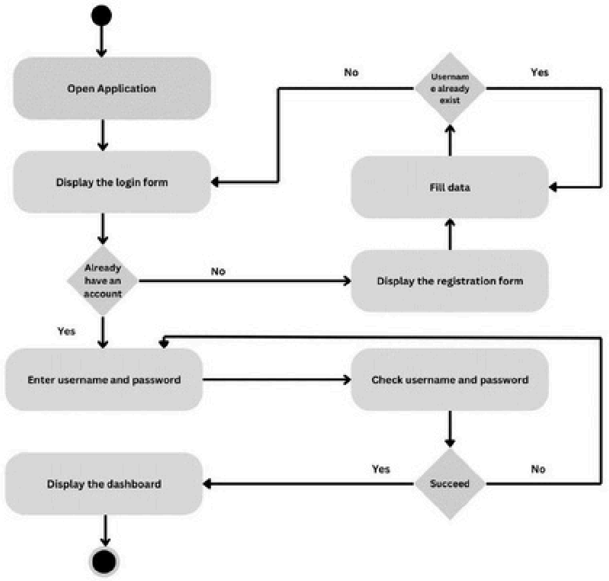
Exit conditions:

- Users view the dashboard.
- Making changes to the dashboard as preferred.

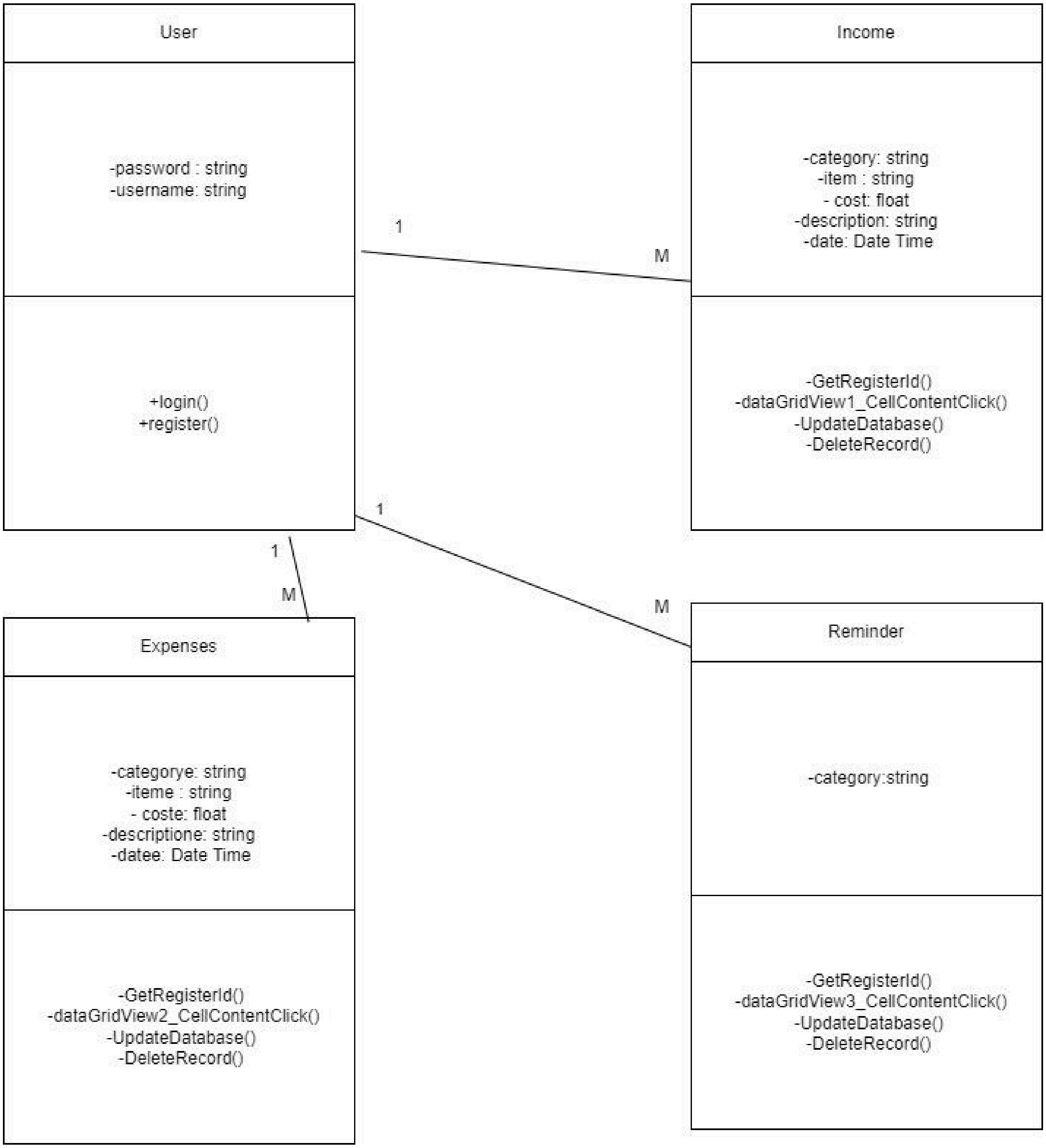
Quality requirements:

The system should respond within 20 seconds.

6.3 UML Activity Diagram (Login and Registration)



6.4 UML Class Diagram



7. Validation and Testing

7.1 Test Cases:

- Example: Test case for successful login with correct credentials.
- Example: Test case for adding a new income entry.

7.2 Validation Criteria:

- Ensure that all functionalities work as expected, and the software meets performance benchmarks.

4. System Design

4.1 System Architecture:

The system follows a client-server architecture:

Client: Windows Forms application (C#)

Server: MySQL database for storing user data, income, expenses, and reminders.

4.2 User Interface Design:

Dashboard: Provides an overview of financial data.

Income/Expenses: Separate pages to manage income and expenses.

Reminders: Interface to set and manage financial reminders.

Charts: A graphical representation of financial trends.

4.3 Database Design:

Users Table: Stores user information, including hashed passwords.

Income Table: Stores income records.

Expenses Table: Stores expense records.

Reminders Table: Stores reminder details.

5. Implementation and Development

5.1 Tools and Technologies:

Programming Language: C#

IDE: Visual Studio

Database: MySQL

Libraries: BCrypt for password hashing

5.2 Development Phases:

Phase 1: Requirements gathering and analysis.

Phase 2: System design (UI/UX and database)

Phase 3: Implementation (coding and development)

Phase 4: Testing (unit, integration, and system testing)

Phase 5: Deployment

6. Testing and Validation

6.1 Test Strategy:

Unit Testing: Each module (e.g., income management, login) was evaluated individually.

Integration Testing: Ensured that different modules work together seamlessly.

System Testing: Evaluated the entire system for performance and usability.

6.2 Validation:

Verified that the software meets all functional and non-functional requirements as specified in the SRS.
Gathered feedback from a small group of university students for further refinement.

7. User Manual

User Manual

1. Introduction

1.1 Purpose:

This user manual provides guidance on how to use the financial management software. The software is designed to help university students track their income, expenses, and set financial reminders.

2. Getting Started

2.1 System Requirements:

- **Operating System:** Windows 10 or later
- **RAM:** 4 GB minimum
- **Storage:** 50 MB free space
- **Additional Software:** .NET Framework 4.7.2 or later

2.2 Installation Guide:

1. Download the installer from the provided link.
2. Run the installer and follow the on-screen instructions.
3. Once installed, launch the application from the start menu.

2.3 Creating an Account:

1. Open the application and click on "Register."
2. Enter your username and password.
3. Click "Signup" to create your account.

3. Navigating the Interface

3.1 Dashboard Overview:

- The dashboard is the first screen you see after logging in. It provides a summary of your total income, total expenses, and current balance (income and expenses) according to some time durations.

3.2 Using the Menu:

- **Income/Expenses:** Navigate to this section to add, read, update, or delete income and expense entries.
- **Reminders:** Access this section to manage your financial reminders.
- **Charts:** View visual representations of your financial data.

4. Managing Finances

4.1 Adding Income/Expenses:

- Go to the "Income/Expenses" section.
- Click on "Add Income" or "Add Expense."
- Fill in the required details and click "Add."
- Fill ID only for update and delete otherwise not necessary.
- To view newly entered data click “Read.”

4.2 Setting Reminders:

- Go to the "Reminders" section.
- Click on "Add Reminder."
- Fill in the required details and click "Add."
- Fill ID only for update and delete otherwise not necessary.
- To view newly entered data click “Read.”

4.3 Viewing Charts:

- Go to the "Charts" section to view a graphical representation of your financial data.

5. Account Management

5.1 Profile Settings:

- Click on your username in the top-right corner to access profile settings.
- Update your profile information and save changes.

5.2 Logging Out:

- Click on the "Logout" button in the top-right corner to securely log out of the application.

6. Troubleshooting

6.1 Common Issues:

- **Show Password:** Use show password to see your password when login or registering.
- **Login Issues:** Ensure your credentials are correct and that the Caps Lock key is not enabled.

6.2 Contact Support:

- If you encounter any issues, contact support at groupbg@gmail.com or 011118282.

7. FAQs

- **How do I reset my password?** Use the "Forgot Password" feature on the login page.
- **How can I export my data?** Currently, exporting data is not supported but will be available in future updates.

8. Glossary

- **Income:** Money received, typically on a regular basis.
- **Expense:** Money spent, or cost incurred.
- **Reminder:** A prompt or notification for upcoming tasks or deadlines.

8. Responsibility Matrix

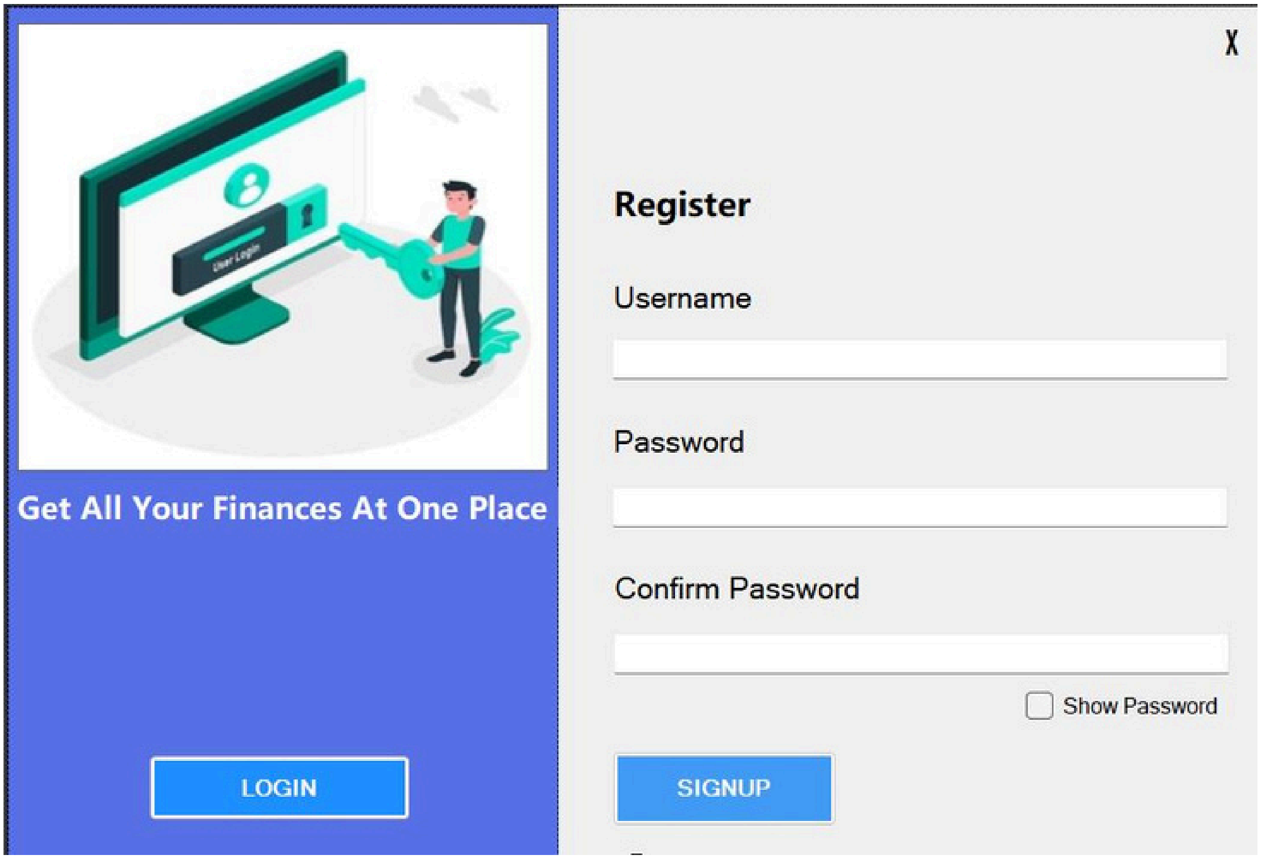
Requirement Gathering	M Pabasara
System Design (UI/UX, Database)	AMKL Thilakarathna
Coding and Development	All Team members
Testing and Validation	AMKL Thilakarathna, S Rukshitha
User Manual Preparation	AGU Sandeepa
Final Documentation and Report	P H S S Siriwardhana, G T sathsarani
Presentation Preparation	BGTN Gamage

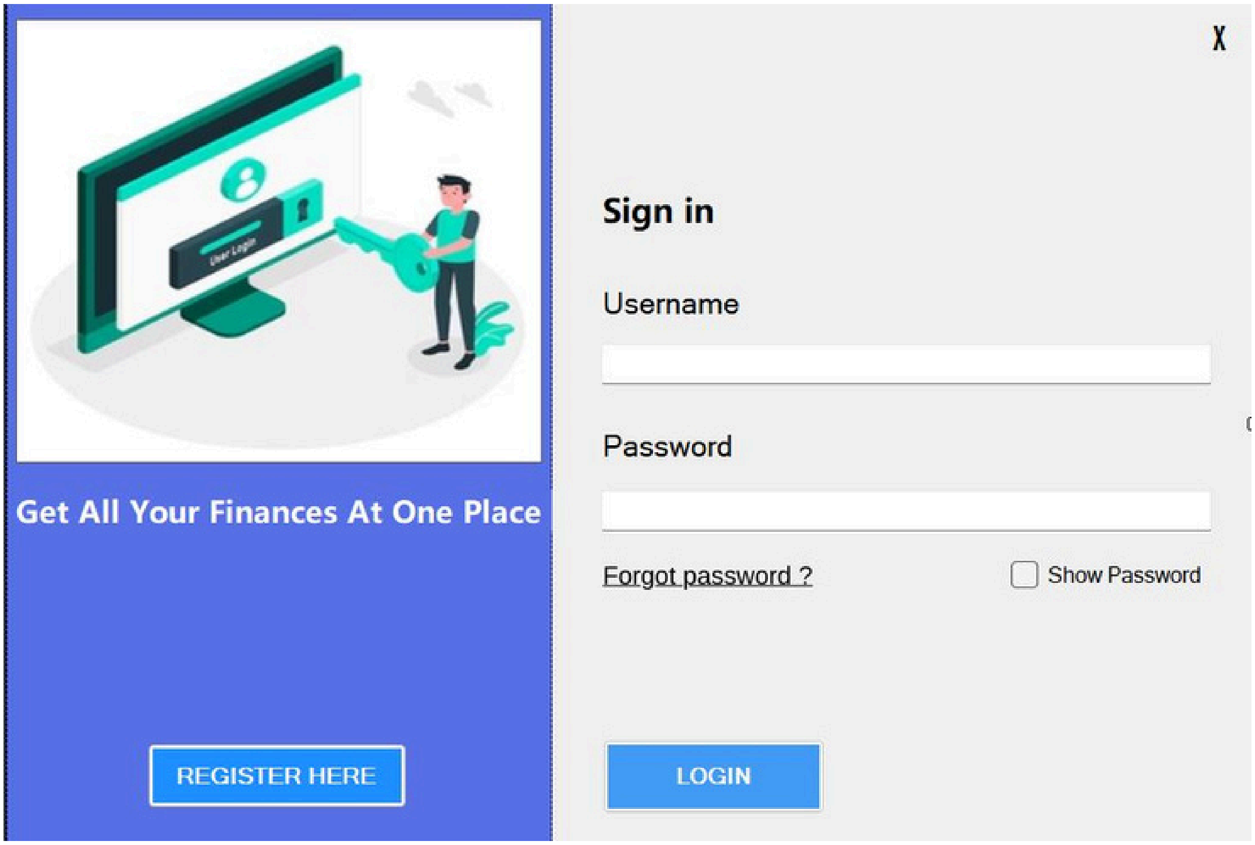
9. Conclusion

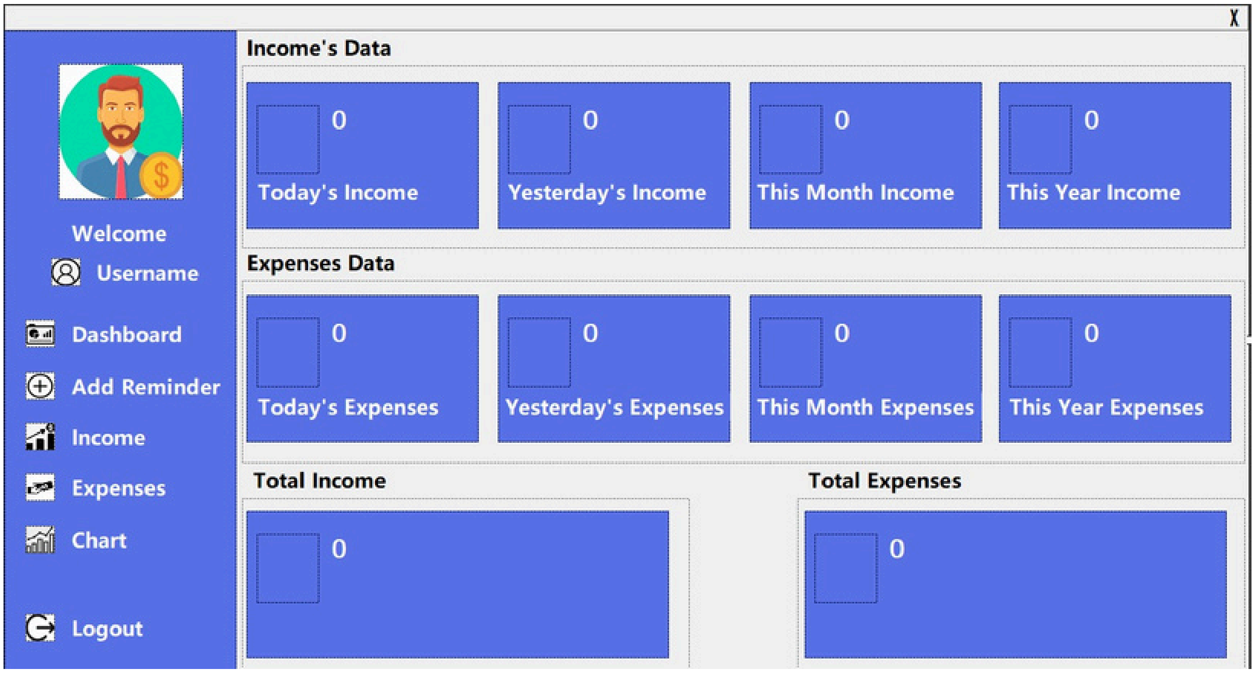
The financial control software undertaking successfully finished its desires of providing a reliable, user-friendly tool for college students. The gadget addresses not unusual economic management demanding situations through intuitive layout, sturdy features, and steady facts dealing with. Moving ahead, there is potential for in addition improvement, including adding features for facts export and greater advanced monetary making plans equipment.

10. Appendices

Appendix A: Screenshots




X
Sign in
Username
Password
Forgot password ?
□ Show Password
LOGIN
Get All Your Finances At One Place
REGISTER HERE
User Login



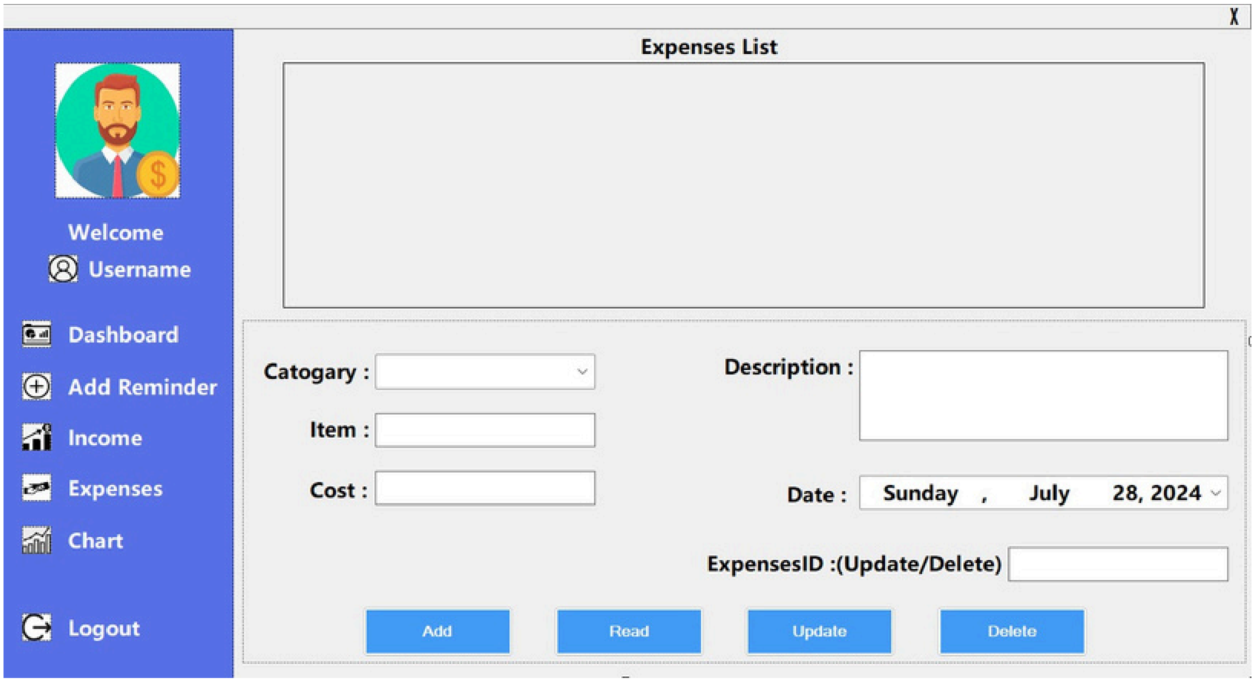
Income's Data

0	0	0	0
Today's Income	Yesterday's Income	This Month Income	This Year Income

Expenses Data

0	0	0	0
Today's Expenses	Yesterday's Expenses	This Month Expenses	This Year Expenses

Total Income	Total Expenses
0	0



Expenses List

Catogary :

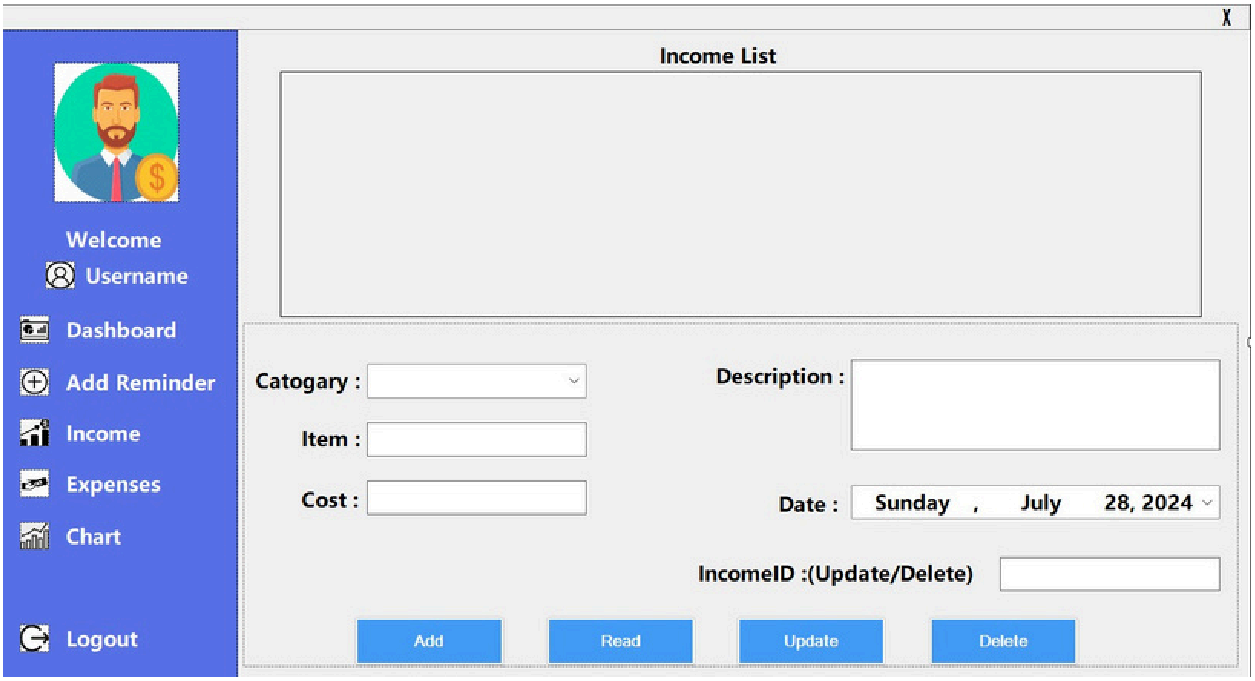
Item :

Cost :

Description :

Date : Sunday , July 28, 2024

ExpensesID :(Update/Delete)



Income List

Catogary :

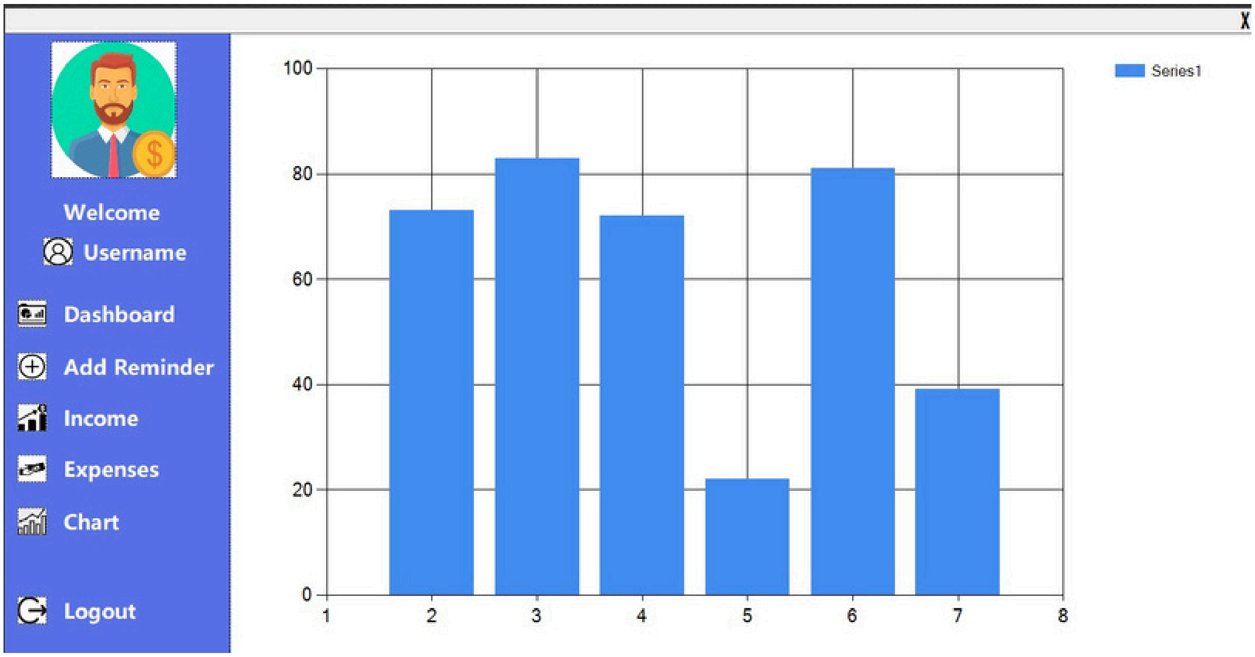
Item :

Cost :

Description :

Date : Sunday , July 28, 2024

IncomeID :(Update/Delete)



Presentation Video Link

Click Here To Watch The Video

Team Members

Index no	Name
31229	AMKL Thilakarathna (Leader)
31870	Rukshitha S
29554	G T sathsarani
31794	Gamage BGTN
31133	Sandeepa AGU
29397	Pabasara M M
29375	P H S S Siriwardhana