Interim Report

Level 02

Expense Tracker System

Team ToDo

Wickramasingha W.U.M.	214228B
Wijesinghe W.D.A.C.	214235T
Senarathna G.G.P.C.	214189E
Priyashan H.M.L.S.	214162P
Karunarathna W.M.W.G.M.S.	214101F

Supervised by: Ms. R.G.C. Upeksha

Faculty of Information Technology

University of Moratuwa

2023

Abstract

This project aims to develop an innovative Expense Tracker System website, offering users a comprehensive and user-friendly platform to manage and monitor their finances effectively. The system will provide a streamlined interface for recording expenses, categorizing transactions, generating insightful reports, and setting budgetary goals. Leveraging modern web technologies, the website will prioritize security, accessibility, and ease of use, empowering individuals and businesses to gain better control over their financial activities. Through intuitive design and robust functionalities, this Expense Tracker System seeks to revolutionize personal and organizational expense management, fostering financial awareness and responsible spending.

Considering those requirements, we are designing a system that facilitates all the requirements as a Progressive Web Application (PWA). Users who want to track their financial circumstances.

To provide a better application we analyze our customer requirements and design the unified model language diagrams before implementation. For developing our system, we chose technologies after a better research program. ReactJS for the frontend, ASP.NET for the backend, and MS SQL for the data management system of our system.

Contents

Chapter 1 Introduction 8
1.1 Introduction 8
1.2 Background and Motivation 8
1.3 Aim and Objectives 9
Aim: 9
Objectives: 9
1.4 Problem in brief 10
1.5 Summary 10
Chapter 2 Existing Solutions 11
2.1 Introduction 11
2.2 Similar Products 11
2.2.1 Monefy 11
2.2.2 Expensify 12
2.2.3 mint 13
2.3 Summary 14
Chapter 3 Technologies Adapted 15
3.1 Introduction 15
3.2 Front-end technology 15
3.2.1 React JS 15
3.3 Back-end technology 16
3.3.1 MS .net framework 16
3.4 Database Technology 16
3.4.1 Microsoft SQL server 17
Chapter 4 Proposed Solution 18
4.1 Introduction 18

4.2 Software Model 18
4.3 Users, Activities, inputs, and outputs 21
4.4 Process 23
4.5 Our Approach 24
4.5.1 User's Data Privacy System 24
4.5.2 UI and UX Desing 24
4.5.3 User Managing System 25
4.6 Project Management 25
4.7 Summary 26
Chapter 5 Analysis and Design 27
5.1 Class Diagram 27
5.2 Use case Diagram 28
5.3 Activity Diagrams 30
5.4 Sequence Diagrams 41
5.5 ER Diagram 54
Chapter 6 Implementation 55
6.1 Login Page 55
6.2 Administrator's Dashboard 56
6.3 User's Dashboard 57
6.4 Transaction 58
6.5 View Transaction 59
6.6 Add Transaction 60
6.7 Create Saving 61
6.8 Analysis 62
6.9 Create Budget 63
6.10 Create Reminder 64

6.11 Send Alert 65
6.12 Add card 66
6.13 View Report 67
6.14 Setting 68
6.15 Create Before Account Overview 69
Chapter 7 Discussion 70
7.1 Introduction 70
7.2 Evaluation and Testing 70
7.3 How our solution differs from similar works done by others 70
7.4 Further Work 70
7.5. Summary 71
Chapter 8 Reference 72
Appendix A - Individual Contribution 73
Appendix B – Action Plan 77

List Of Figures

Figure 1:Monefy 12
Figure 2:Expensify 13
Figure 3:mint 13
Figure 4:Class Diagram 27
Figure 5:Use Case Diagram(a) 28
Figure 6:Use Case Diagram(b) 29
Figure 7:Activity Diagram(Login) 30
Figure 8:Activity Diagram(Add Income) 30
Figure 9:Activity Diagram(Search Records) 31
Figure 10:Activity Diagram(Create Budget) 32
Figure 11:Activity Diagram(Create Remainder) 33
Figure 12:Activity Diagram(Search User) 33
Figure 13:Activity Diagram(Create User) 34
Figure 14:Activity Diagram(Upload Receipt) 34
Figure 15:Activity Diagram(Download Report) 35
Figure 16:Activity Diagram(Send Alert) 36
Figure 17:Activity Diagram(View Transaction) 37
Figure 18:Activity Diagram(Create Saving) 37
Figure 19:Activity Diagram(Manage User) 38
Figure 20:Activity Diagram(Add Expenses) 39
Figure 21:Activity Diagram(View Report) 40
Figure 22:Sequence Diagram(Add Expenses) 41
Figure 23:Sequence Diagram(Add Income) 42
Figure 24:Sequence Diagram(Send Alert) 43
Figure 25:Sequence Diagram(Upload Receipt) 43
Figure 26:Sequence Diagram(Create Saving) 44
Figure 27:Sequence Diagram(Download Report) 45
Figure 28:Sequence Diagram(Manage User) 46

Figure 29:Sequence Diagram(Create Budget) 47
Figure 30:Sequence Diagram(Search Records) 48
Figure 31:Sequence Diagram(Login) 49
Figure 32:Sequence Diagram(Create User) 50
Figure 33:Sequence Diagram(Create Remainder) 51
Figure 34:Sequence Diagram(Search User) 52
Figure 35:Sequence Diagram(View Transaction) 52
Figure 36:Sequence Diagram(View Report) 53
Figure 37:ER Diagram 54
Figure 38:Interface(User Login) 55
Figure 39:Interface(Administrator's Dashboard) 56
Figure 40:Interface(User Dashboard) 57
Figure 41:Interface(Transactions) 58
Figure 42:Interface(View Transactions) 59
Figure 43:Interface(Add Transaction) 60
Figure 44:Interface(Create Saving) 61
Figure 45:Interface(Analysis) 62
Figure 46:Interface(Create Budget) 63
Figure 47:Interface(Create Remainder) 64
Figure 48:Interface(Send Alert) 65
Figure 49:Interface(Add Bank Card) 66
Figure 50:Interface(View Report) 67
Figure 51:Interface(Setting) 68
Figure 52:Interface(Home) 69

Chapter 1 Introduction

1.1 Introduction

Expense tracking is a crucial part of financial management, helping you understand your spending habits and make informed decisions. Common methods for tracking expenses include manual methods like pen and paper or spreadsheet software, as well as digital solutions such as mobile apps, online platforms, and dedicated software. Key elements to record include the date, description, category, amount, payment method, and receipts.

By consistently tracking your expenses, you can gain insights into where your money goes, allowing you to create a budget, cut unnecessary expenses, and work towards your financial goals. Whether you prefer traditional methods or modern digital tools, finding a tracking method that suits your lifestyle is essential for effective financial management. More than just a tool for record-keeping, this system aspires to become an indispensable companion in fostering financial awareness, encouraging responsible spending, and facilitating informed decision-making. Through meticulous attention to detail and a commitment to user needs, this Expense Tracker System project endeavors to redefine the standards of financial management, setting a new paradigm for efficiency, accessibility, and financial well-being.

1.2 Background and Motivation

Expense tracking is vital for financial management. However, it can be challenging due to various factors. Manual tracking with pen and paper or spreadsheets often results in incomplete records and forgotten expenses. Bank statements offer convenience but lack transaction details and categorization. Additionally, the increasing use of mobile apps raises privacy concerns, as these apps may access sensitive financial information.

The genesis of the Expense Tracking System emerged from a critical observation of the challenges individuals and organizations face in managing their financial records effectively. In a fast-paced, digitally-driven world, the complexity of financial transactions often leads to scattered records, hindering a clear overview of expenditure and revenue streams. The motivation stemmed from a desire to alleviate the burden of manual record-keeping, offer a centralized platform for financial data, and provide users with actionable insights to make informed financial decisions. Moreover, considering the dynamic nature of today's economy, the need for a robust, user-friendly system to monitor expenses, set budgets, and track financial goals became apparent. This system aims not only to streamline financial management but also to empower users with the knowledge and tools to navigate their financial landscapes confidently. The goal is to revolutionize how individuals and businesses interact with their finances, promoting financial stability, and fostering a culture of informed financial management.

Furthermore, some individuals find it easier to overspend when using digital tracking methods. Balancing convenience with accuracy and privacy is essential for effective expense tracking.

1.3 Aim and Objectives

Aim:

To develop an Expense Tracking System to track users' financial performance and get a proper idea about their financial requirements.

Objectives:

- To make user authentication and authorization for all the users.
- Design the appropriate system including all user requirements.
- Keep records of users' expenses and incomes.
- To get their financial reports.
- Create savings and budget plans.
- Develop the system for solving the problem.

- Evaluation of the proposed solution.
- Preparation of final documentation.

1.4 Problem in brief

The problem with expense tracking lies in incomplete data, forgotten transactions, and inconsistent categorization. Manual methods are time-consuming while relying solely on bank statements lacks detail. Privacy concerns and the risk of overspending through mobile apps further complicate the process.

1.5 Summary

Through Chapter 1 of this report, we provide a basic introduction to the Expense Tracker System that we expect to develop. It outlines the system's purpose in revolutionizing expense monitoring, budgeting, and financial decision-making for both individuals and organizations. In the next chapter, similar projects for this project will be mentioned and discussed the differences between those projects and ours.

Chapter 2 Existing Solutions

2.1 Introduction

Chapter 2 focuses on documenting the different approaches to addressing the same problem which this application selected to offer a solution for. In the modern world, the Expense Tracker System is useful for individuals and organizations. Furthermore, there are many applications dedicated to Expense Tracker System in the world too. This chapter describes related products in the market same as the proposed solution.

2.2 Similar Products

The software listed below can be identified as similar solutions that directly focus on some of the areas of our problem to be solved.

2.2.1 Monefy

Monefy is an expense tracking system application used mostly by individuals. It is a personal finance application that makes money management easy. The app is designed to streamline expense tracking and help you save money. It can create categories and start tracking essential expenses.

But Monefy can’t be used for advanced financial analysis. It has some syncing issues , security concerns, and limited integrations.

Figure 1:Monefy

2.2.2 Expensify

Expensify is a popular expense management software designed to streamline the process of expense reporting, tracking, and reimbursement for individuals, small businesses, and larger enterprises. Here are some key features and descriptions of Expensify,

- Expense Tracking
- Receipt Capture
- Automated Reporting
- Advanced Features for Businesses

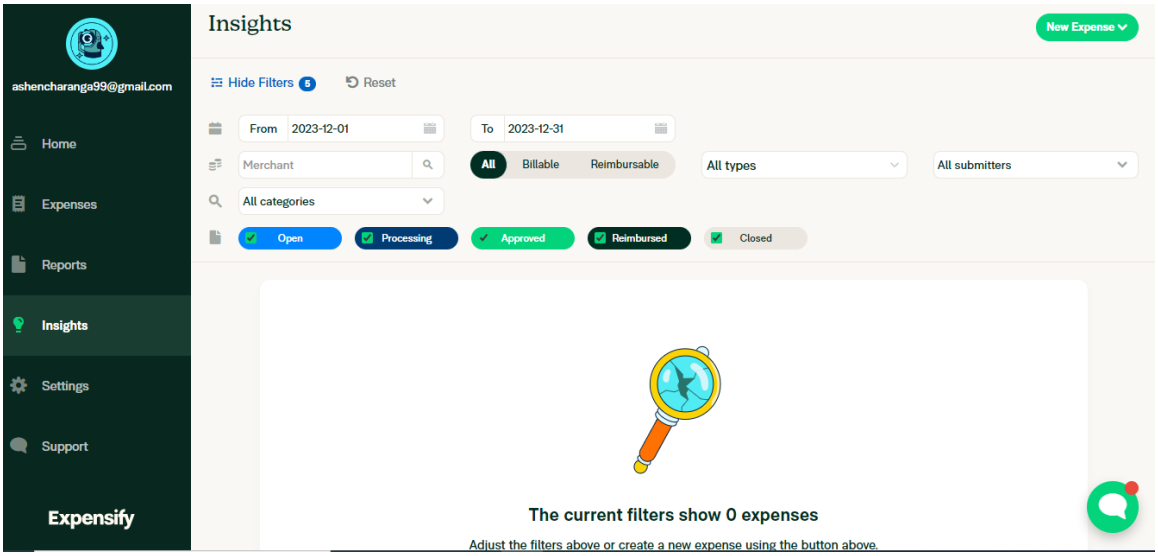

Figure 2:Expensify

2.2.3 mint

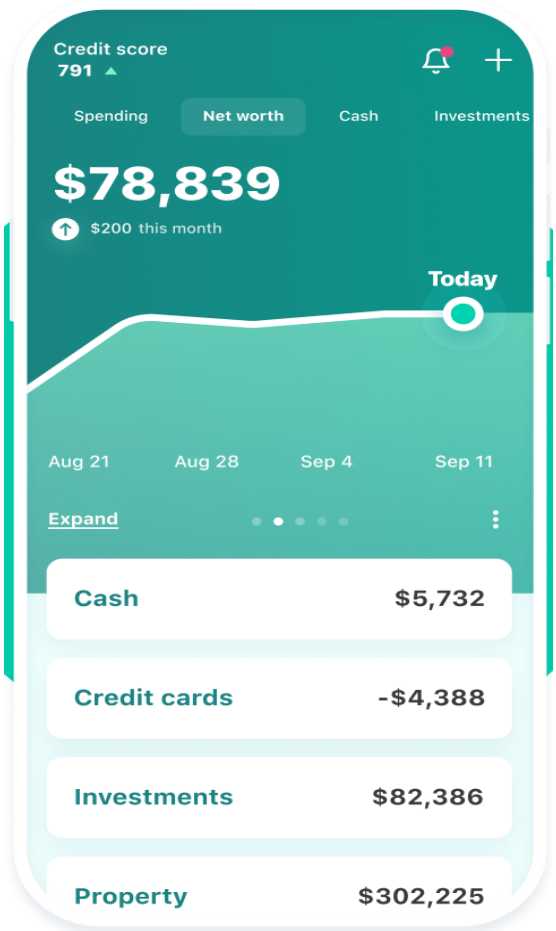

Figure 3:mint

Mint is a widely used personal finance management website that offers comprehensive tools for expense tracking, budgeting, and financial planning. It has some key features like Real-Time Expense Tracking, Budgeting Tools, and Bill Payment Reminders. While Mint offers numerous advantages in expense tracking and financial management, there are some limitations or drawbacks users might encounter that are security concerns, lack of investment tracking, etc.

2.3 Summary

So, according to the above solutions, there are similar features and drawbacks in each tracking system. Therefore, we are considering our system demonstrates the following unique features,

- Very simple user interface.
- Report generation and result dashboard.
- Create Savings and budget plans.
- Create upcoming payment reminders.

It gives an insight into the customer needs and preferences.

Chapter 3 Technologies Adapted

3.1 Introduction

To build the proposed solution, various technologies are used. They can be divided into the following categories.

- Front-end technology
    - React JS
- Back-end technology
    - MS .net framework
- Database technology
    - MS SQL Server

3.2 Front-end technology

The front-end is the interfaces within the proposed system. And this is the point at which end users engage with the system. The system needs to provide a better user experience through interfaces from the viewpoint of the end user. Therefore, we have selected React JS to develop the front-end.

3.2.1 React JS

React is a popular JavaScript library for building user interfaces, particularly web-based applications. It is often used for front-end development because it allows developers to create reusable components that can be rendered on the page, making it easier to build and maintain complex user interfaces. Since the proposed system is CRM software, there are so many individual sections that should be attached together and reused. The functionality of React may be a huge advantage while the development of front-end.

One of the key benefits of React is that it uses a virtual DOM (Document Object Model). Therefore, when data in a React application changes, the virtual DOM will determine the minimal set of changes that need to be made to the actual DOM. This could lead the application to improve the performance.

React also has a strong developer community, with a wealth of resources and libraries available to help developers build and maintain their applications. Since the team

members do not have a lot of experience of building front ends and using react for development, this can be a main benefit for the team.

Due to the efficiency, flexibility, and strong developer community of React, it is easier to build the front-end of the proposed system.

3.3 Back-end technology

All the functionalities of the proposed system are within the back-end development. And it is the core of the application. Therefore, back-end technology is selected carefully. That is why Microsoft .net Framework is selected to build the back end of the system.

3.3.1 MS .net framework

The .NET framework is a popular choice for back-end development because it provides a powerful platform for building and running web-based applications and services. Also, it is flexible and scalable, allowing developers to build applications for a wide range of environments and platforms.

One of the key benefits of the .NET framework is that it provides a large set of libraries and APIs that developers can use to access a variety of different services and resources. This includes support for networking, data access, security, and many other common backend tasks.

The .NET framework also includes several tools and technologies that make it easier for developers to build and maintain applications, including a powerful runtime environment, a comprehensive set of development tools, and a large and active developer community.

3.4 Database Technology

The database can be identified as a major entity of the system. A database is typically designed to store and access data. A well-designed database is crucial to the system since the database stores all the relevant insights regarding the system. To develop such a database, the team has decided to use Microsoft SQL server as the technology.

3.4.1 Microsoft SQL server

SQL server has been introduced and maintained by Microsoft so far. Because of its popularity and many other facts, it is widely used as a Database Management System (DBMS) by many developers. Also, it is compatible with .net framework too.

Some of the features of SQL server are,

Reliability: Reliability and uptime of SQL server is making it an excellent choice for mission-critical applications that require a high level of availability.

Performance: SQL Server is designed to handle large volumes of data and transactions, and it is optimized for fast and better performance.

Security: There are many security features to protect data from unauthorized access, such as encryption, authentication, and access controls within SQL servers.

Scalability: This allows us to handle increasing amounts of data and workloads as the needs of the application grow.

Management: Handy features like backup and recovery, monitoring, and performance tuning can be used within SQL server

Chapter 4 Proposed Solution

4.1 Introduction

Our system, Expense Tracker software, will be used to manage/track incomes and expenses. Through the software, we try to provide an interface for the user to manage with the administration and through that make it easy for the user to manage their incomes and expenses efficiently.

Here the customer can track their income and expenses of their life and rate the income and expenses and through that, the user can analyze their income and expenses and easily manage their life.

In addition, the user can easily create a budget for specific time in their life and can easily track expenses. In addition, the user can easily create a saving and link to their bank account. In addition to that, this system will send payment reminders to the user. Here we have provided the facility to take the remainder too and then the system will track the payments remainder.

4.2 Software Model

We are using the Agile Scrum model to develop the system. Since the user requirements have a possibility to change and we would have a ready to ship model after every scrum, we have a good chance to get the user's idea about the project and deliver what the customer expects well.

Also on the other hand, we will be working with the industry, and we will be able to get hands on experience of how Agile Scrum method will be used in the industry which would be useful for our future career, since agile scrum is one of the widely used software models in modern tech world.

Agile methodologies, including Scrum, are known for their flexibility, iterative approach, and customer collaboration. Here are some advantages of applying Agile Scrum to the development of a money tracker system:

- Flexibility and Adaptability:
    - Agile methodologies, including Scrum, are designed to be flexible and adaptable to changing requirements. In the context of a money tracker system, financial tracking needs may evolve, and Agile allows the development team to respond quickly to these changes.

- Incremental Development:
    - Agile Scrum emphasizes incremental development through short iterations called sprints. This allows for delivering a working product at the end of each sprint. In the case of a money tracker system, users can start using and benefiting from the system sooner, even if not all features are fully implemented.

- User Involvement:
    - Agile Scrum encourages frequent and direct collaboration with end-users. This ensures that the development team stays aligned with user needs and priorities. In the context of a money tracker system, user feedback can be incorporated into the product quickly, improving its usability and functionality.

- Continuous Improvement:
    - Agile promotes continuous improvement through regular retrospectives at the end of each sprint. The team reflects on what went well and what can be improved, fostering a culture of learning and refinement. This is valuable for enhancing the money tracker system based on user feedback and changing requirements.

- Early and Predictable Delivery:

- Agile Scrum promotes the delivery of a potentially shippable product at the end of each sprint. This allows stakeholders to see progress regularly and ensures that the most valuable features are delivered first. For a money tracker system, users can benefit from the system's functionality sooner.

- Risk Mitigation
  - Agile Scrum allows for early identification of issues and risks. Regular reviews and retrospectives provide opportunities to address challenges promptly. This is particularly important for a money tracker system where accuracy and security are critical, and any issues need to be addressed swiftly.

- Transparency:
  - Agile Scrum promotes transparency through various ceremonies, such as sprint planning, daily stand-ups, and sprint reviews. This transparency is crucial for a expense tracker system, where users need confidence in the accuracy and security of financial data.

- Customer Satisfaction:
  - By involving customers in the development process and delivering increments of functionality regularly, Agile Scrum aims to enhance customer satisfaction. For a money tracker system, this means that users are more likely to get a product that meets their needs and expectations.

The Agile Scrum SDLC model can bring numerous benefits to the development of a money tracker system, ensuring flexibility, user involvement, and the early delivery of valuable features.

4.3 Users, Activities, inputs, and outputs

Users		
End User	Activities	• Login using username and password. • Add transactions (incomes / expenses). • Take description for transactions. • Attach receipt for expenses (optional). • View transactions (incomes / expenses). • Filter transactions. • View report. • Download report. • Create saving. • Create budget. • Create remainder. • Search records. • Choose time-period.
	Inputs	• Username and Password. • Incomes. • Expenses. • Saving amount. • Budget amount. • Input date/description for remainder.
	Outputs	• User profile. • Added transaction (alert). • Payment remainder (alert).

		• Created saving (alert). • Created budget (alert). • View report.
Administrator	Activities	• Loing using username and password. • Create user. • View transactions. • View report. • Create saving. • Create budget. • Create remainder. • Send alert. • Maintain event. • Maintain user privacy. • Manage users. • Calculate balance. • Search / filter users.
	Inputs	• Username and password. • Unique id for new user and details. • Add details for report. • Add details for saving. • Add details for budget. • Add details for remainder.
	Outputs	• Alert for created user. • Alert for add transaction. • Alert for created saving.

		• Alert for created budget. • Alert for created remainder. • Report.

4.4 Process

Our software will manage the incomes/expenses of user in specific time-period. Firstly, a new customer will be registered to the application by entering the required data. This registration is done by the administration. Then at the same time, a user profile page for the user will be created.

The Expense Tracker System provides a comprehensive and user-friendly solution for effective financial management. Initiated by the administrator, users are created with specific roles and permissions. Upon login, users encounter a well-designed dashboard showcasing an overview of the software's functionalities. The core features include adding transactions, which can be categorized into income and expense, with an option to specify sources and attach receipts. The system prioritizes user privacy, ensuring the secure maintenance of financial records.

Users can view transactions effortlessly, with filtering options and the choice to delve into additional details. The software also enables the creation of savings, allowing users to set targets and include notes for better financial planning. Budget creation is streamlined, empowering users to allocate funds to various expense categories. Reminders and alerts enhance financial discipline, with notifications sent at predefined intervals.

One standout feature is the report generation capability, offering a comprehensive overview of expenses and incomes. Users can filter reports based on specific criteria and even download them for external use. The search functionality facilitates the retrieval of records, and users can choose time periods for a more focused analysis. Additionally, events can be maintained within the system, providing a holistic view of financial activities.

The system's user management is robust, allowing administrators to efficiently control user roles and permissions. Before creating an account, users can explore the software's details. Once an account is created, a detailed dashboard materializes, offering a snapshot of the user's financial landscape. This dashboard serves as a central hub for navigating through the various features, ensuring a seamless and intuitive user experience.

In summary, the Expense Tracker System excels in its ability to address diverse financial needs. From creating accounts and adding transactions to generating detailed reports and setting savings goals, the system promotes financial awareness and accountability. The emphasis on user privacy and efficient user management further elevates the software's utility. Whether tracking expenses, adhering to budgets, or receiving timely reminders, the system caters to users' financial well-being with a user-centric approach.

4.5 Our Approach

The front-end development team, leveraging technologies such as React, focuses on creating a user-friendly interface. This team designs the dashboards and interfaces that users interact with, ensuring a seamless and visually appealing experience.

The back-end development team, utilizing the power of .NET, works on the logic and functionality that drives the system. They implement the core features such as user creation, transaction handling, and data storage using MS SQL. The back-end is the engine that powers the entire system, handling complex operations while maintaining data integrity and security.

4.5.1 User’s Data Privacy System

The security team focuses on user privacy and data protection. They implement encryption protocols, access controls, and security measures to safeguard sensitive financial information. This team works in tandem with other development teams to ensure that the system complies with industry standards and regulations.

4.5.2 UI and UX Desing

The user experience (UX) and user interface (UI) design team collaborates closely with front-end developers to create an intuitive and visually appealing design. They consider

user feedback, industry best practices, and accessibility standards to enhance the overall user experience.

4.5.3 User Managing System

The Users Management System for the Expense Tracker adopts a multifaceted approach to ensure seamless functionality and user satisfaction. The User Creation module, led by the front-end development team, provides administrators with an intuitive interface for adding users and assigning roles. The back-end team, utilizing .NET, handles the logic behind user creation, implementing secure authentication and role-based access control.

The Transaction Management module involves collaboration between front-end and back-end teams to enable users to add income and expenses. Front-end developers design the transaction interfaces, while back-end developers ensure secure data processing and storage in MS SQL. The optional features, such as specifying income sources and attaching receipts, are seamlessly integrated for user convenience.

4.6 Project Management

The management of the Expense Tracker System development progress is a systematic and collaborative process that involves several key components to ensure efficiency, quality, and successful delivery. Here's a breakdown of how the progress is Clearly define the project scope, objectives, and deliverables.

Resource Allocation: Identify and allocate resources, including development teams, tools, and technologies.

Develop a realistic timeline, breaking down tasks into manageable sprints or milestones.

Form interdisciplinary teams, including front-end and back-end developers, UI/UX designers, security experts, testers, and project managers.

Establish regular communication channels to facilitate collaboration and information sharing.

Agile Development:

Iterative Approach: Adopt an Agile methodology to allow for iterative development, enabling continuous feedback and adaptation.

Sprints: Break down the development process into sprints with defined goals for each iteration.

4.7 Summary

This chapter provides an overall idea of our proposed solution, users, input, output, and technology of our system. In addition, we describe the software process model and the methods we use to address the problems of other modules we used.

Chapter 5 Analysis and Design

5.1 Class Diagram

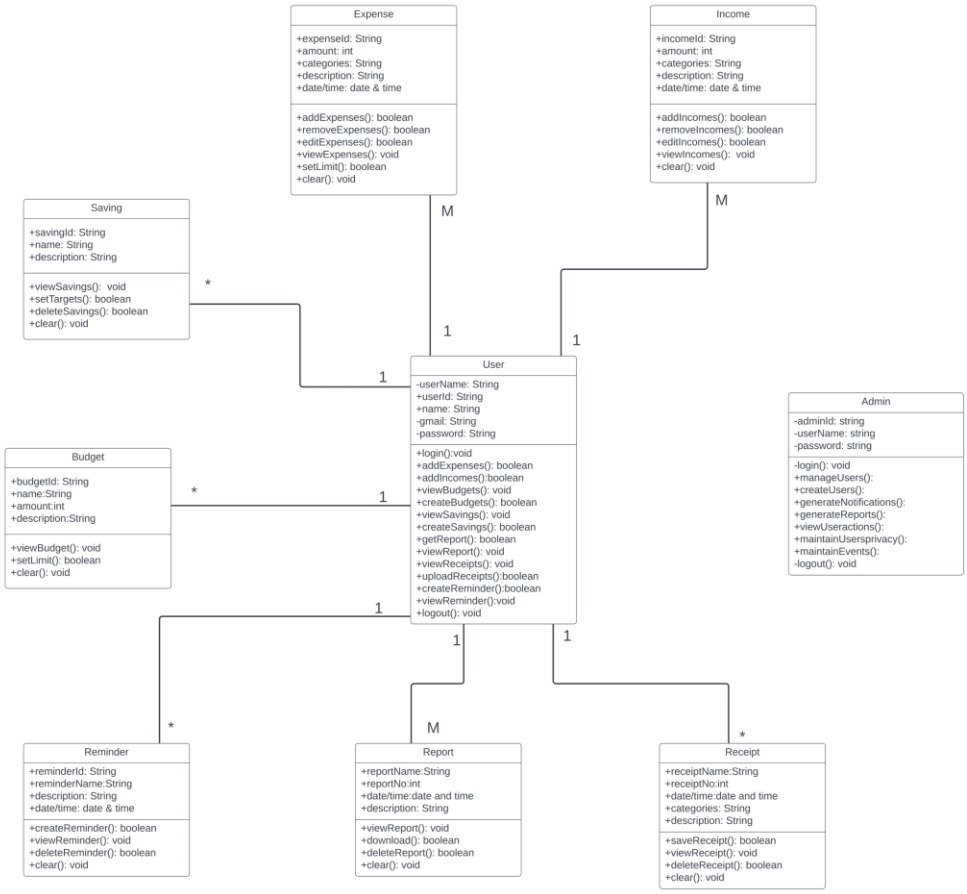

Figure 4:Class Diagram

5.2 Use case Diagram

Figure 5:Use Case Diagram(a)

Figure 6:Use Case Diagram(b)

5.3 Activity Diagrams

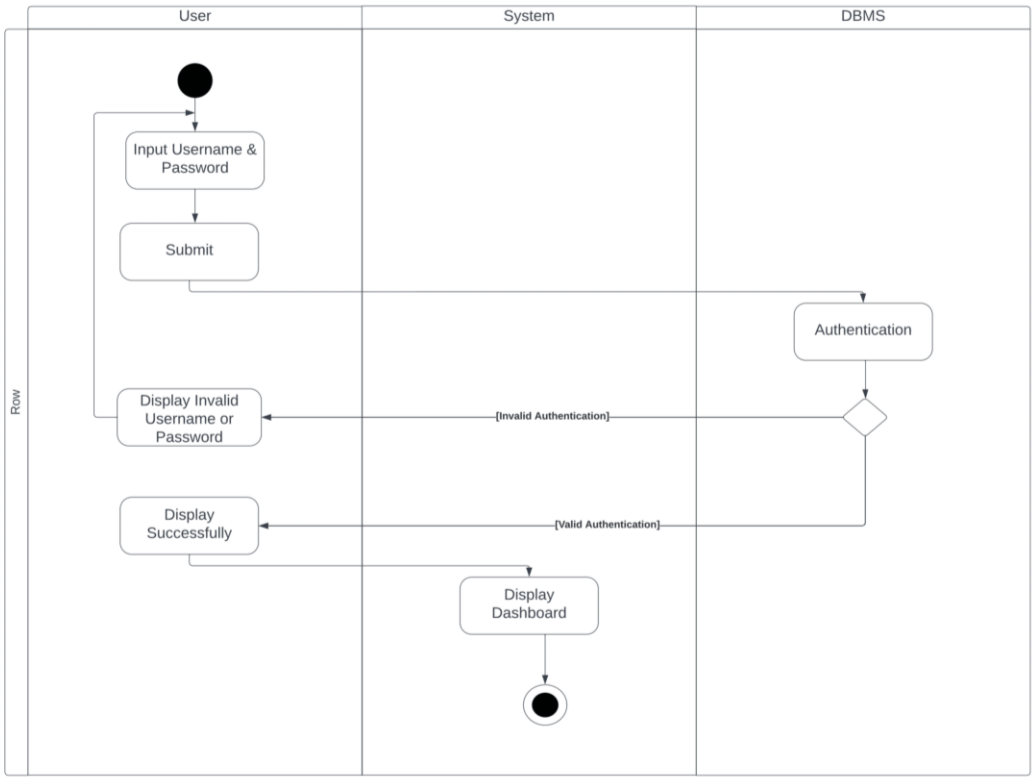

Figure 7:Activity Diagram(Login)

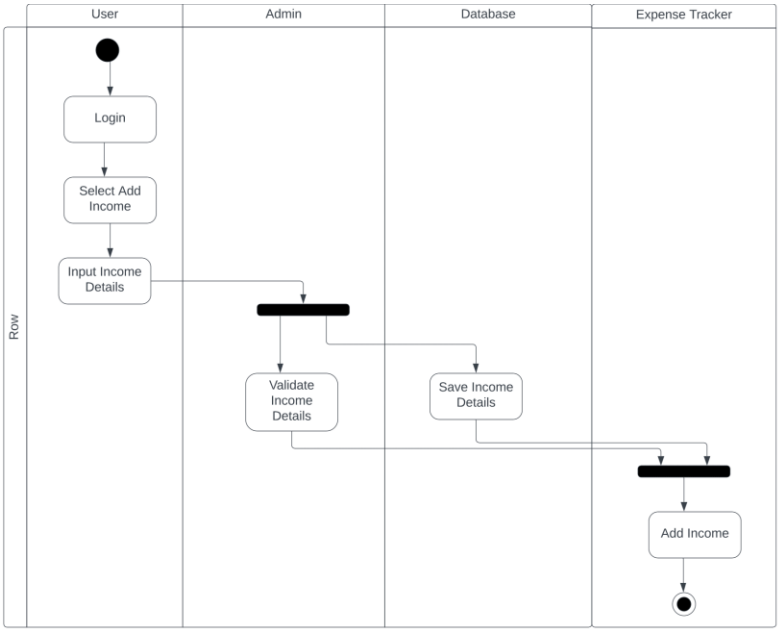

Figure 8:Activity Diagram(Add Income)

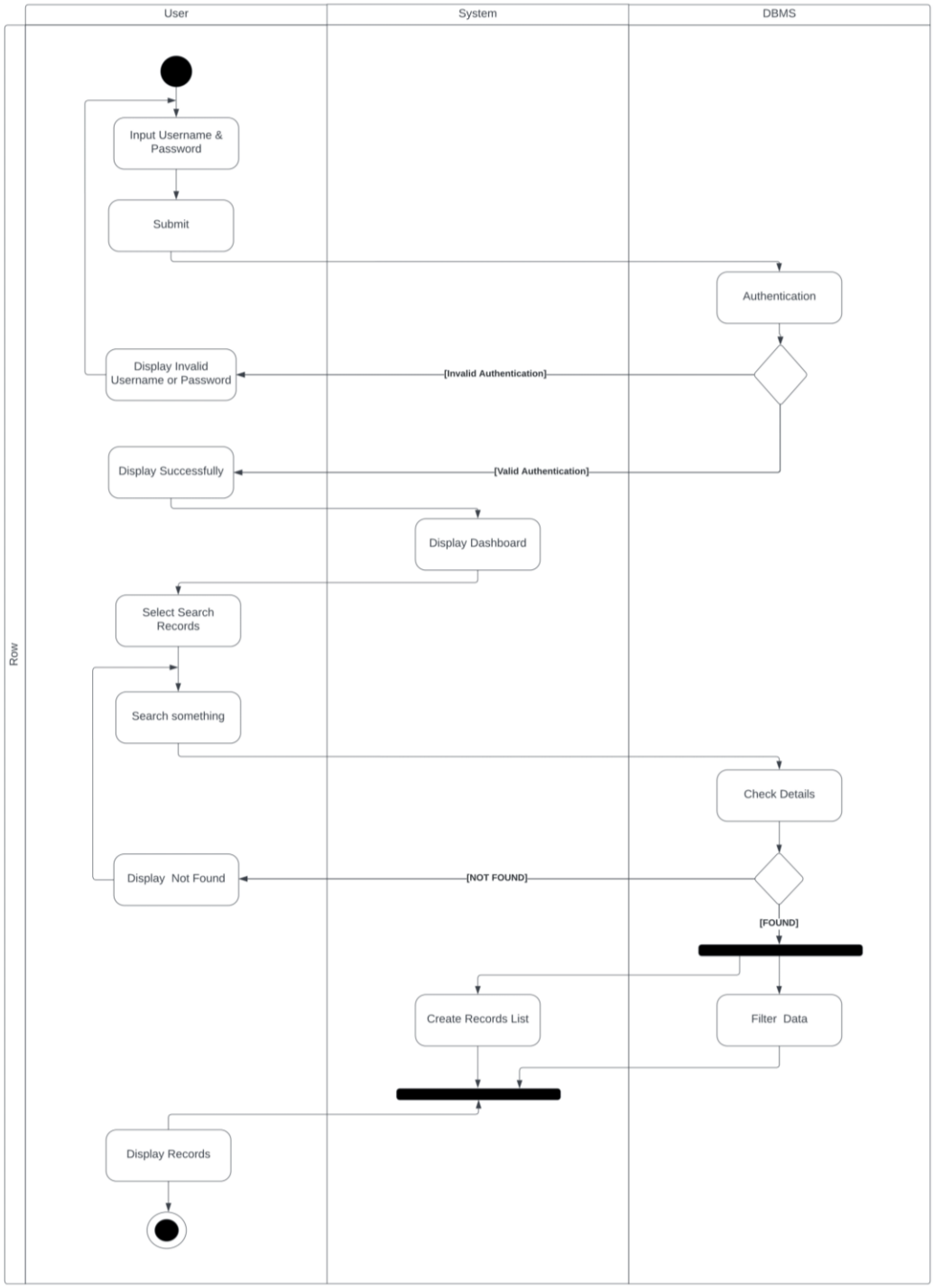

Figure 9:Activity Diagram(Search Records)

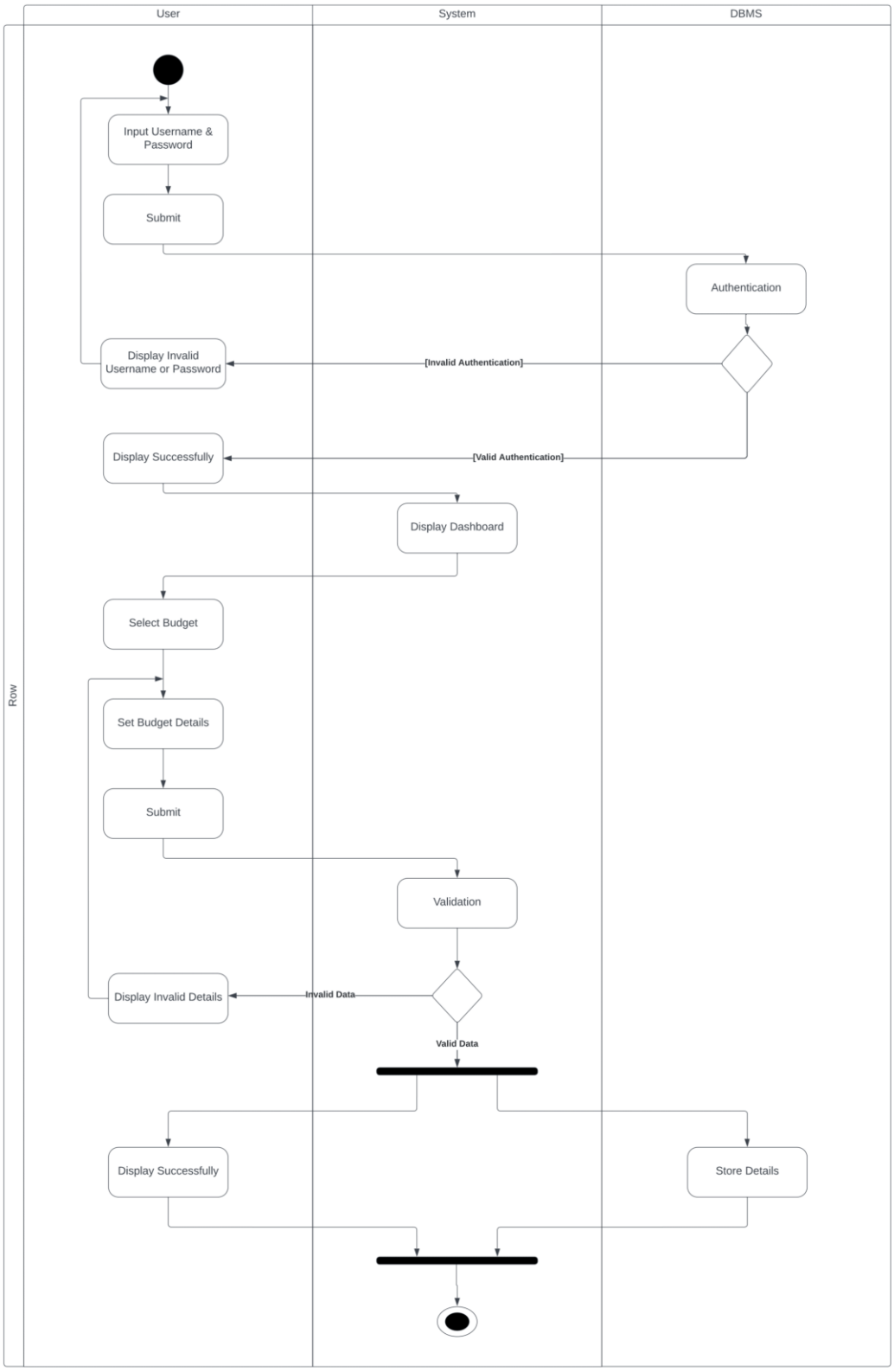

Figure 10:Activity Diagram(Create Budget)

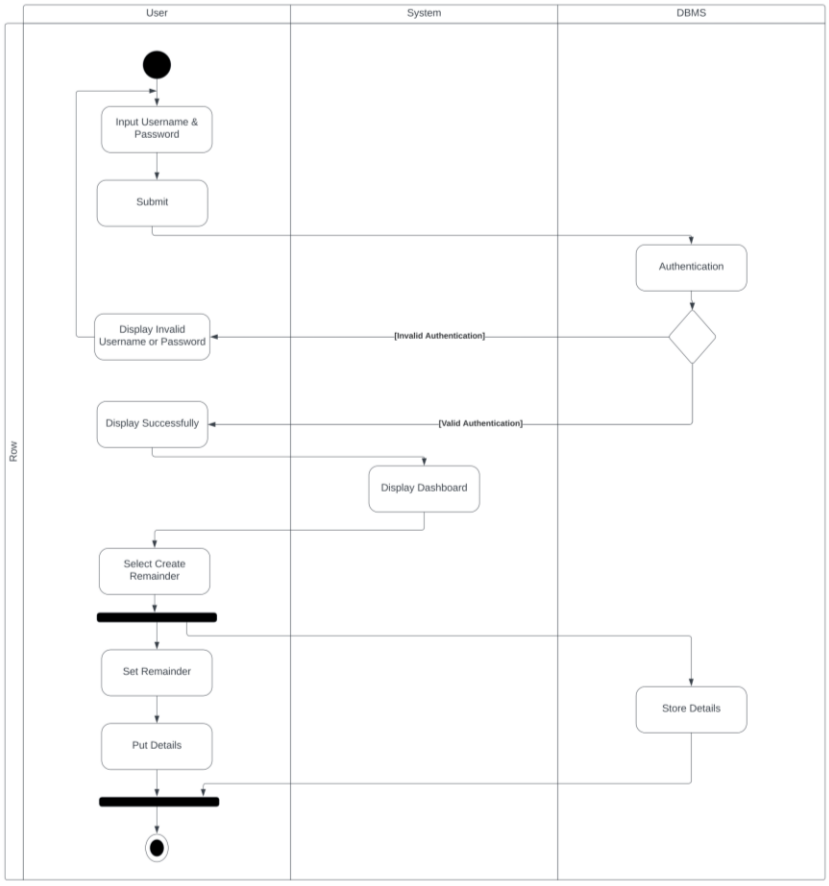

Figure 11:Activity Diagram(Create Remainder)

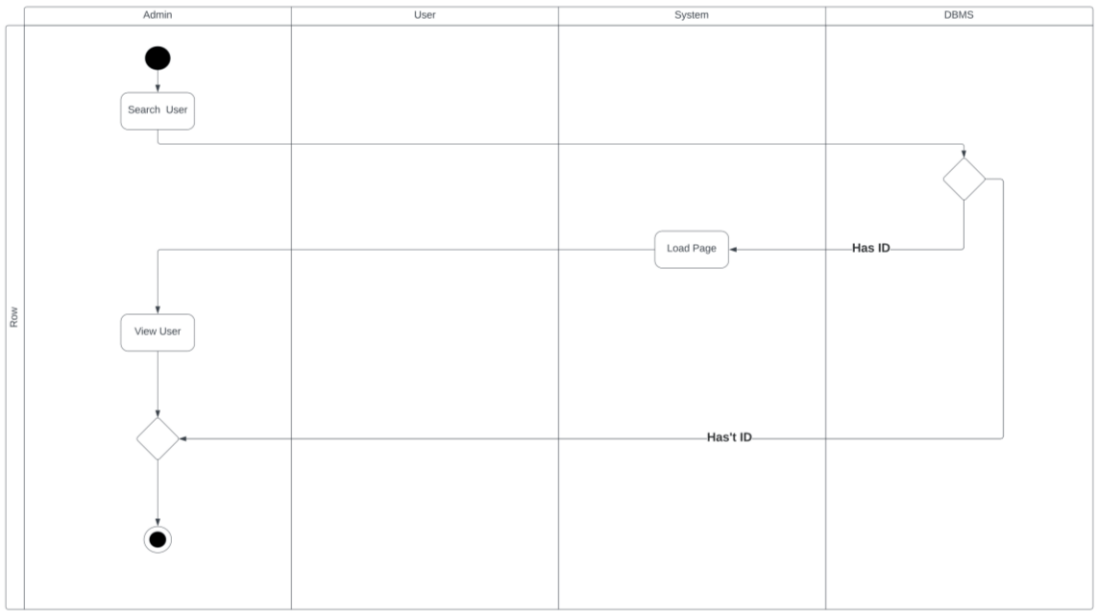

Figure 12:Activity Diagram(Search User)

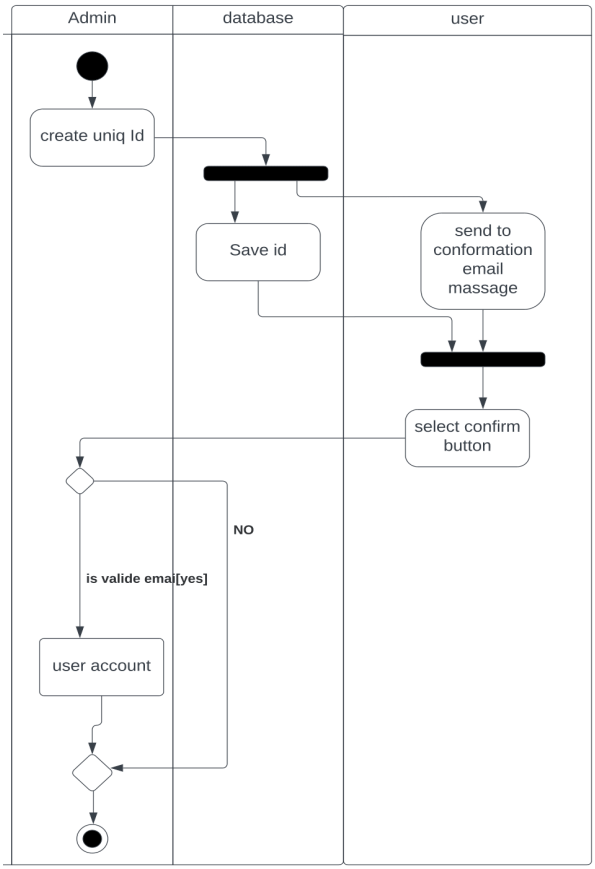

Figure 13:Activity Diagram(Create User)

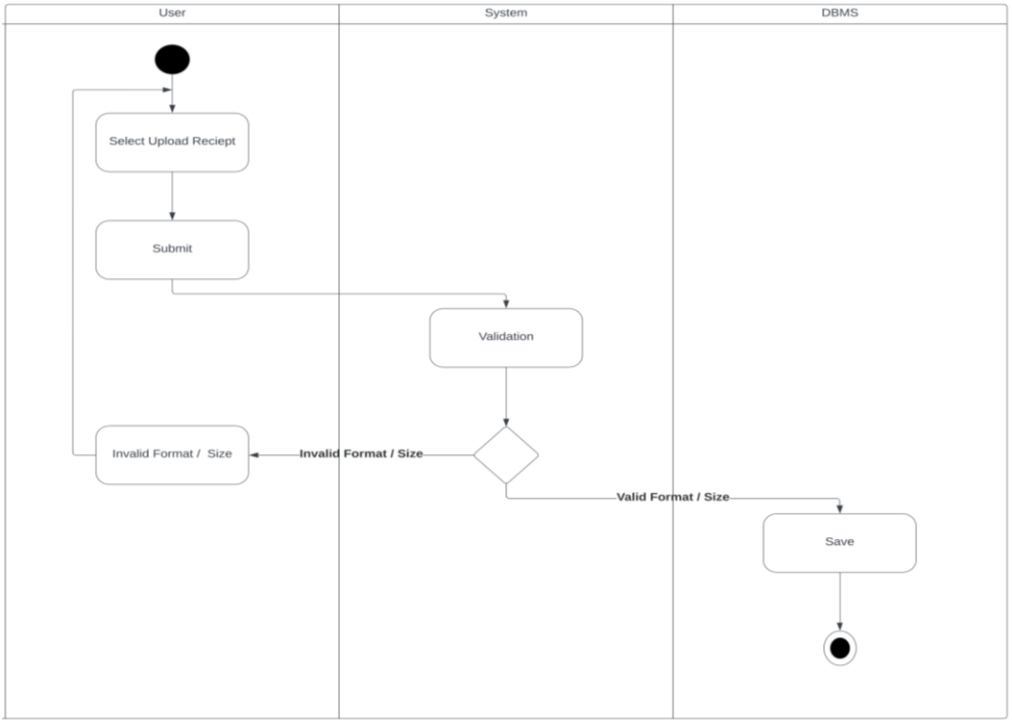

Figure 14:Activity Diagram(Upload Receipt)

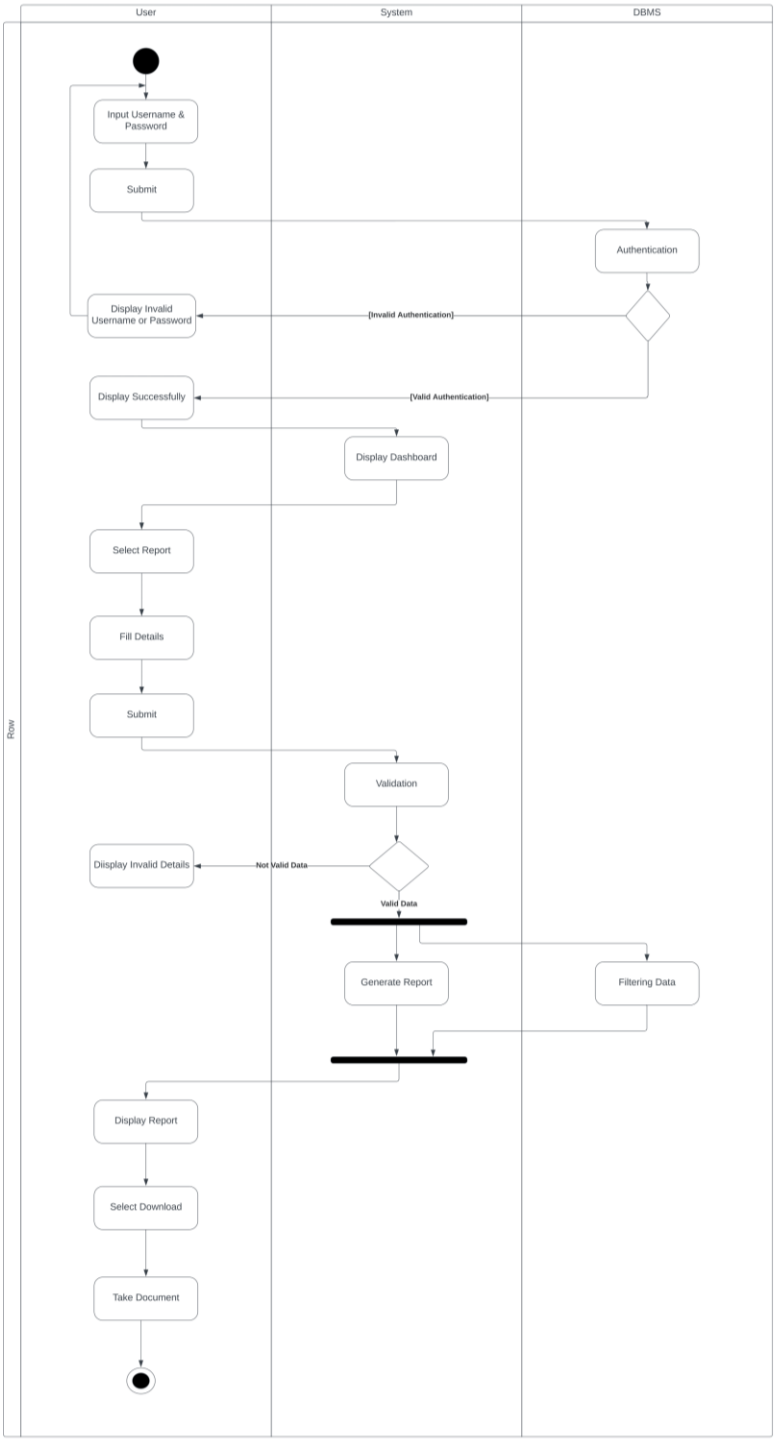

Figure 15:Activity Diagram(Download Report)

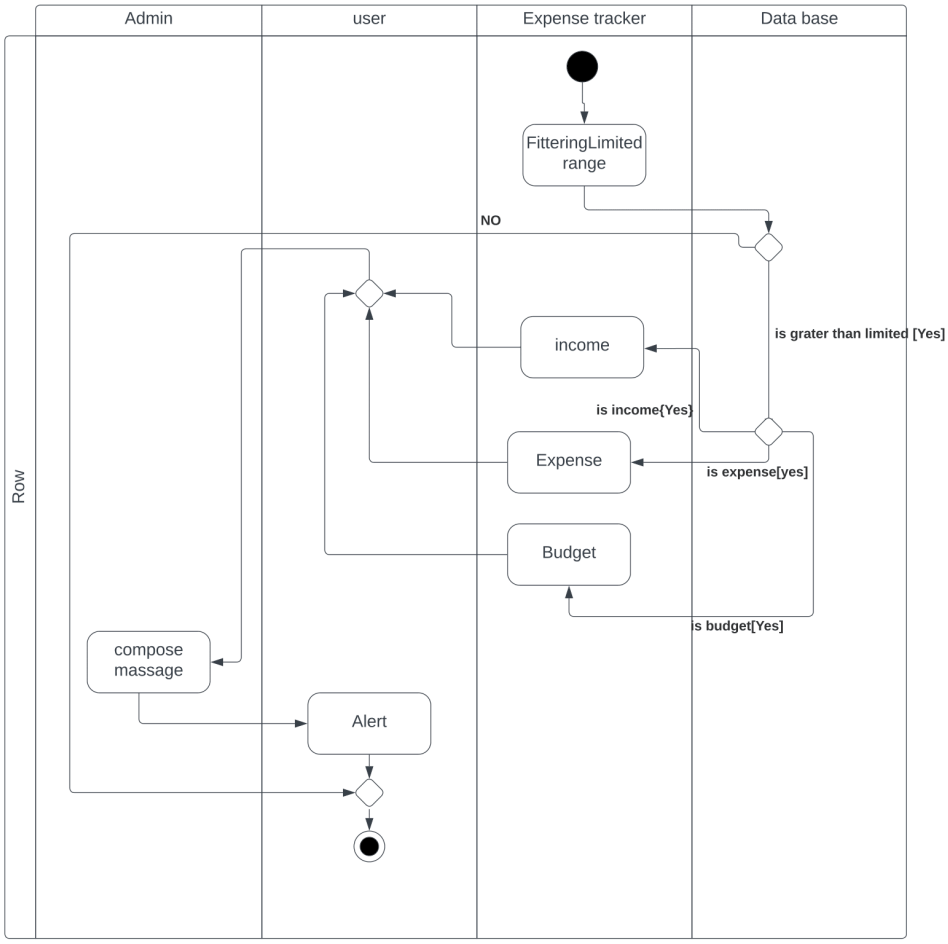

Figure 16:Activity Diagram(Send Alert)

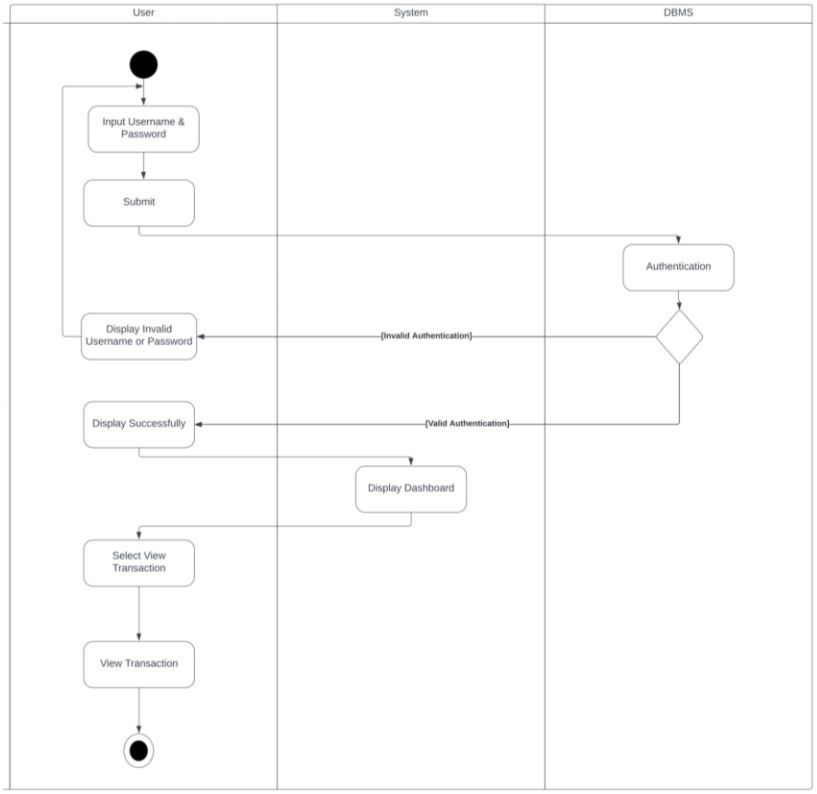

Figure 17:Activity Diagram(View Transaction)

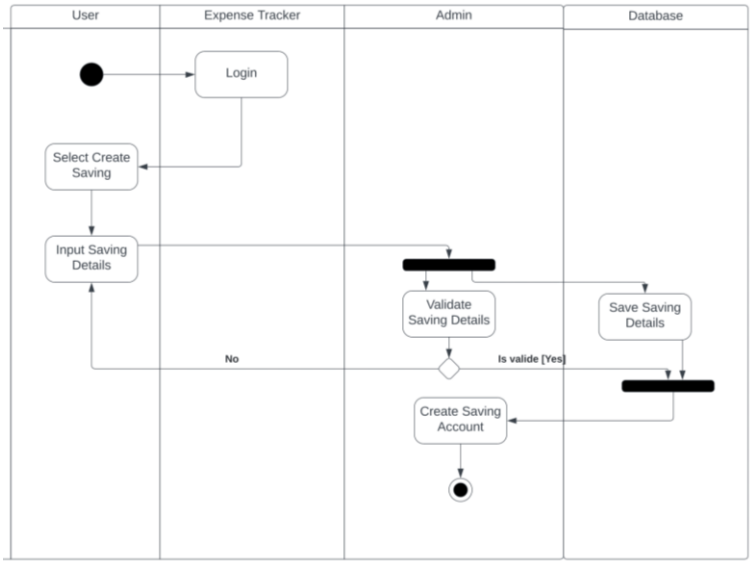

Figure 18:Activity Diagram(Create Saving)

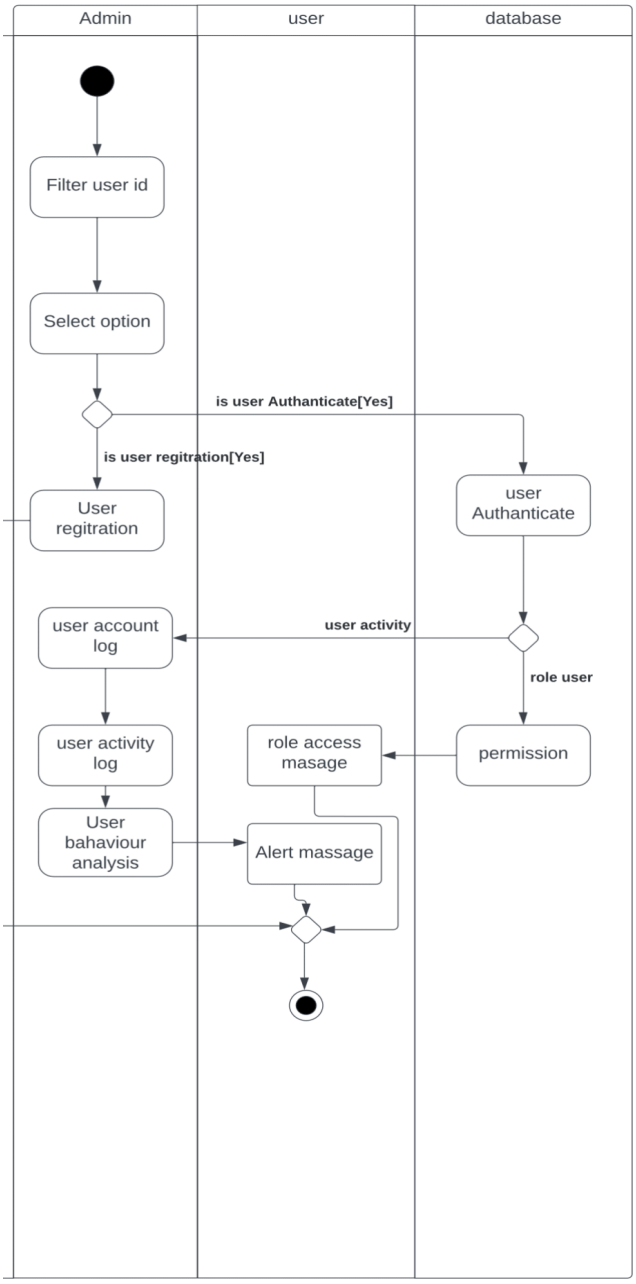

Figure 19:Activity Diagram(Manage User)

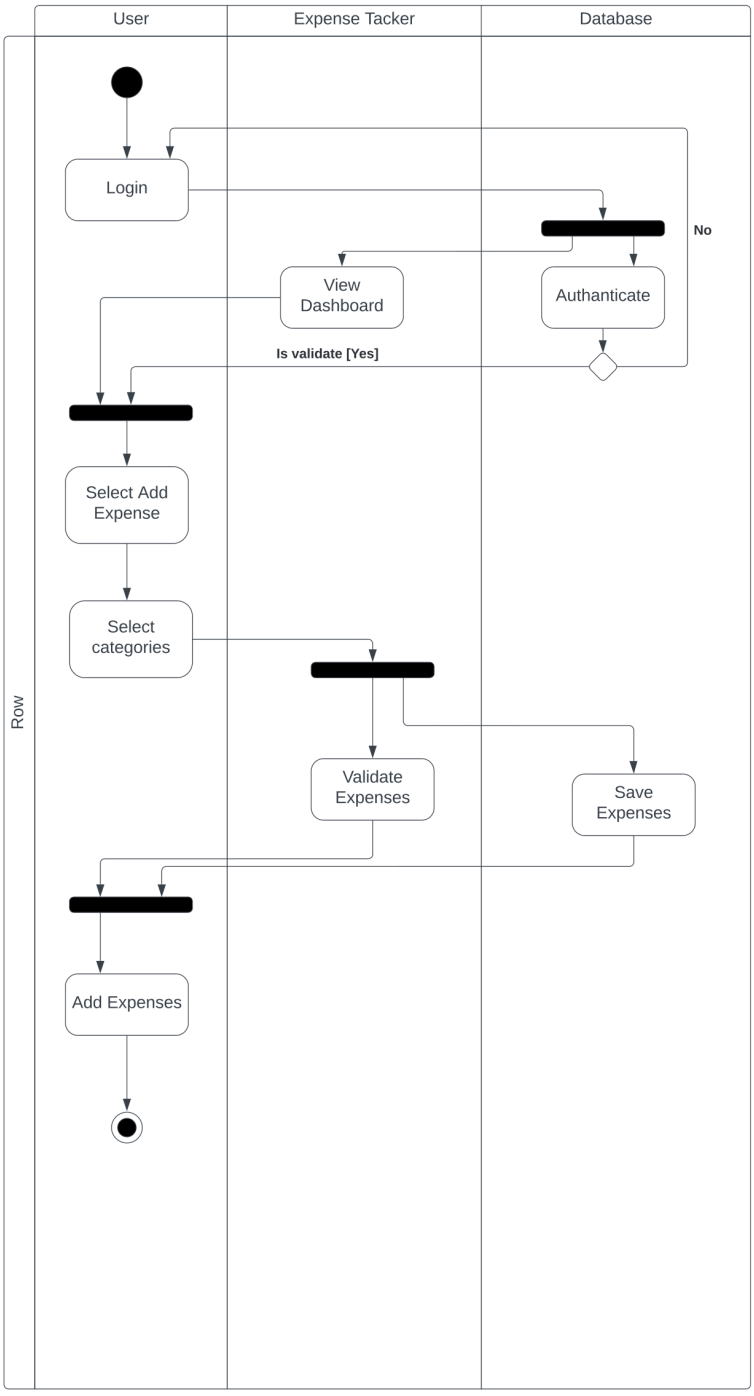

Figure 20:Activity Diagram(Add Expenses)

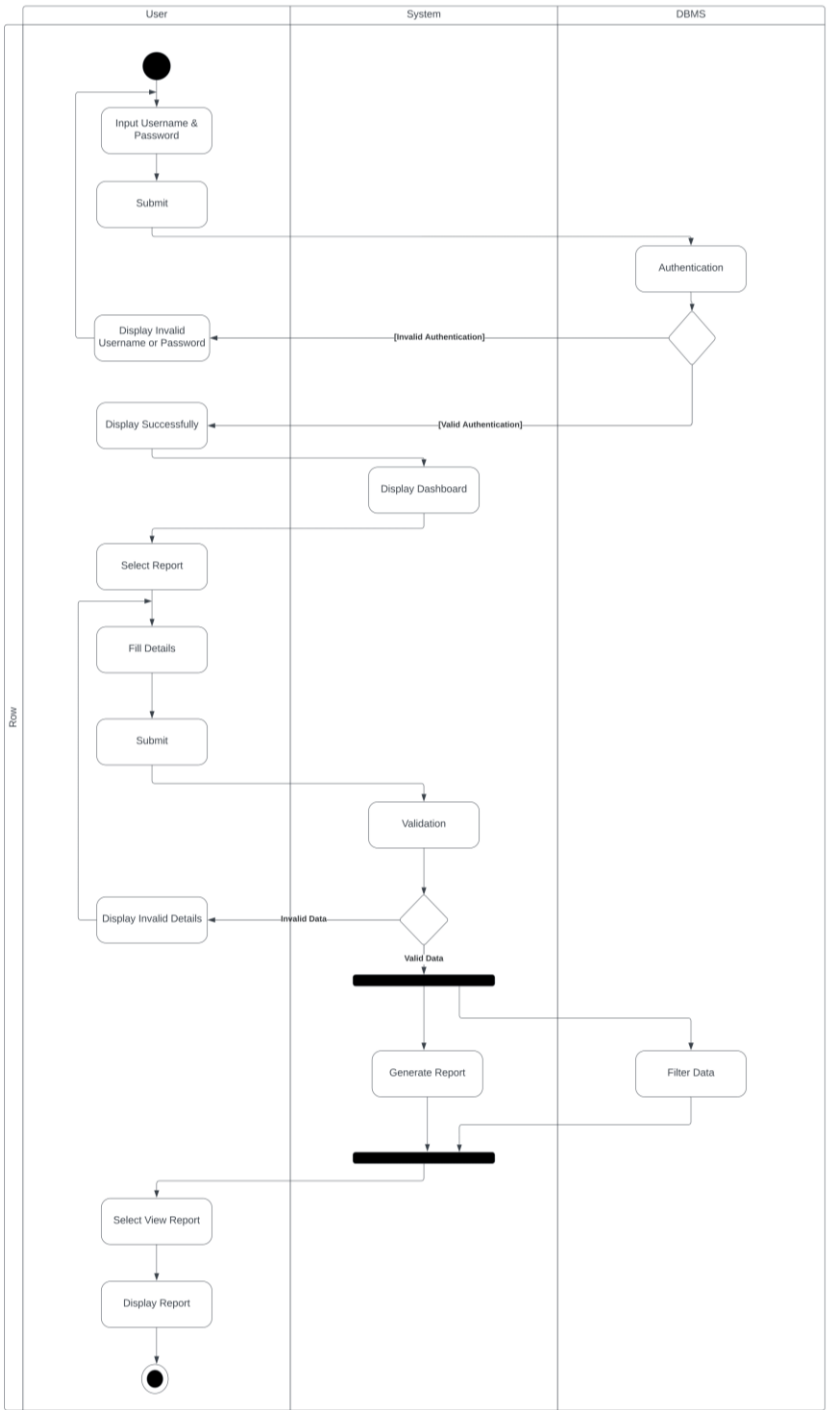

Figure 21:Activity Diagram(View Report)

5.4 Sequence Diagrams

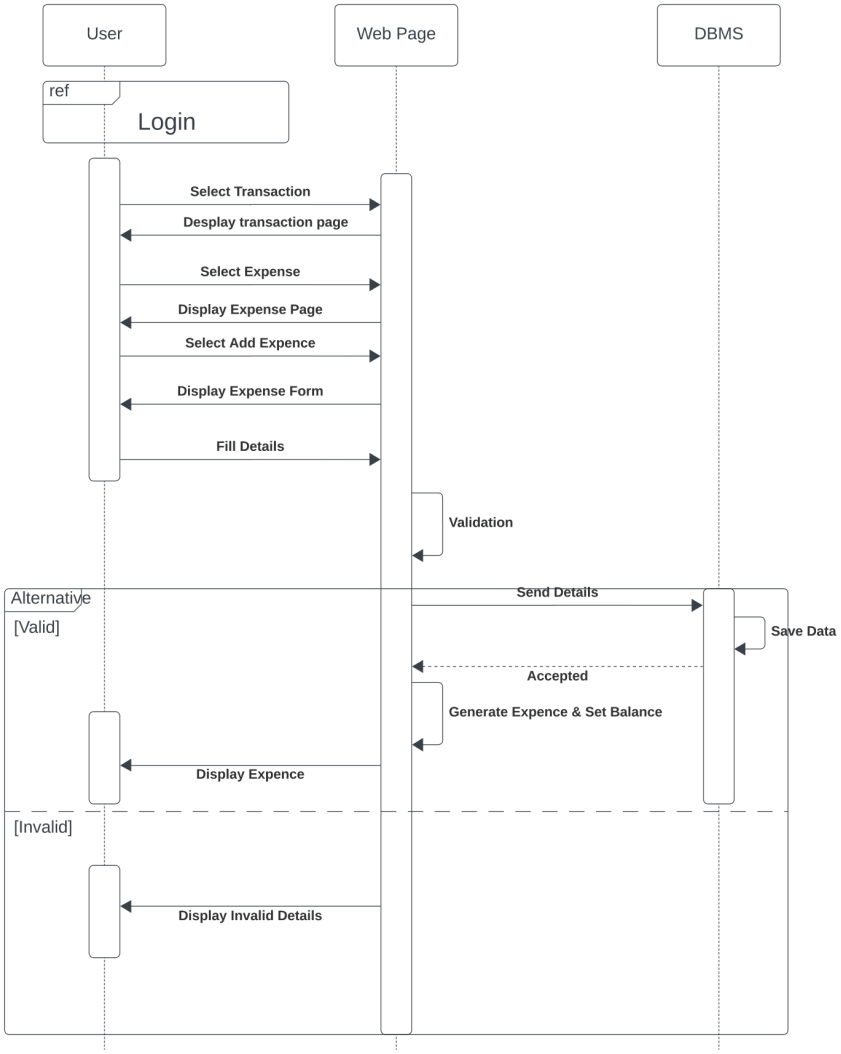

Figure 22:Sequence Diagram(Add Expenses)

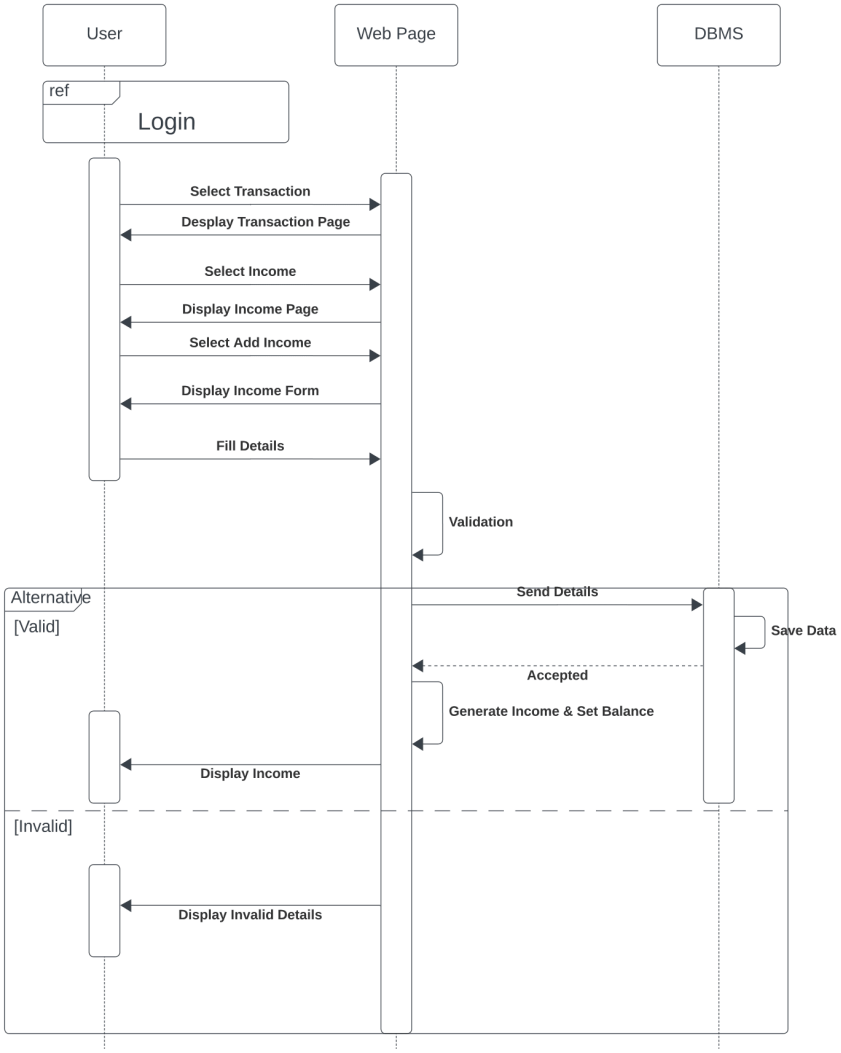

Figure 23:Sequence Diagram(Add Income)

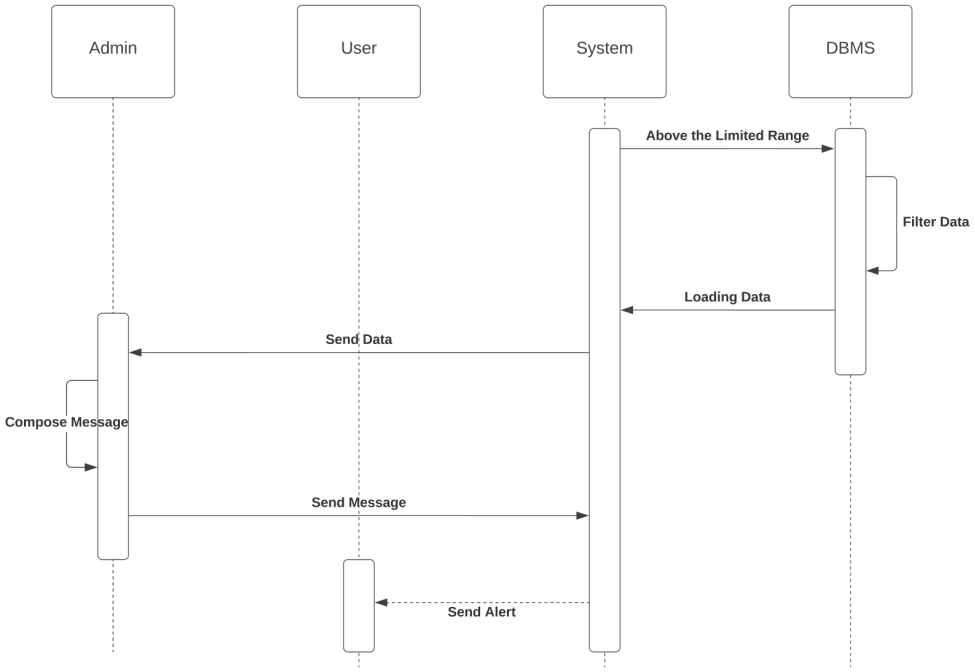

Figure 24:Sequence Diagram(Send Alert)

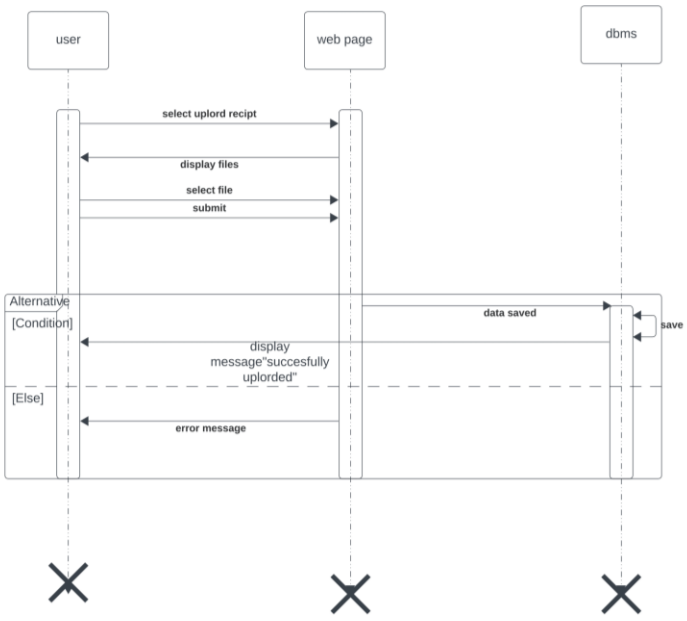

Figure 25:Sequence Diagram(Upload Receipt)

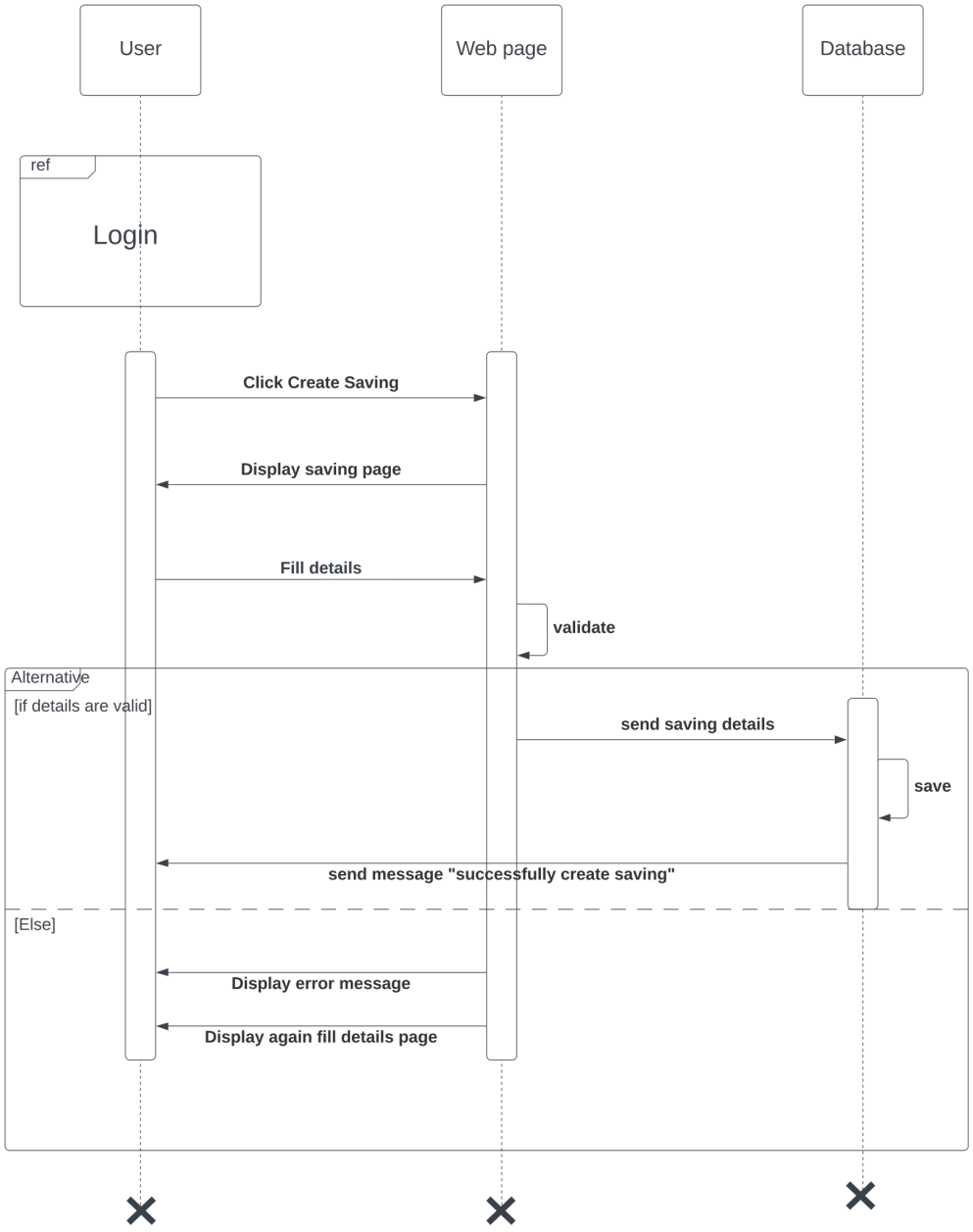

Figure 26:Sequence Diagram(Create Saving)

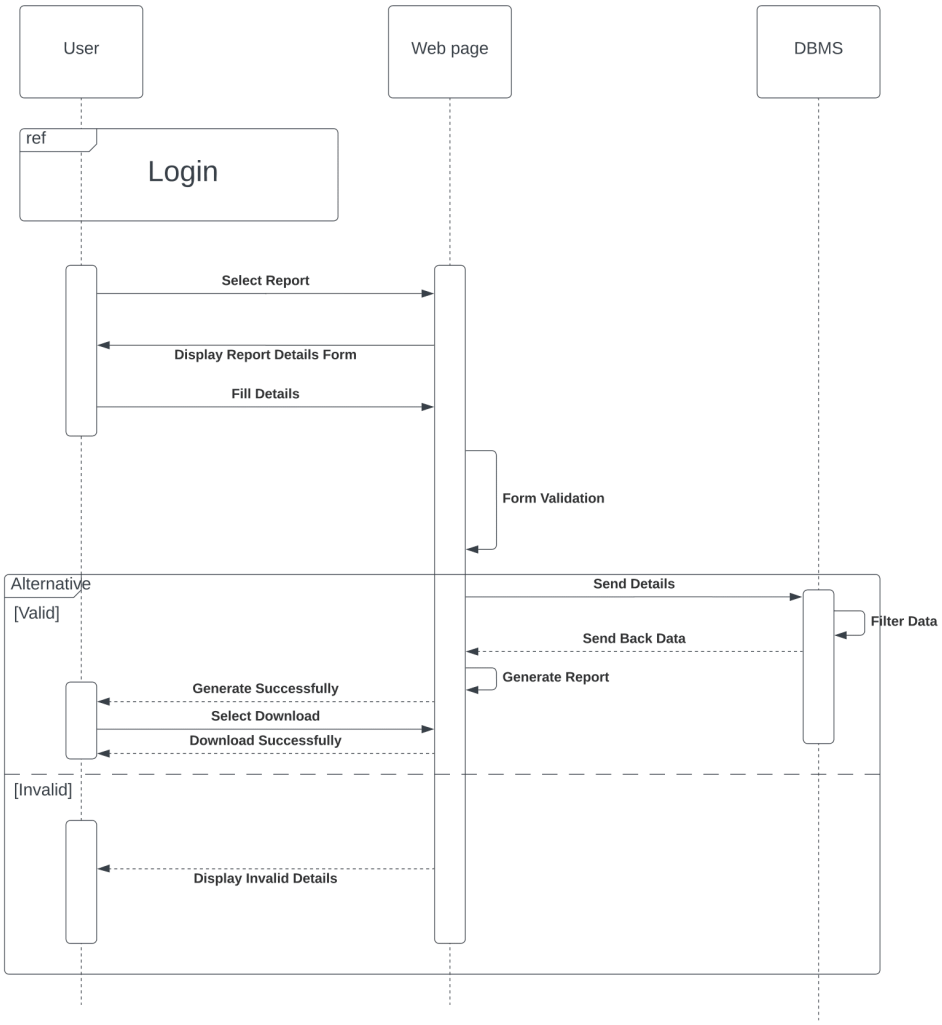

Figure 27:Sequence Diagram(Download Report)

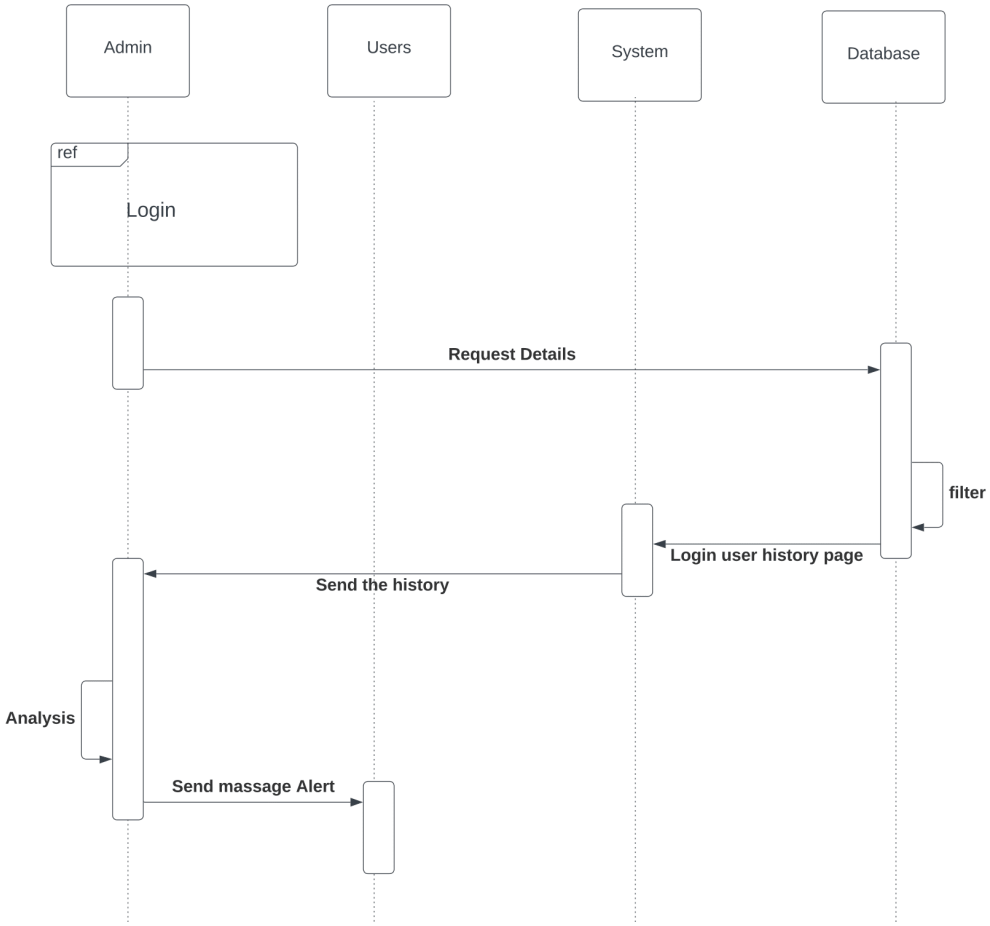

Figure 28:Sequence Diagram(Manage User)

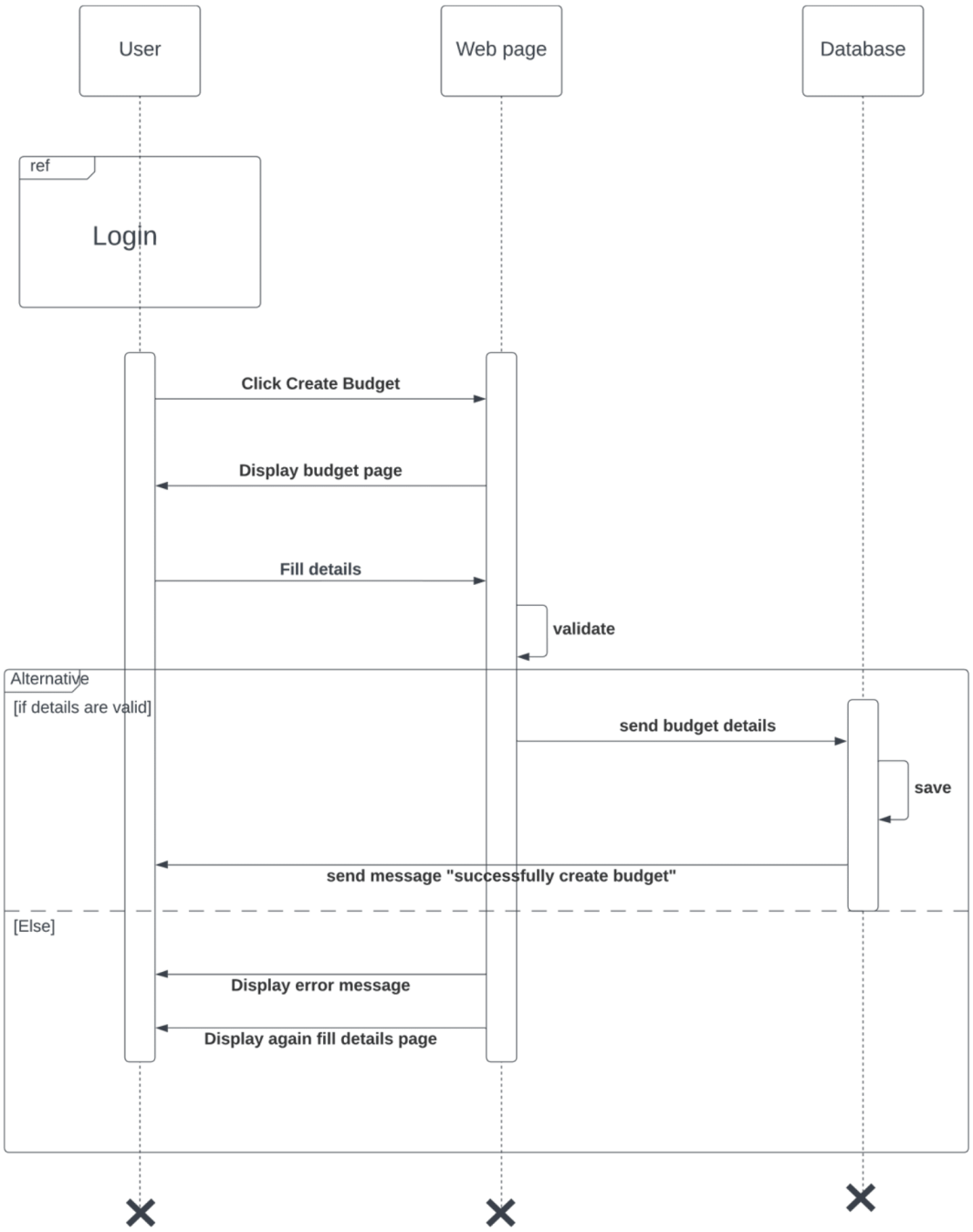

Figure 29:Sequence Diagram(Create Budget)

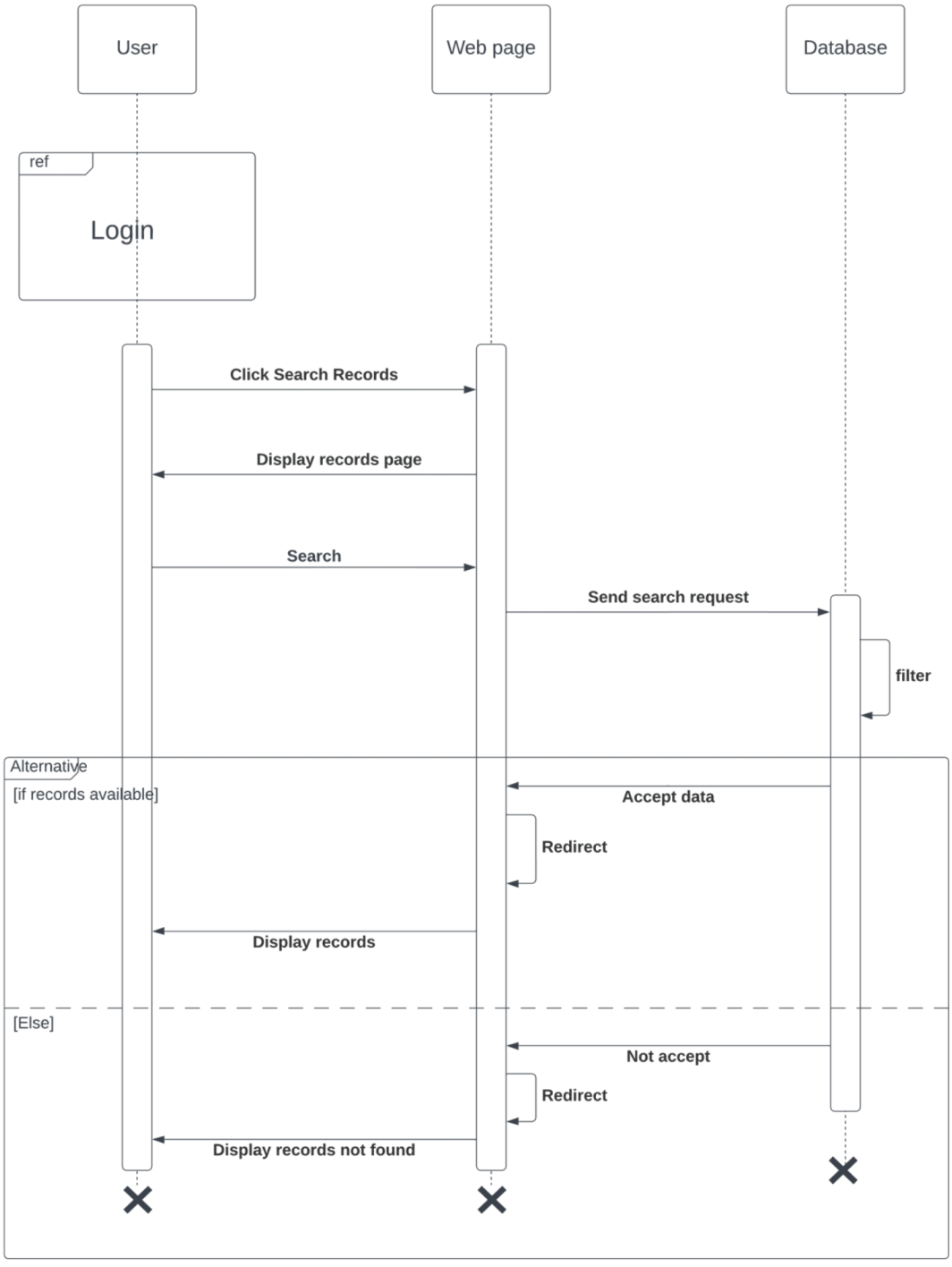

Figure 30:Sequence Diagram(Search Records)

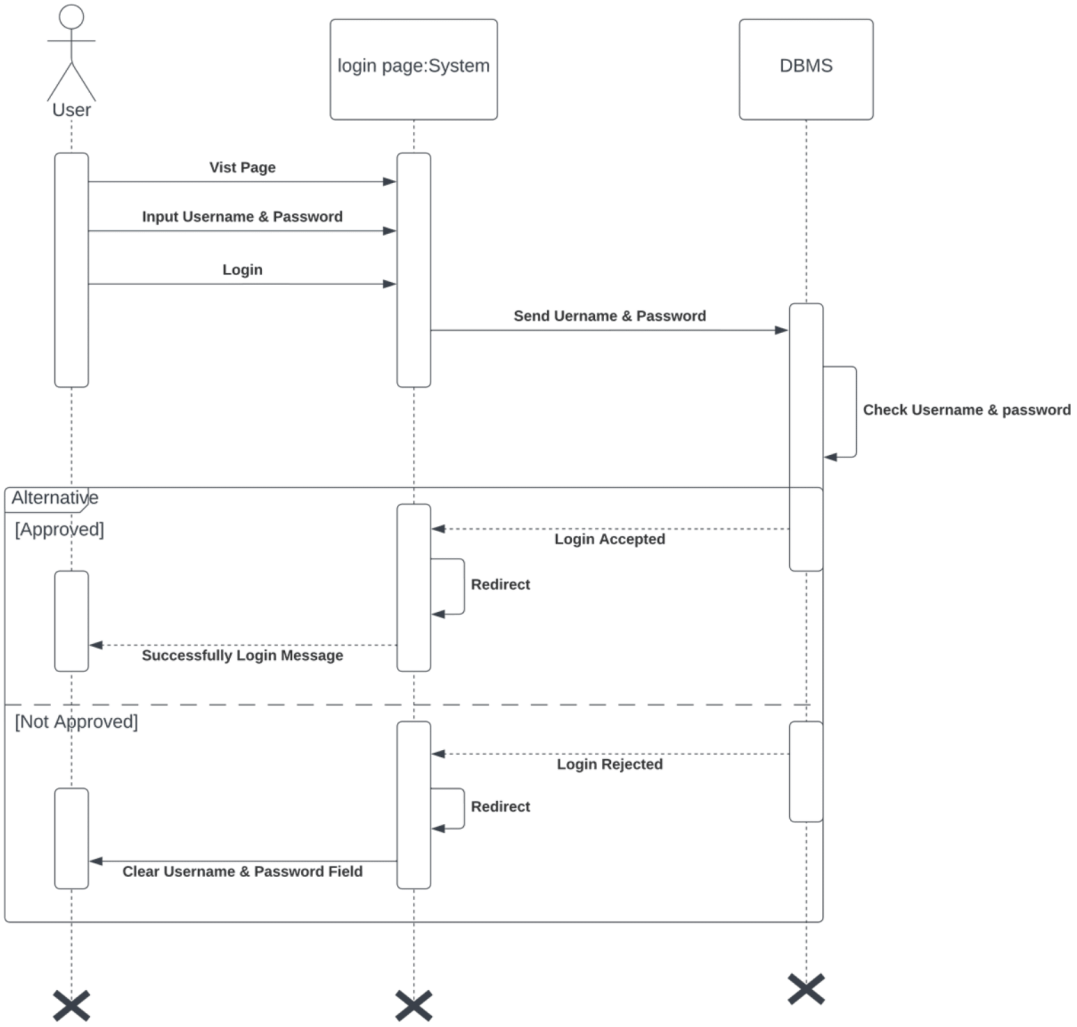

Figure 31:Sequence Diagram(Login)

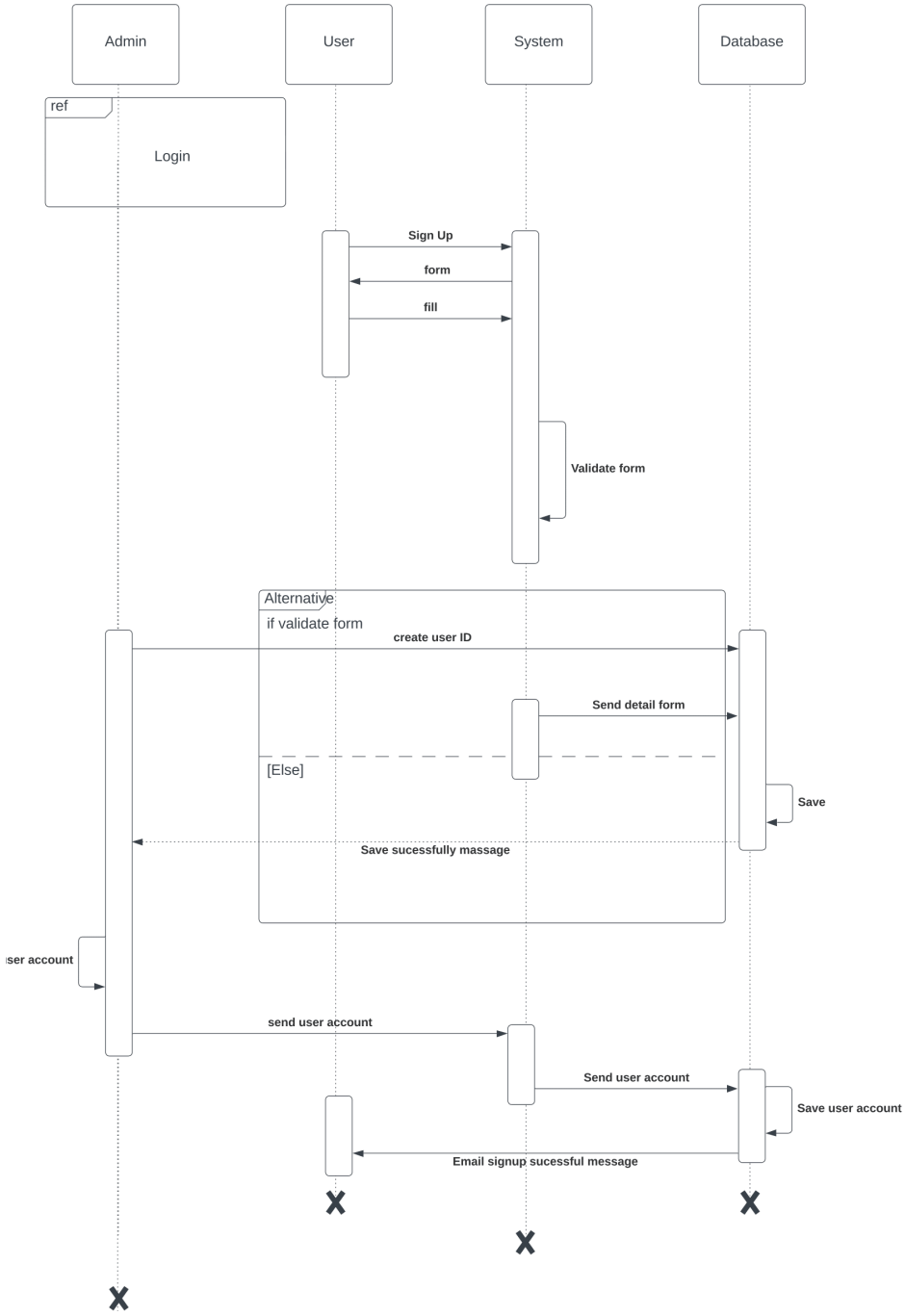

Figure 32:Sequence Diagram(Create User)

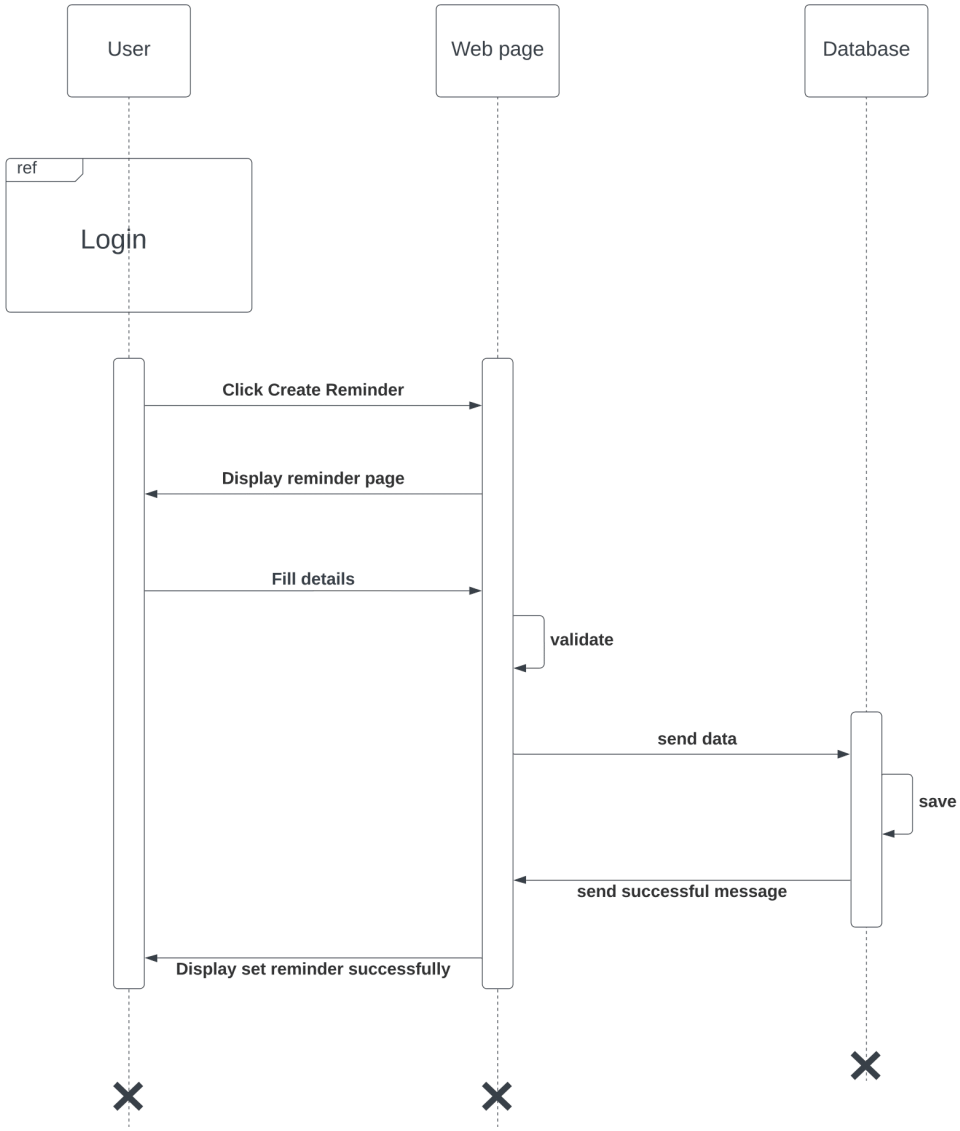

Figure 33:Sequence Diagram(Create Remainder)

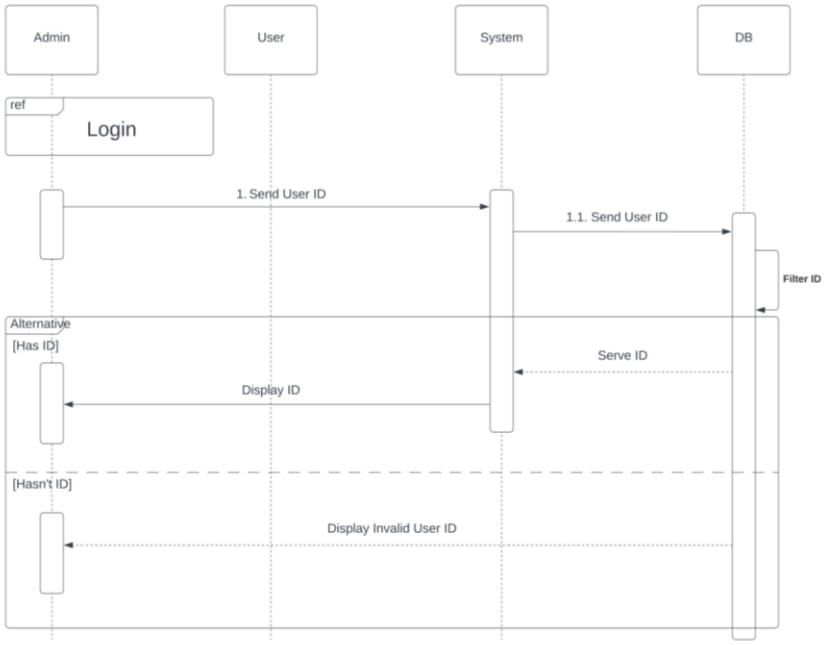

Figure 34:Sequence Diagram(Search User)

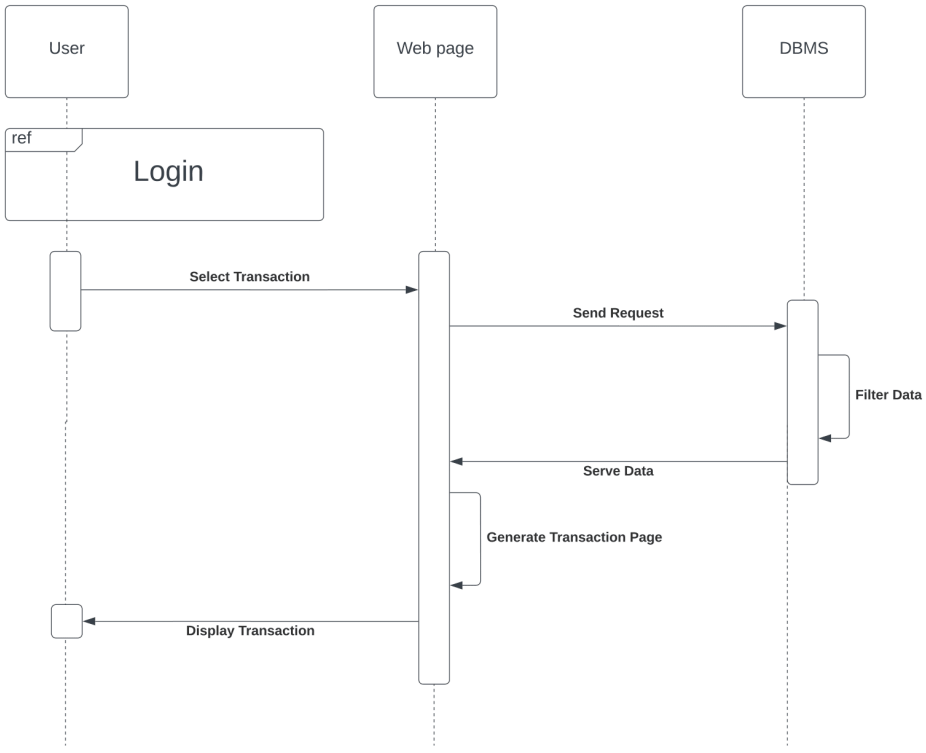

Figure 35:Sequence Diagram(View Transaction)

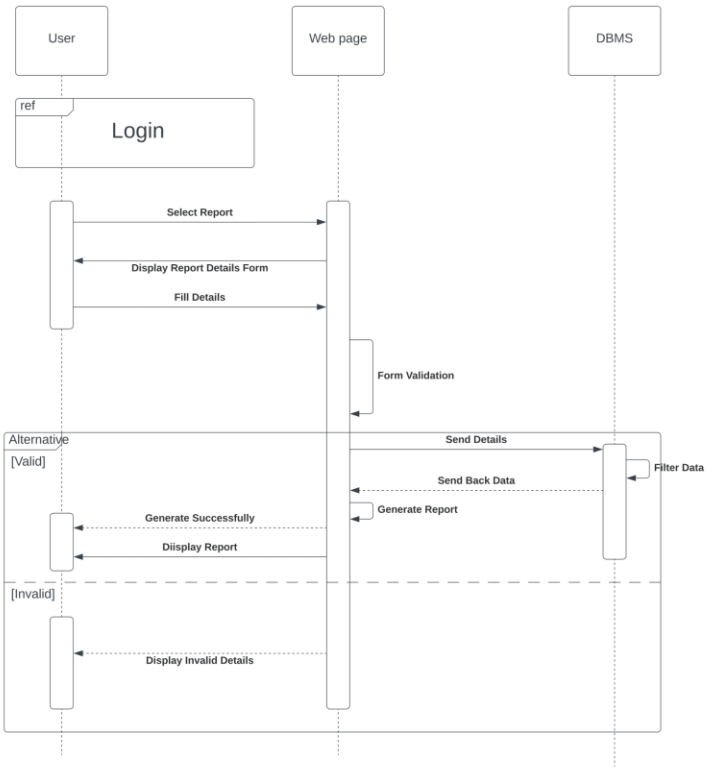

Figure 36:Sequence Diagram(View Report)

5.5 ER Diagram

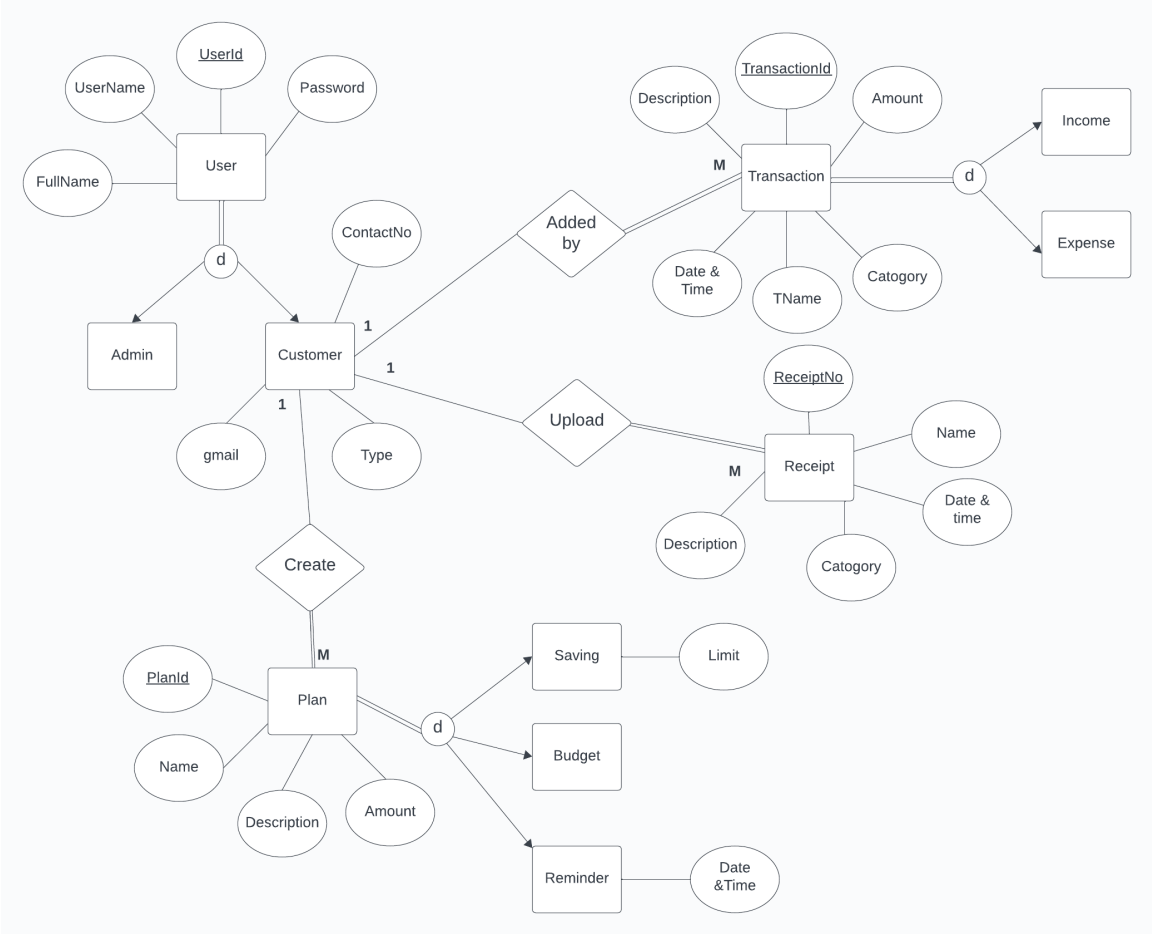

Figure 37:ER Diagram

Chapter 6 Implementation

6.1 Login Page

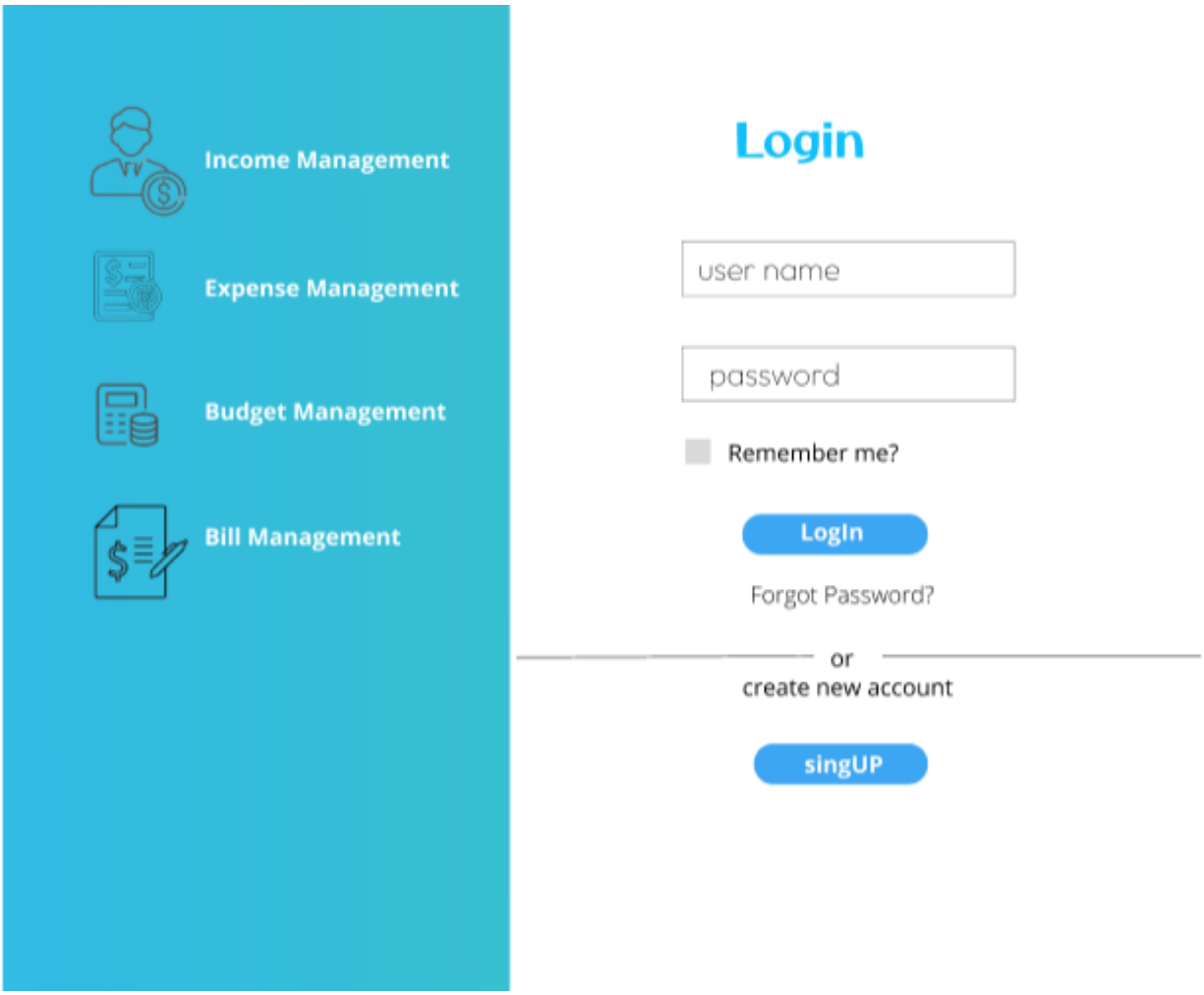

Figure 38:Interface(User Login)

6.2 Administrator's Dashboard

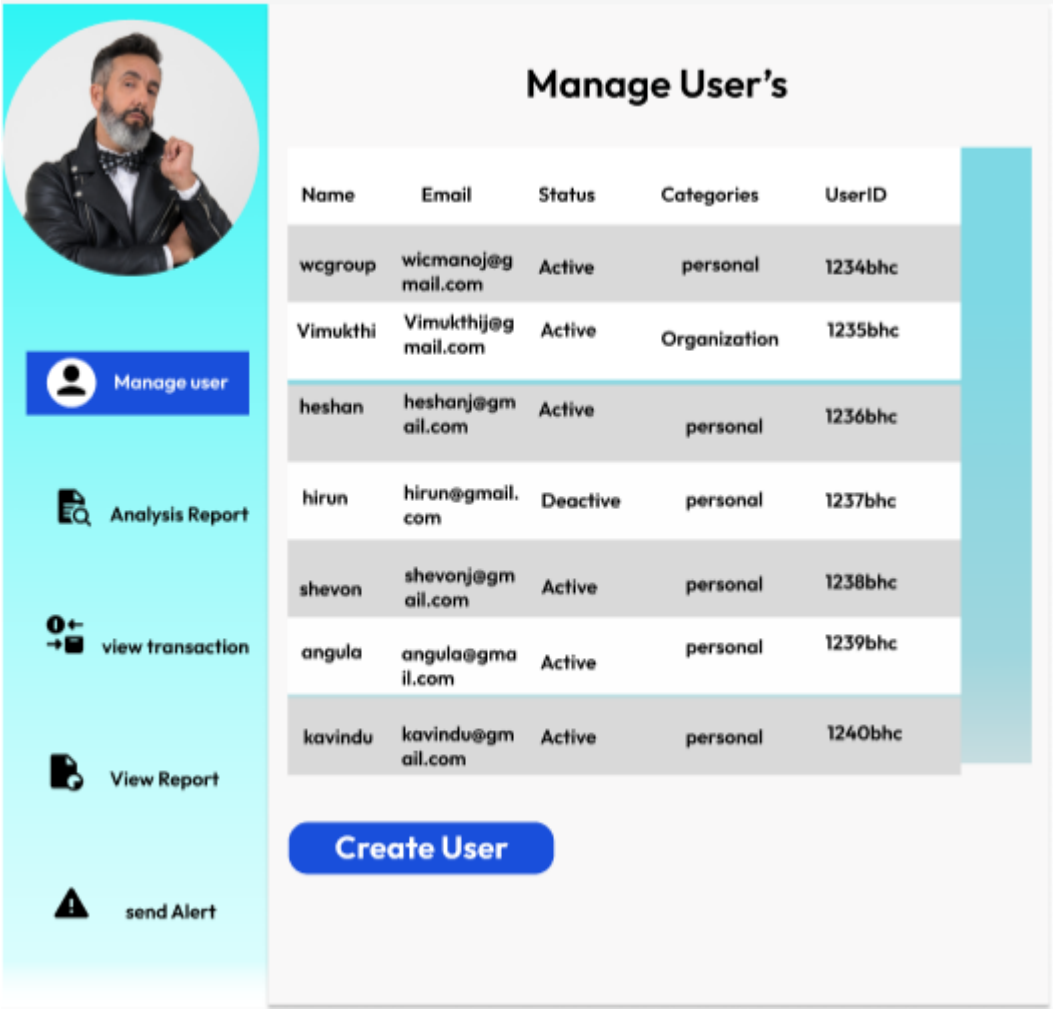

Figure 39:Interface(Administrator's Dashboard)

6.3 User's Dashboard

Figure 40:Interface(User Dashboard)

6.4 Transaction

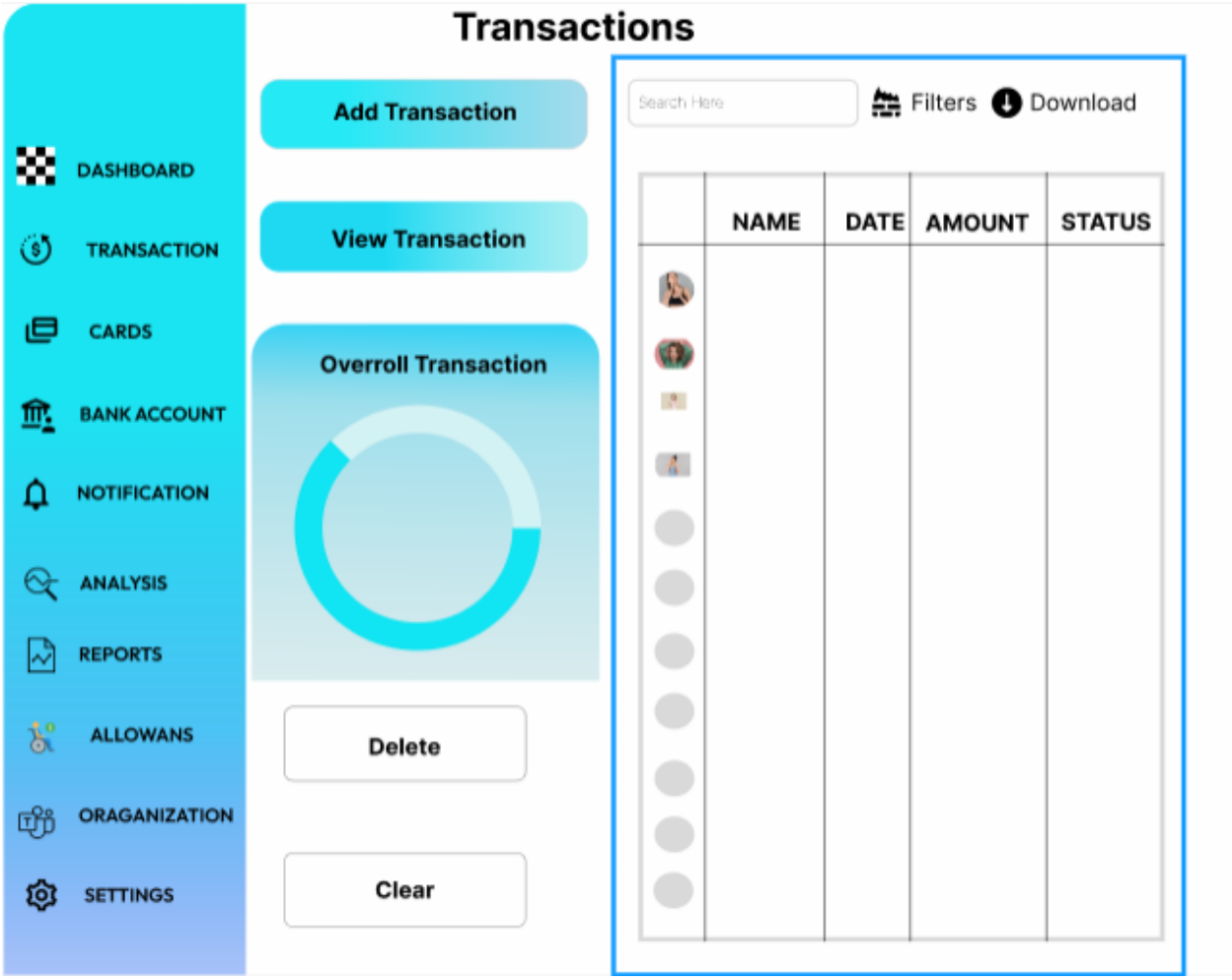

Figure 41:Interface(Transactions)

6.5 View Transaction

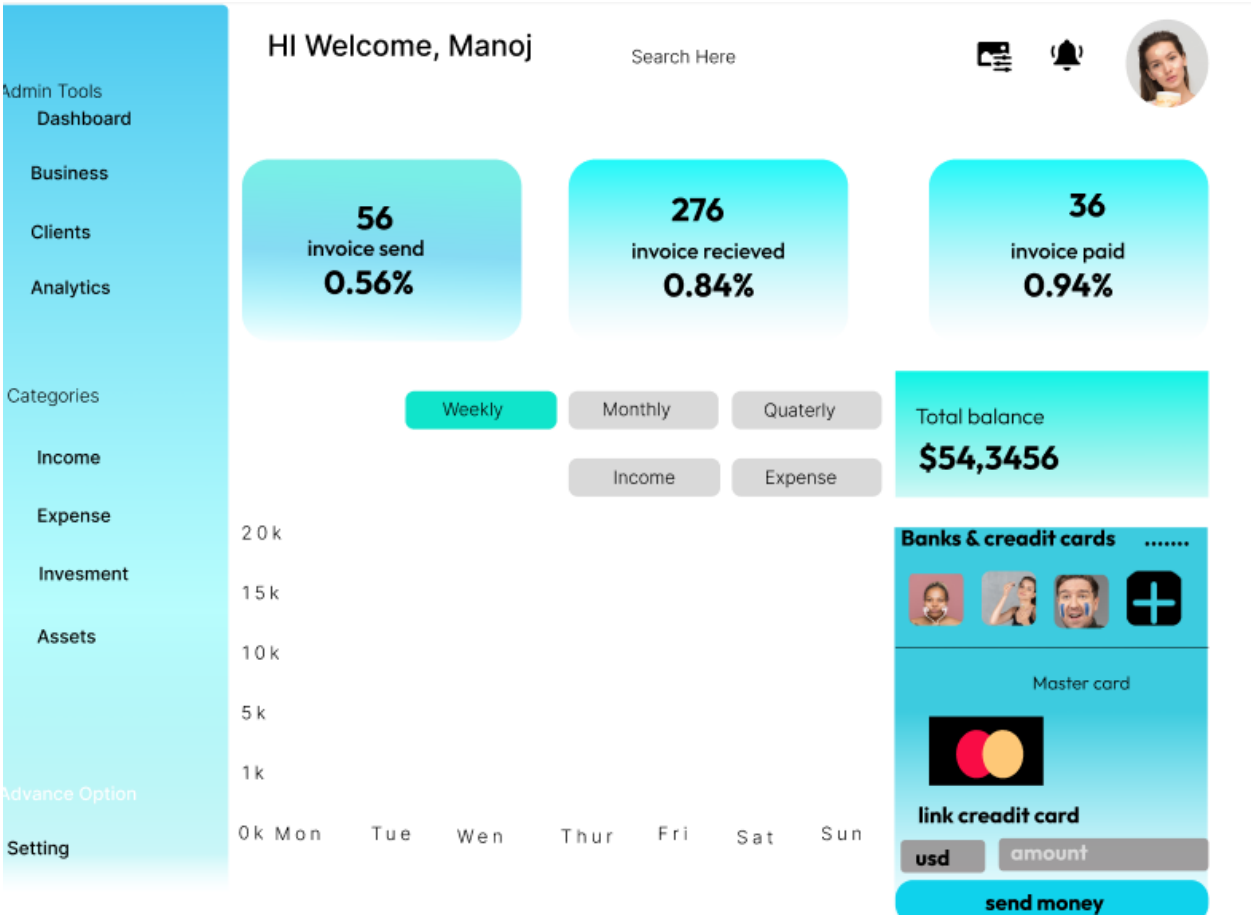

Figure 42:Interface(View Transactions)

6.6 Add Transaction

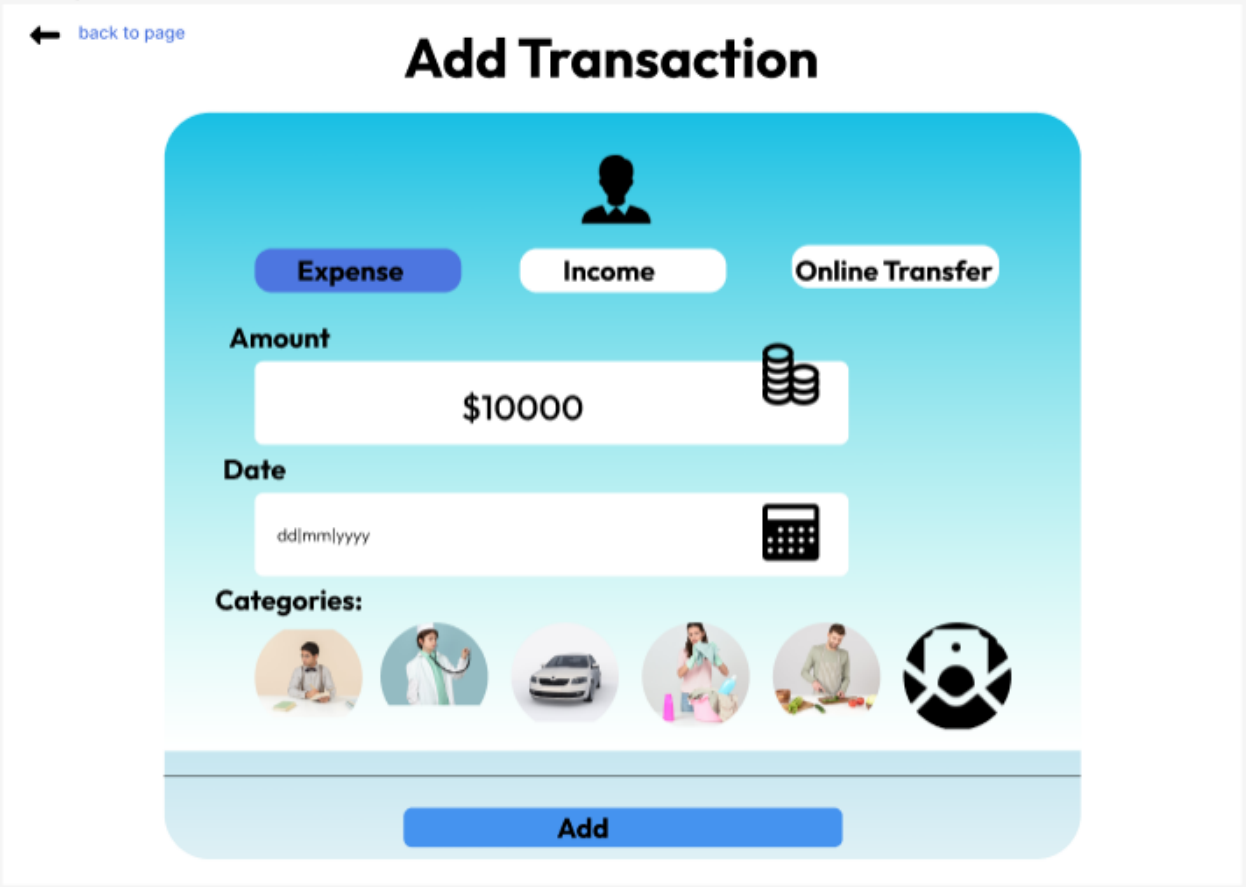

Figure 43:Interface(Add Transaction)

6.7 Create Saving

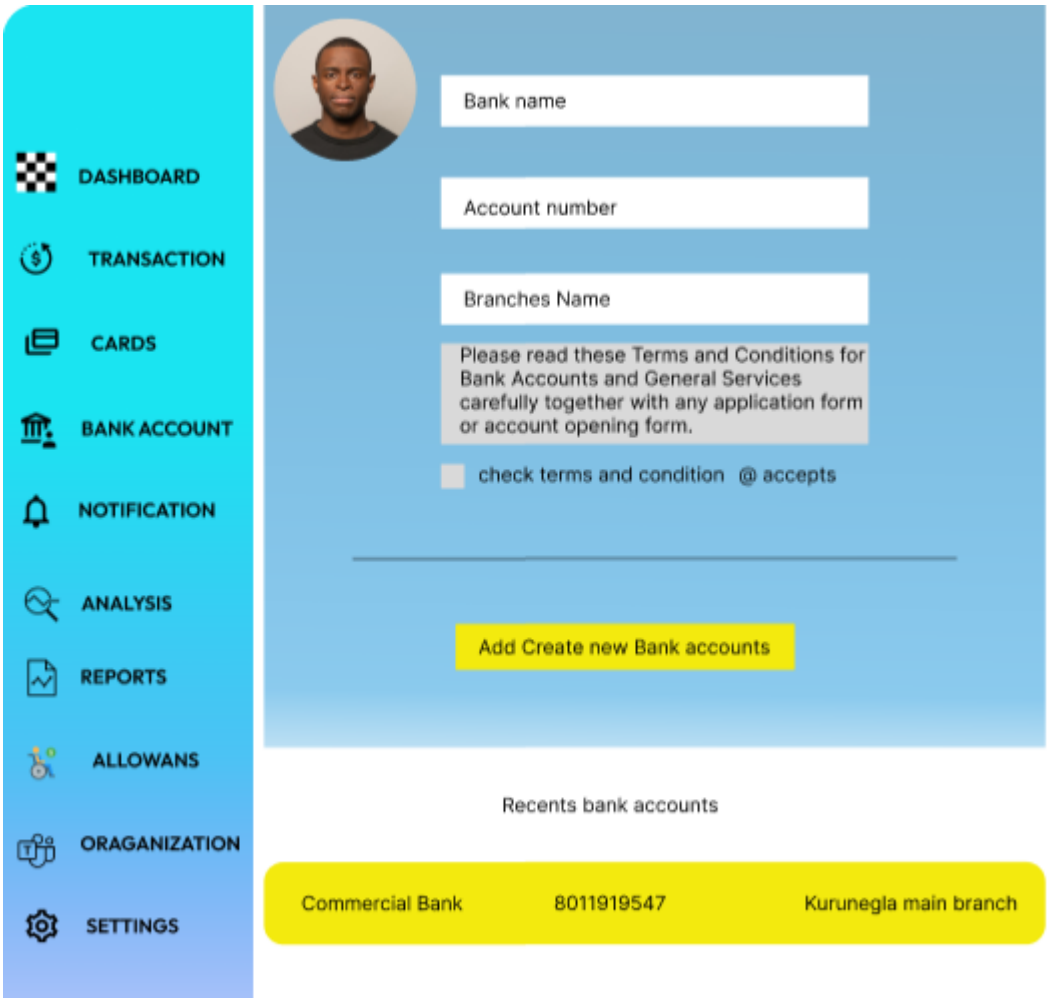

Figure 44:Interface(Create Saving)

6.8 Analysis

Figure 45:Interface(Analysis)

6.9 Create Budget

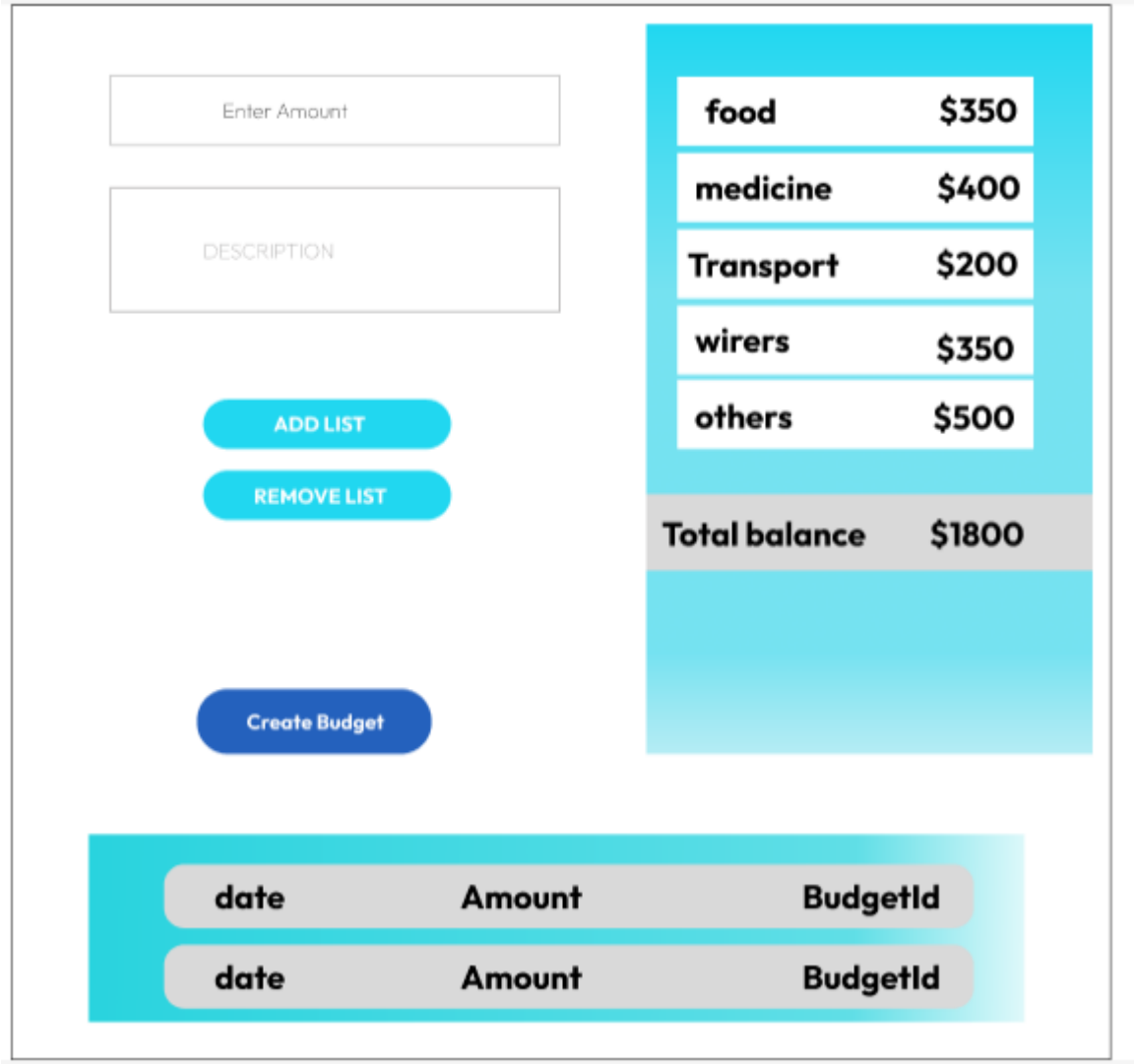

Figure 46:Interface(Create Budget)

6.10 Create Reminder

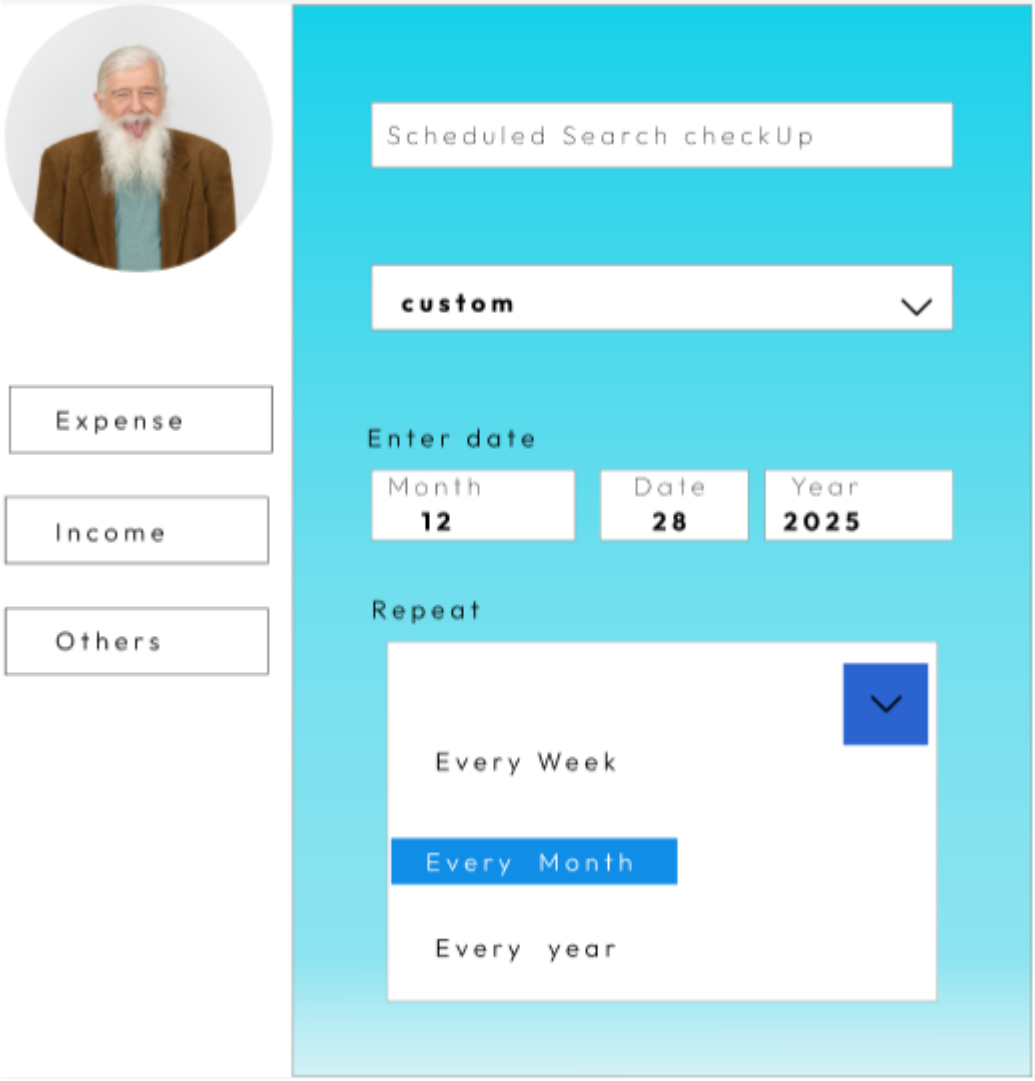

Figure 47:Interface(Create Remainder)

6.11 Send Alert

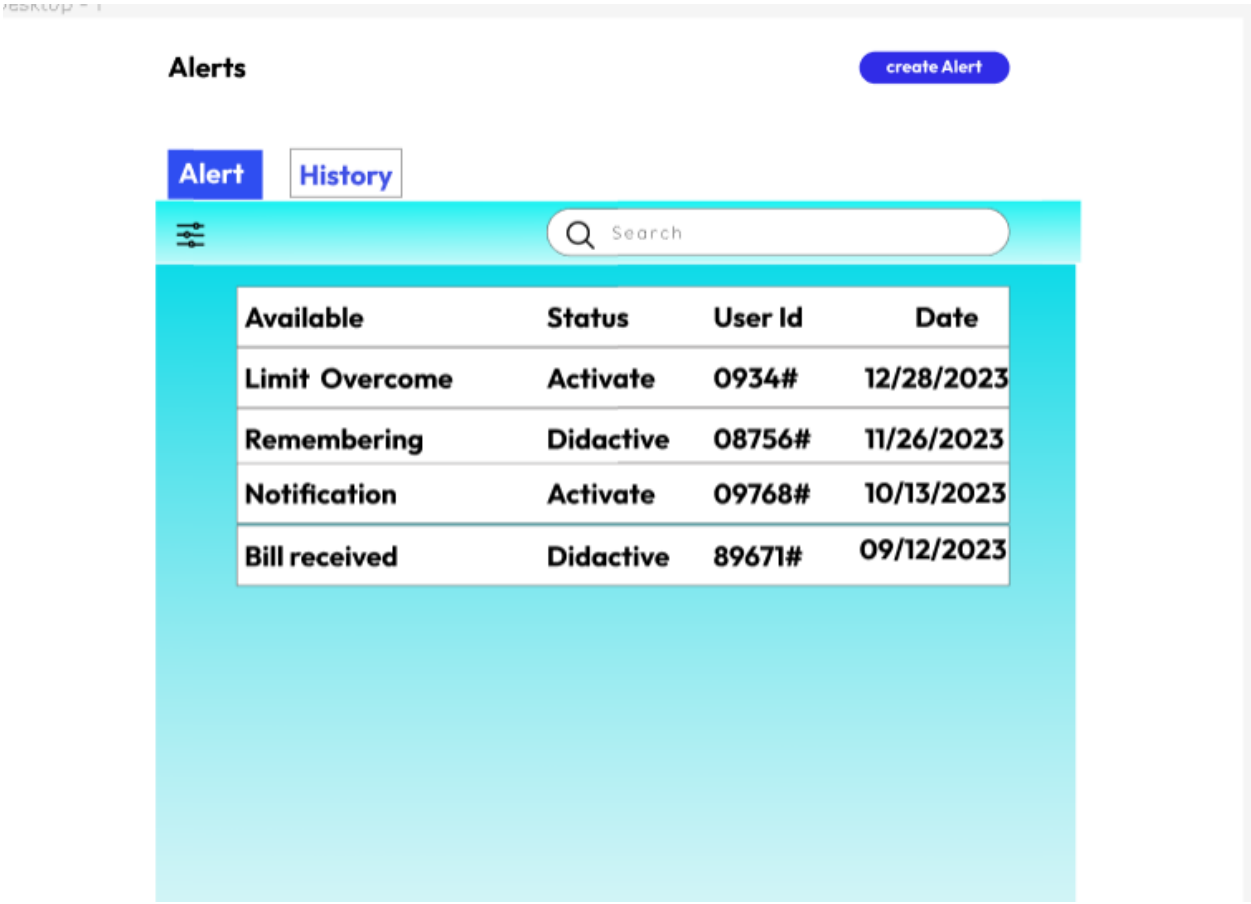

Figure 48:Interface(Send Alert)

6.12 Add Card

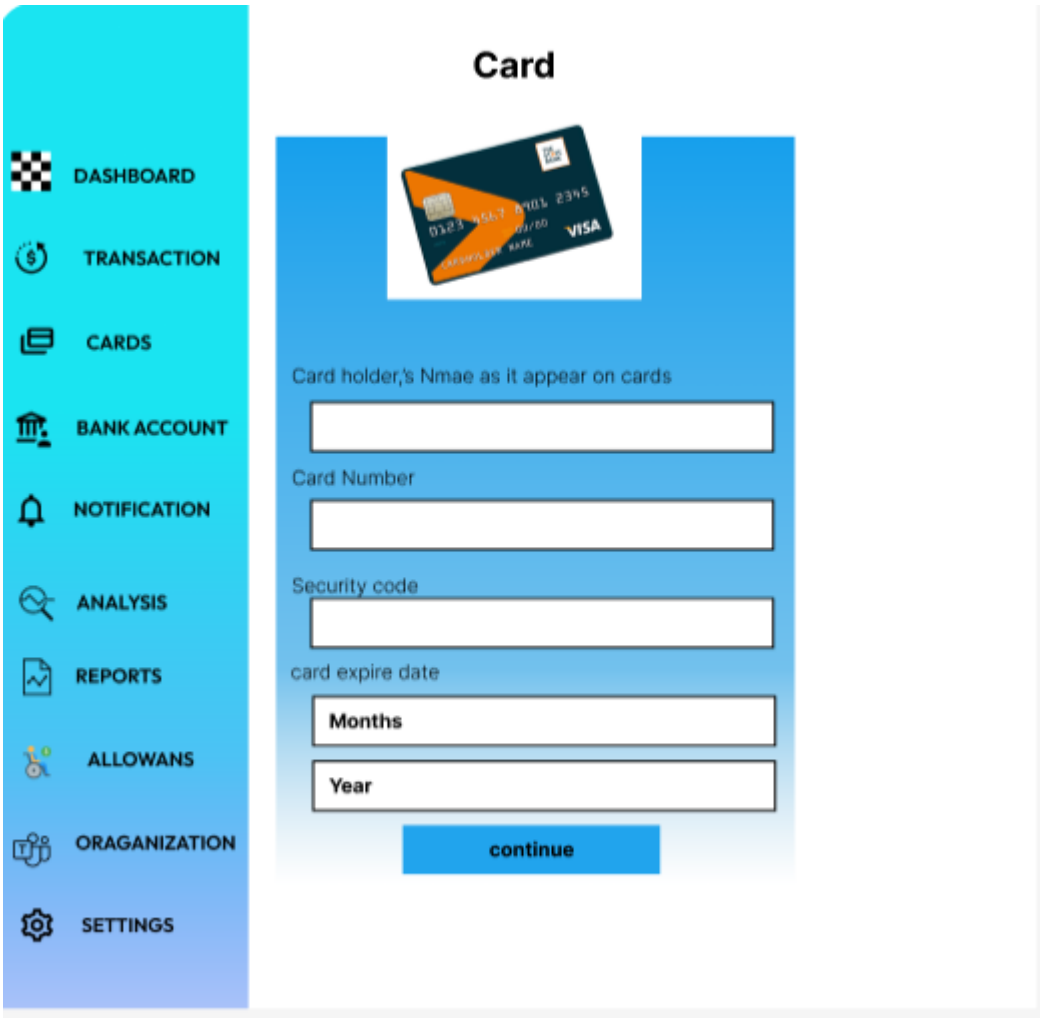

Figure 49:Interface(Add Bank Card)

6.13 View Report

Figure 50:Interface(View Report)

6.14 Setting

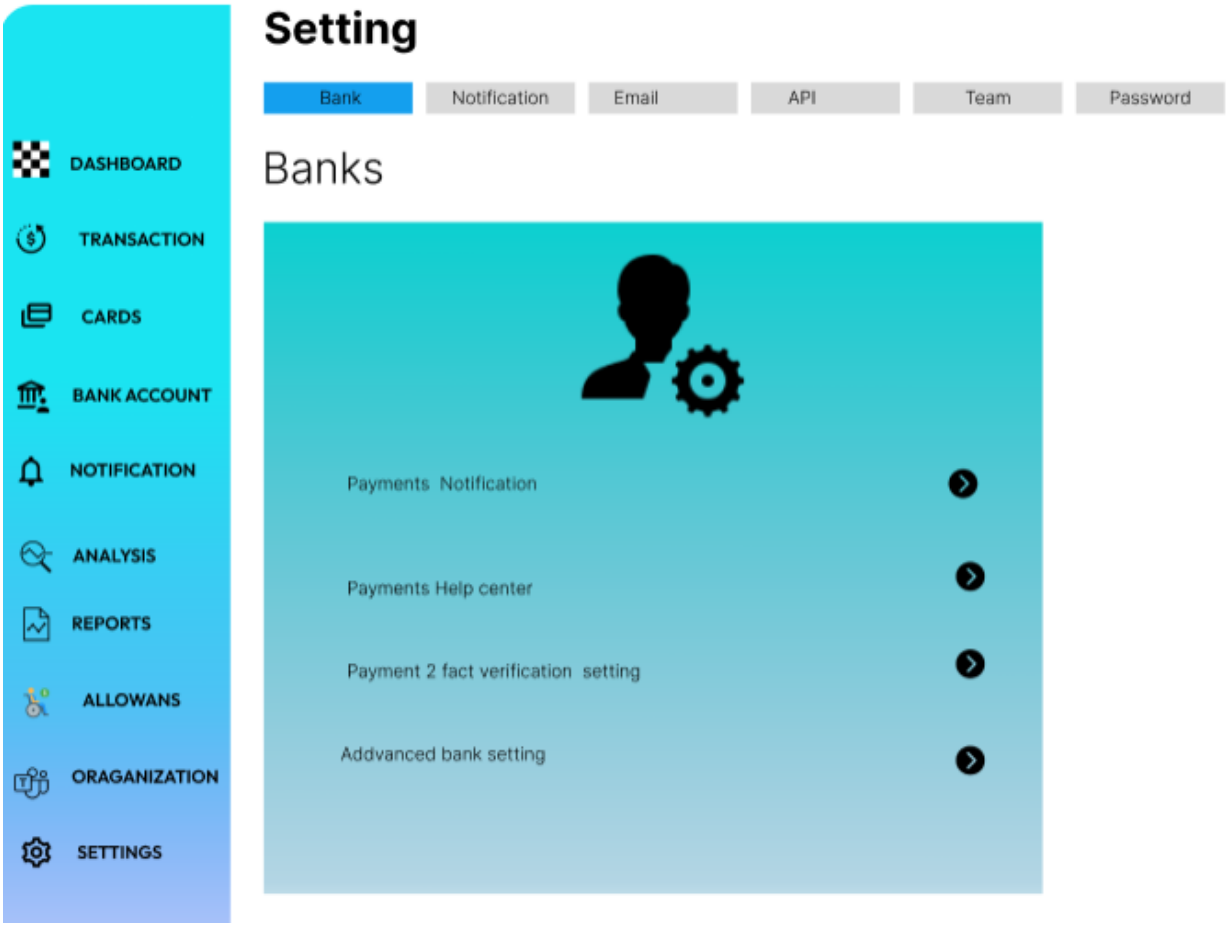

Figure 51:Interface(Setting)

6.15 Create Before Account Overview

Figure 52:Interface(Home)

Chapter 7 Discussion

7.1 Introduction

This chapter briefly discusses the evaluation and testing of our project, how our solution differs from similar works done by others, and the further work of our project.

7.2 Evaluation and Testing

As we are still in the early stage of creating our project, we have not yet conducted any evaluations or tests on any part of it. This means currently we have not assessed the functionality, reliability, or any other performance.

7.3 How our solution differs from similar works done by others

There are many Expense tracker systems available in the market, and most users are using these systems to track their expenses and get ideas about financial requirements. Monefy, Expensify, and Mint are some examples of the most used expense tracking systems. Apart from the common features that can be seen in these existing systems, in our system, there are some additional features like report generation and result dashboard, creating saving plans and budget plans, a simple user interface, uploading receipts, and upcoming payment reminders.

7.4 Further Work

As further work, we must develop the front-end, back-end, and database of our system. For that, we use React JS, .NET, and Microsoft SQL, respectively. Also, we need to integrate the tracking system software used by the company to get real-time updates on the progress. Then we need to create and maintain a form for report generation. Also need to develop upload receipts, notification system, analysis reports, and store expenses and incomes as per the categories. We need to integrate payment reminders and create savings plans and budget plan options. Finally, we build good-performing and secure system.

7.5. Summary

This chapter highlights some key points that were discussed in the previous chapters. That concludes the evaluation and testing and similar systems in the market. And explained the further work that we must do.

Chapter 8 Reference

[1] "ReactJS Tutorial - javatpoint," *www.javatpoint.com*. [Online]. Available: https://www.javatpoint.com/reactjs-tutorial. [Accessed: 30-Dec-2023].

[2] "NET," *Microsoft*. [Online]. Available: https://dotnet.microsoft.com/en-us/. . [Accessed: 30-Dec-2023].

[3] "MS SQL Server Tutorial," *Tutorialspoint.com*. [Online]. Available: https://www.tutorialspoint.com/ms_sql_server/index.htm. [Accessed: 30-Dec-2023].

Appendix A - Individual Contribution

214228B - Wickramasingha W.U.M

The proposed solution is CRM software, and it has many divided modules. Followings are my responsibilities,

- View Transaction
- User dashboard
- Manage user
- Notification system

Transaction view by user and Administrator. Transaction divide by many parts I consider the view Transaction development my role .The responsibilities are accurate Transaction details view user and Administrator . Categories view separate Transaction .display accurate Total Transaction based on ensure the more accurate analysis and other prediction parts because this parts always updated and data inconsistency .

Monitor expense tracker system, set budget limitations, and define conditions for alerts. Receive real-time updates on user expenses. When users exceed budget limits, promptly send alerts with detailed descriptions of overages, providing guidance on corrective actions. Ensure efficient financial management and user compliance with budgetary constraints prowess in full-stack development, ensuring secure access control, seamless user experiences, and efficient data management for optimal performance and reliability

In approaching the UI design for an Expense Tracker System, the focus is on creating an intuitive, visually appealing, and user-centric interface. Here's an individual approach to UI design for the Expense Tracker System.

I contributed to drawing the ER diagram, Use case diagram, and Class diagram. Then I designed Activity diagrams and Sequence diagrams for the above features of our system. According to the design ideas, we started learning UI/UX design principles with Figma. Then I gave my contribution to making a prototype of the UI to show how it.

214101F - Karunarathna W.M.W.G.M.S

In our project, I've been working on different tasks. First, I looked at and assessed a few existing expense tracker systems, and refered and noted the benefits, drawbacks, and areas that may need to improve. I planned the feature's scope after determining the requirements. I was involved in drawing Use cases, class, activity, and sequence diagrams related to view reports, adding expenses, and creating saving pages which improved my comprehension of the system.

I started utilizing the Figma tool to create the UI/UX designs for the features for which I am responsible for the design concepts we as a team decided on.To start my part, I have watched some YouTube videos to study React because we are using React for the front-end development of our system. For the back-end development, we are using the .NET framework. So, I have started studying .NET also, by watching YouTube videos. We are using Microsoft SQL to design and implement the database for our application. So, I have started studying that as well.

Additionally, we are using the MS.NET framework to construct the backend, and I am taking a course on MS.NET to learn and begin developing the project report generation part.

214235T - Wijesinghe W.D.A.C

Our project is to build a complete system of Expense Tracking system which used to track users' financial performance and get a proper idea about their financial requirements.

As planned, I was assigned to handle system login, creating budget plans, and admin dashboard for our project. I contributed to drawing the ER diagram, Use case diagram, and Class diagram. Then I designed Activity diagrams and Sequence diagrams for the above features of our system. According to the design ideas we decided on as a team, we started creating the UI designs for the features of our project. So, I started learning

UI/UX design principles with Figma. Then I gave my contribution to making a prototype of the UI to show how it.

I gave my contribution to creating the interim report and writing the SRS document.

For our system's front-end development, we intended to use ReactJS. I followed some tutorials on YouTube and documents on the web. As well as for backend development we are using .NET and for the databases, we are using MS SQL. I followed some learning materials on websites and YouTube.

214189E - Senarathna G.G.P.C

Firstly, I looked at and assessed a few existing expense tracker systems, and referred to and noted the benefits, drawbacks, and areas that may need to improve.

I planned the feature's scope after determining the requirements.

I was involved in drawing ER, Use cases, class, activity, and sequence diagrams related to view reports, adding expenses, and creating saving pages which improved my comprehension of the system. After, I started interface design with the Figma tool to create the UI/UX designs for the features for which I am responsible for the design concepts we as a team decided on.

Before starting my part, I watched some YouTube videos and resources to study React because we are using React for the front-end development of our system. For the back-end development, we are using the .NET framework. So, I have started studying .NET also, by watching YouTube videos. We are using Microsoft SQL to design and implement the database for our application. So, I have started studying that as well.

214162P - Priyashan H.M.L.S

I am responsible for add income, download report, and upload receipts to the project.

Our work began with the designing of UML diagrams. I was involved in designing the use case diagrams and class diagrams. And I designed the activity diagrams and sequence diagrams pertaining to my part.

I learned some technologies using some existing software. For this functionality, I design UI for the frontend design using Figma. I used some YouTube videos to get a basic knowledge to develop the functionality.

For web development, I started to study React tutorials on YouTube. Because we planned to use React JS as front-end technology. Other than that I also started to study ASP.net as our backend technology using related documents and attractive YouTube videos and also I started refreshing my knowledge of CSS, HTML, and JS. For the databases, we are using MS SQL. I followed some tutorials on websites and YouTube.

Appendix B – Action Plan

Task	2023				2024				
	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May
Learning about Technologies									
Requirements Analyzing									
Database Design									
UI Design									
Database Development									
Build the Solution									
Testing									
Implementation									