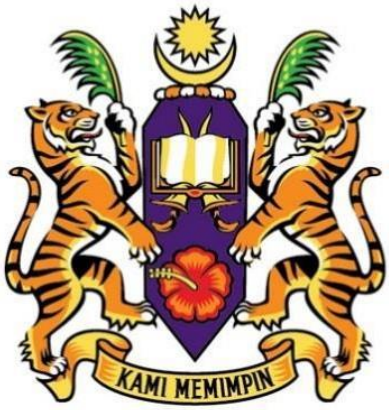


UNIVERSITI SAINS MALAYSIA

SCHOOL OF COMPUTER SCIENCES
UNIVERSITI SAINS MALAYSIA

Software Requirement Analysis and Modeling

Documenting Requirements

Project title:

Smart Life Manager System

Lecturer: Dr. Wan Mohd Nazmee Wan Zainon

Name
Adel Idris Gumma Aljeelani Bukseisa
Muaadh Esmail Saif Abdulkhaleq

Abstract

Numerous issues must be resolved during life. Each day, people do a variety of tasks and are sometimes required to multitask. As a result, many individuals today need something intelligent to help them handle real-world issues, such as reminding them of key occasions, assisting them in becoming their financial planners, and even providing advice on their health. The primary objective of this project is to assist users in managing their time effectively, acting as their financial planner, and managing their health on a regular basis. Our system's primary goal is to offer intelligent advice to users, such as reminders or notification alerts about everyday activities such as events, medicine, and spending limits. Next is designed to organize the user's everyday activities and assist the user in managing their time effectively. The third goal is to lead the user toward a better lifestyle, and the last goal is to aid the user in tracking daily expenditures and assisting them in resolving their own money problems as well as those involving a third party. Our system has just one stakeholder: the user. The user must create an account in order to get a user ID and password for system access. Our system's primary form of interaction is a two-way conversation between the user and the machine. For instance, if the user sets a reminder, the system will inform the user when the date and time arrive. Additionally, the user may learn about their physical condition and expenditure budget via this program.

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed description of the requirements of the Smart Life Manager System. This system handles our daily activities and helps us to overcome incoming problems in our daily life, such as Important dates and times that need to be remembered, focusing on our health, and financial planning. However, this document describes the stakeholders, objectives, and goals of the system. In addition to describing functional requirements and quality requirements this document models the functional requirements with use cases, interaction diagrams, and class models. This document is intended to direct the design and implementation of the target system.

1.2 Overview

The remaining parts of this document give detailed specifications for the Smart Life Manager system. It is organized as follows:

- **Section 2: General Description** This section introduces the list of stakeholders, visions, objectives, and goals of the system. The requirement sources and the elicitation techniques are also introduced in this chapter.
- **Section 3: Specific Requirements** This section categorizes the requirements into functional requirements, quality requirements, other requirements, and constraints. The use cases for the functional requirements are also specified in this chapter.
- **Section 4: Analysis Models** This section includes analysis models for data perspective, functional perspective, and behavioral perspective.
- **Section 5: An appendix** This section contains a survey questionnaire, interview questions, and user interface.

1.3 Glossary - Definitions, Acronyms, and Abbreviations

Term	Definition
Member	Member is a user that has a profile in the system and access to all features provided by the system**.**
Admin	System administrator who is permitted to manage the entire system.
Stakeholder	A stakeholder is a person that has an interest in the successful implementation of the system.
GUI	Graphical User Interface.
UML	Unified Modelling Language.
OS	Operating System.

Table 1: Glossary and Definitions

1.4 References

[1] K. Pohl and C. Rupp, Requirements engineering fundamentals: a study guide for the certified professional for requirements engineering exam, foundation level, IREB compliant, San Rafauel, CA: Rocky Nook, 2015

[2] "IEEE Recommended Practice for Software Requirements Specifications," in IEEE Std 830-1998, 20 Oct. 1998.

2. General Description

After providing an overview of the system, there are a few points regarding the smart life management system that needs to be clarified. This part focuses on clarifying the list of stakeholders, as well as the vision and what we aim to achieve. The requirements sources and the elicitation techniques used to elicit the needs are discussed in the latter portion of this section.

2.1 List of Stakeholders

Stakeholders can be defined as individuals or a group of individuals who have a direct or indirect interest in the company and can be affected by the company's procedures, objectives, and policies. The main stakeholders in the company include shareholders, management, employees, customers, creditors, suppliers, government agencies, and society. Identifying stakeholders is not a one-step process but is done throughout the development life cycle. Furthermore, a stakeholder is a context object, and they will vary throughout the system development life cycle.

Determining the list of stakeholders and identifying needs is very important in the project life cycle. As the non-identification of the stakeholders' needs to vague and incomplete requirements and specifications leads to problems in the system, and it will be costly to identify and solve these problems. There is also saying that the effort required for correction is by a factor of 100 when defects are found in large and critical projects.

End-users, project build teams, and authorities are examples of universal stakeholder types. The system's end-users include direct users, secondary users, and beneficiaries. They can range from users who utilize the system to address specific problems or make their jobs easier to those who are only interested in the system's products.

The project build team is responsible for ensuring that the system manages its scope and remains practical considering stakeholders' demands and expectations. Each member of the project build team is responsible for a distinct aspect of the system's development.

Authorities imply that there are persons who are not involved in the project but have direct authority and access to it. Legal officials and corporate board members are examples of authority. It is also critical to enlist their help in order to preserve consensus throughout the system development process.

Stakeholders of this system are as the table below:

Categories	Stakeholders	Responsibility
End-user	Member	Registered user will be able to see and use all Smart Life Manager features and facilities that system provide.
Project build team	Project Manager	Responsibility for managing and planning projects and supervising their operations, coordinates interactions with users and clients.
	System Administrator	Support and maintain the system to ensure that the system is free from error.
	Developers	Develop the system
Authorities	Board member of Smart Life Manager	Provide guidance and support to the successful implementation of the system.

Table 2: Stakeholder of the System

2.2 Visions

Vision is about an idea or wishes to change the current reality. It is important as it is the starting point for any project or anything. Vision is the basis for making decisions as it states a goal, "WHAT" to do but not the process of achieving it, "HOW?". Vision strongly affects requirements engineering as it guides requirements definition and development of the system.

This vision of the Smart Life Manager System is to provide a system that can handle our daily activities and help us to overcome incoming problems in our daily life in one application. instead of wasting phone memory to download several apps.

2.3 Goals

The Goal of the Smart Life Manager system is to provide one application that combines all other application's functions to become one, so they do not have to download several applications to have several functions in a different application. Besides, some people like to have a calendar function in an app to help them record daily activities, they also want to involve their friends in some events, reminders to remind them of events, medication time, and expenses limits, and payment due dates. They also like to have a health guide to guide their daily meals and want to check their health condition anywhere anytime, therefore, with features above in one application, instead of downloading several apps, our application can meet their expectation in just one system. Therefore, we will build different functions according to each objective, respectively. In addition, one of the goal modeling techniques is using AND/OR Trees. AND/OR consist of nodes which are the goals. The goals of the Smart Life Manager system are:

Goals	
Goal.1	One application combines all other application's functions.
Goal.2	Aim to provide smart guidance for users such as provide reminders or notification alerts to users about their daily activities such as events, medication, and expenses limits.
Goal.3	Aim to manage user's daily activities and help users on managing time wisely.
Goal.4	Aim to help users to remind them of taking the medicine.
Goal.5	Aim to assist users in managing their expenses limits and payment due dates.
Goal.6	Aim to assist users to track daily expenses and help them to solve their money issues and issues involve the third party.

Table 3: Goals of the system.

These goals can be decomposed using AND-decomposition and OR decomposition. OR-dependency means that satisfying one of the sub's goals is enough to satisfy the super-goal. The goal model is drawn and displayed as below:

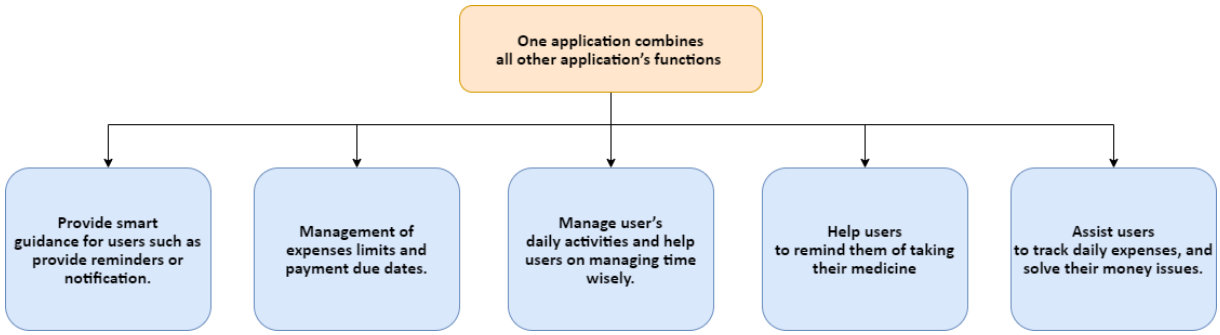


Figure 1: Goals Module.

2.4 Requirement Sources

2.4.1. Stakeholders of The Project

Stakeholders	Description	Name
Member	The end-user of the system.	Ahmed Ali Taki Al-Din Imran bin Mohammed
Project Manager	The stakeholder that manages the development of the system	Adam
System Administrator	The stakeholder who's primary for monitoring and maintenance of the system.	Khaled
Developers	The person who develops the system.	Esmail Saif Rayan Bin Adel
Board member of Smart life Manager	The stakeholder that oversees the best interest of the Smart life Manager	Afifi bin Omar

Table 4: List of stakeholders.

2.4.2. Existing Similar System

There are several other systems out there that do comparable tasks to the one we are working on. We searched the Play Store (Android) and the App Store (IOS) for a few systems and downloaded them to better understand and evaluate them. Simple Calendar by Simple Mobile Tools, Health Pal Fitness - Weight Loss Coach & Pedometer by Digit Grove, and Mobiles: Budget Planner by Mobil slabs are the three solutions that we looked at.

2.5 Elicitation Techniques

The elicitation of requirements is a critical component of requirement engineering. The information acquired during requirement engineering serves as the foundation for requirements elicitation. This knowledge includes the system context of the system to be built, as well as the requirement sources to be examined.

In elicitation, the elicitation method is used to assist the requirements engineer in collecting the stakeholders' requirements and knowledge. There is no uniform elicitation method; rather, the technique is used according to the circumstances, which enables the requirement elicitation process to be tailored to the project's limitations in order to elicit requirements as completely and comprehensibly as possible.

Three elicitation methods are utilized in this project: an interview, a questionnaire, and observation.

2.5.1. Interview

During the interview, the requirement engineer poses prepared questions to stakeholders and documents their responses. It is possible to address any questions that may emerge throughout the interview process. The requirement engineer may uncover subconscious requirements throughout this process by asking clever questions.

In this project, interviews with various stakeholders were conducted by email, phone, and face-to-face. We interviewed users, members of the development team, and stakeholders of the c systems. This is to get a better understanding of the obstacles they encountered. Additionally, by interviewing stakeholders, we can get a comprehensive understanding of the services offered by this platform.

2.5.2. Questionnaire

To extract needs from stakeholders, questionnaires may include open, closed, or both types of questions. Questionnaires may elicit a significant quantity of data in a very short period of time and at a cheap cost. When the answers to questions are predetermined, stakeholders who are unable to communicate their knowledge clearly may nevertheless complete and provide the evaluation.

In this research, questionnaires are sent to key stakeholders, including users, students, and other stakeholders, to ascertain their perspectives on the benefits of having a platform that manages their daily life. Additionally, to elicit feedback on their requirements for the new system that is going to be created.

2.5.3. Observation

The requirement engineer observes and records the operating procedures and processes while he or she is in the same place as the system's users or specialists. These observations serve as the basis for developing the requirements.

Observation is carried out in this research by watching Health Pal Fitness and Budget Planner users while they use these apps to exercise and manage their budget. This approach enables us to ascertain the strengths and weaknesses of the current system and to make improvements based on this knowledge. As a result, a system that benefits all stakeholders may be established.

3. Specific Requirements

3.1 External Interface Requirements

External interface requirements are an essential kind of functional requirement for embedded systems. They describe the page components the end-user will see. They may contain a page list, design components, important stylistic motifs, and even artistic aspects if they are critical to the product's success. They also describe how your product will interact with other elements. We describe all inputs and outputs, including hardware, software, connectivity, and a mockup prototype, in this section.

3.1.1.User Interfaces

When the user enters the Smart life Manager application, the user will be presented with the login-in page. The user needs to enter the email with the password to enter the app. in case of the user does not have an account, the user can create a new account by choosing Sign Up button down.

The Sign-Up interface will appear, the user enters his/her name with the email, then enters the date of birth, age, gender, and enters some personal information such as the expense limit, weight, and height. In the Add Expenses interface, the user can add new expenses, by entering the date with the category, and entering the amount with notes if he wants. The user can add a group of people for the program to calculate the money that is spent by each person separately. In the main interface, there are several options that the user can choose, Time Manager, Health Manager, and Money Manager.

If the user selects Time Manager, the user will be shown a calendar to arrange his day, as well as giving him/her all the details about the events. The user can add new events as much as he/she wants, there is also the option to add an event through the QR scanner. In addition, the user can edit or delete any event stored in the calendar.

if the user has chosen Health Manager from the main interface, three options will appear to the user, Health Calculator, Weight Tracker, Health Consultancy. The user can choose what he/she wants and use other features provided by the application, such as Expense Summary, which clarifies the Expense and gives a graph about it. The application also communicates with notifications on the phone to give an alert for the various events and features that the application has.

Figures of the user interface are in **Appendixes C.**

3.1.2.Hardware Interfaces

The Smart Life Manager application is easy to use and does not have many complications that may require smart devices with supernatural capabilities. The application needs a smart mobile phone, it does not have to be new and has strong capabilities. The application can work on any device that can connect to the Internet and it has a camera so that some features need to use the camera such as QR code, and has a minimum space of 100MB, with a capacity of 1GR. If the

user owns a device with the aforementioned specifications, he will be able to use the program without any problems he may face.

3.1.3. Software Interface

The application requires either an Android or IOs operator system, which will be available in English. The system will need to connect to the Internet because some features need the Internet. The user needs to allow the application to use the camera to use some of the features provided by the application.

3.1.4. Communications Interfaces

Communication is one of the important actions in the application as they depend on each other. Since registering a new user or logging in, you need a connection in the Database to verify the user and allow him to use the application. It is also called (SQL) and to connect to SQL the application needs to be connected to the Internet. Also, the app connects with the calendar and time zone to give accurate reminder Notifications created by the user.

3.2 Functional Requirements

3.2.1. Functional Requirement-1ID: FR1

TITLE: Add the events

Description: The Smart life Manager app provides a calendar for the user to add events.

3.2.2. Functional Requirement-2ID: FR2

TITLE: Update the events

Description: The Smart life Manager app provides the user to update an existing event in the calendar.

3.2.3. Functional Requirement-3ID: FR3

TITLE: Delete the events

Description: The Smart life Manager app provides the user to delete an existing event in the calendar.

3.2.4. Functional Requirement-4ID: FR4

TITLE: Add the Reminders

Description: The Smart life Manager app provides the user to set reminders for the events that they set. The system will notify the user when the event date or time is near.

3.2.5. Functional Requirement-

5ID: FR5
TITLE: Update the Reminders
Description: The Smart life Manager app provides the user to update existing reminders according to their preferences. The system will notify the user when the event date or time is near

3.2.6. Functional Requirement-

6ID: FR6
TITLE: Delete the Reminders
Description: The Smart life Manager app provides users to Delete existing reminders. As much as they want.

3.2.7. Functional Requirement-

7ID: FR7
TITLE: Scan QR Code
Description: The Smart life Manager app provides users to scan the QR code that is provided by the event and set it into the calendar.

3.2.8. Functional Requirement-

8ID: FR8
TITLE: Online Invitation
Description: After the users create a new event in the calendar. The Smart life Manager app provides users to create online invitations to invite friends and family to that event.

3.2.9. Functional Requirement-

10ID: FR10
TITLE: Health calculator
Description: The user can use the health calculator to calculate their BMI, Daily Calories, Water requirement calculator, Blood volume, Body Fat Calculator, Ideal Weight calculator, food nutrition content list, and smoking cost.

3.2.10. Functional Requirement-10

ID: FR10
TITLE: Add the Reminder
Description: The Smart life Manager app provides the user to set reminders for their medication, food, and water intake. The system will notify the user through phone notification.

3.2.11. Functional Requirement-11

ID: FR11

TITLE: Update the Reminder

Description: The Smart life Manager app provides the user to update reminders for their medication, food, and water intake. The system will notify the user through phone notification.

3.2.12. Functional Requirement-12

ID: FR12

TITLE: Delete the Reminder

Description: The Smart life Manager app provides the user to Delete reminders for their medication, food, and water intake.

3.2.13. Functional Requirement-13

ID: FR13

TITLE: Track Daily Expenses

Description The Smart life Manager app will track their expenses and notify them if they use over the expenses limit that they set.

3.2.14. Functional Requirement-14

ID: FR14

TITLE: Record Daily Expenses

Description: The Smart life Manager app will record their expenses based on categories.

3.2.15. Functional Requirement-15

ID: FR15

TITLE: Group Module

Description: The Smart life Manager app will Record the expenses for a group of people and calculate the money spent for each person.

3.2.16. Functional Requirement-16

ID: FR16

TITLE: Add the Reminder

Description: The Smart life Manager app provides the user to add reminders for the important payment and the system will notify them through phone notification.

3.2.17. Functional Requirement-17

ID: FR17

TITLE: Update the Reminder

Description: The Smart life Manager app provides the user with Update reminders for the important payment and the system will notify them through phone notification.

3.2.18. Functional Requirement-18

ID: FR18

TITLE: Delete the Reminder

Description: The Smart life Manager app provides the user to Delete reminders for the important payment.

3.2.19. Functional Requirement-19

ID: FR19

TITLE: Expense summary

Description: The Smart life Manager app provides the user to see the Expense summary.

3.3 Use Cases

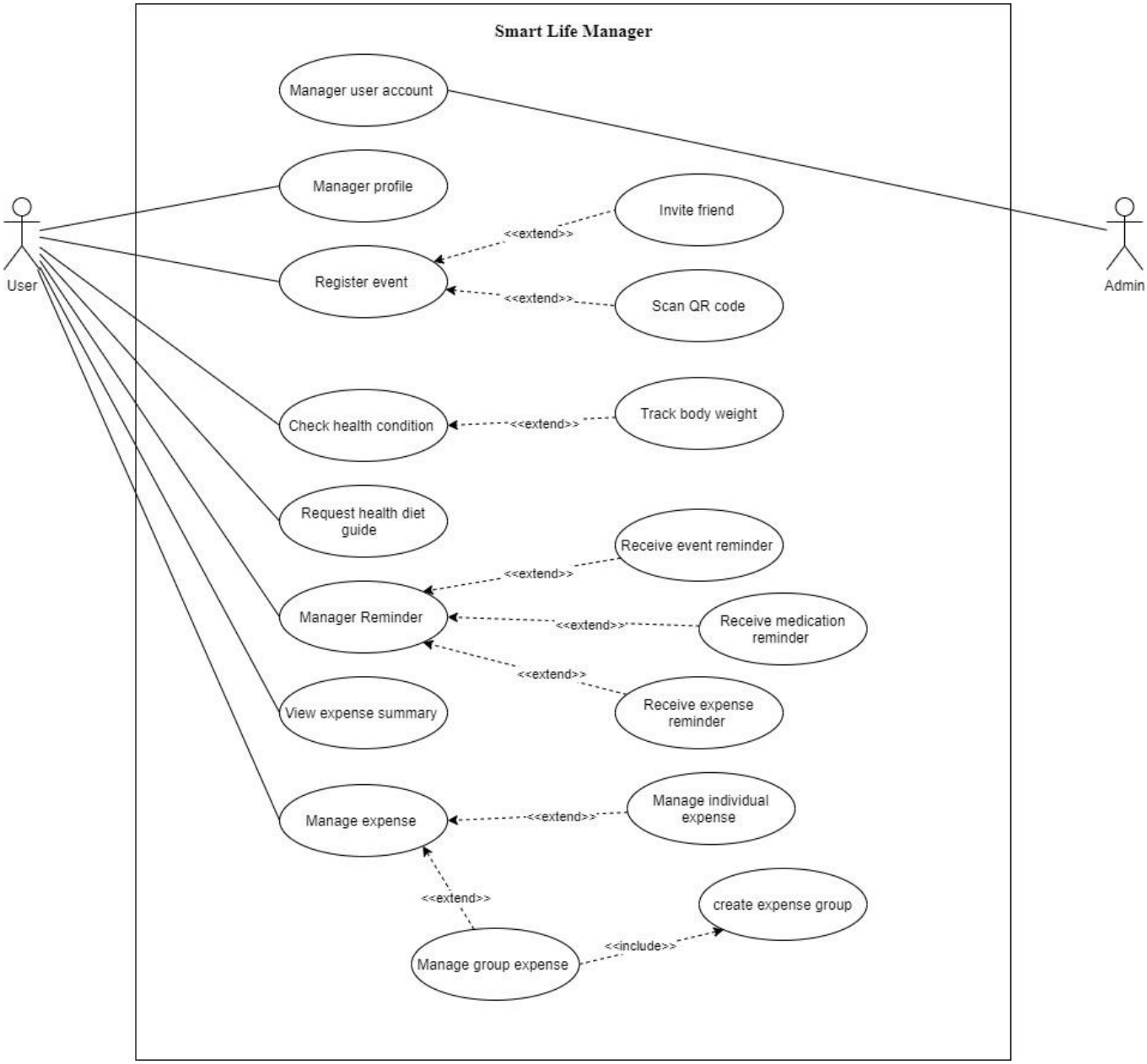


Figure 2: Use case diagram.

3.3.1.Manage user account

No.	Section	Content
1.	Designation	UC-1
2.	Name	Manage user account
3	Authors	Ahmed
4	Triggering event:	Admin wants to add, view, block, or delete user accounts
5	Priority	High
6	Source	Khaled(Admin)
7	Description	Admin can add user accounts to the database. Admin can view the user accounts in the database. Besides that, the admin can also block or delete user accounts
8	Actors	Admin.
9	Pre-conditions	The user must register a profile before the admin can add it into the database The profile must be added to the database before the account can be viewed, blocked, or deleted.
10	Post-conditions	The user's account is added to the database or viewed, blocked, or deleted by the admin.
11	Result	Unable to add user's account into the database. Fail to display user's account Unable to block user's account Unable to delete user's account
12	Main scenario	Admin managing user accounts
13	Alternative scenarios	None.
14	Exception scenarios	None.
15	Qualities	In the Quality Requirements part.

3.3.2.Manage profile

No.	Section	Content
1.	Designation	UC-2
2.	Name	Manage profile
3	Authors	Ahmed
4	Triggering event:	Users want to register, update their own profiles. Users want to delete their own account
5	Priority	High
6	Source	Imran bin Mohammed(User)
7	Description	User fills in their name, email, weight, height and expense limit while registering a profile. User which already has an account can update their own profile (name, email, weight, height, and expense limit) at the profile settings. User can also delete their own account.
8	Actors	User.
9	Pre-conditions	The information must be filled in before the profile can be registered. The account must be added to the database before the user can update their own profile or delete their own account.
10	Post-conditions	The user's account is added to the database or viewed, blocked, or deleted by the admin.
11	Result	Unable to register user's own profile Fail to display user's own profile Unable to Update user's own profile Unable to display account setting Unable to delete user's own account
12	Main scenario	Users managing their own profile and account.
13	Alternative scenarios	None.
14	Exception scenarios	None.
15	Qualities	In the Quality Requirements part.

3.3.3.Register event

No.	Section	Content
1.	Designation	UC-3
2.	Name	Register event
3	Authors	Ahmed
4	Triggering event:	User wants to add, update or delete events. User wants to add or delete an online invitation. User wants to view the weekly to-do list.
5	Priority	High
6	Source	Imran bin Mohammed(User)
7	Description	The user fills in the name, date, time, location, and details of the event in order to add the event to the calendar. Users can also scan QR codes to add an event to the calendar. Users choose an existing event and change its date, time, location, or details. The user picks an existing event to delete it. The user adds or deletes an online invitation of an existing event. User view weekly to-do list
8	Actors	User.
9	Pre-conditions	User must have an account to add events. User must have an existing event to update or delete it. User must have an existing event to create an online invitation
10	Post-conditions	Events are added, updated, or deleted Online invitations are added or deleted Events that are added are displayed by the system
11	Result	Unable to create a new event Fail to display selected event details Fail to update event details Fail to add an online invitation Unable to delete event Fail to display weekly to-do list
12	Main scenario	User managing the events
13	Alternative scenarios	None.
14	Exception scenarios	None.
15	Qualities	In the Quality Requirements part.

3.3.4.Check health condition

No.	Section	Content
1.	Designation	UC-4
2.	Name	Check health condition
3	Authors	Ahmed
4	Triggering event:	When the user clicks in to choose the type of health calculator. When the user clicks into the weight tracker
5	Priority	Medium
6	Source	Imran bin Mohammed(User)
7	Description	There are a few calculators type options for the users to choose from in the system such as BMI, Daily Calories, Water requirement calculator, Blood volume, Body Fat Calculator, and Ideal weight calculator. Each calculator will get different parameters from the user and view different types of results for the user. Users can choose to use each calculator to view a different kind of body condition they want. The weight tracker is to let the user update their weight and a summary of their weight report will view to them when they requested.
8	Actors	User.
9	Pre-conditions	The individual user account must exist
10	Post-conditions	User can view their body condition in a different kind of result.
11	Result	Cannot view the result. 2. Cannot view weight summary report.
12	Main scenario	The user wants to know their body condition.
13	Alternative scenarios	None.
14	Exception scenarios	None.
15	Qualities	In the Quality Requirements part.

3.3.5.Request healthy diet guide

No.	Section	Content
1.	Designation	UC-5
2.	Name	Request healthy diet guide
3	Authors	Ahmed
4	Triggering event:	User faced difficulties in certain diseases or solely want to be more healthy.
5	Priority	Medium
6	Source	Imran bin Mohammed(User)
7	Description	User can choose what problem they faced and the system will show a list of diet plans and solutions about the disease or problem.
8	Actors	User.
9	Pre-conditions	The individual user account must exist.
10	Post-conditions	User can find their related healthy diet plan.
11	Result	Invalid information displayed by the system. The system does not display the diet plan.
12	Main scenario	User searches for a healthy diet plan.
13	Alternative scenarios	None.
14	Exception scenarios	None.
15	Qualities	In the Quality Requirements part.

3.3.6.Manage Reminder

No.	Section	Content
1.	Designation	UC-6
2.	Name	Manage Reminder.
3	Authors	Ahmed
4	Triggering event:	Users want to register, update their own profiles. Users want to delete their own account
5	Priority	Medium
6	Source	Imran bin Mohammed(User)
7	Description	There are few reminders users can make in our system. Event & appointment reminders can be set on the calendar, medication reminders can be set in the health manager, and expenses and loan payment reminders can be set in the expense tracker.
8	Actors	User.
9	Pre-conditions	The individual user account must exist. Events must be set or update in the calendar. Expenses must be recorded in the management expense system
10	Post-conditions	The reminders will notify the user when reaching the date and time.
11	Result	A reminder is unable to be created. Unable to display reminder details. Incorrect reminder details displayed. Changes not saved. The system does not notify the user at the right time.
12	Main scenario	The user wants to set reminders.
13	Alternative scenarios	None.
14	Exception scenarios	None.
15	Qualities	In the Quality Requirements part.

3.3.7.View expense summary

No.	Section	Content
1.	Designation	UC-7
2.	Name	View expense summary
3	Authors	Ahmed
4	Triggering event:	The user wants to view recorded expenses or get a summary of expenses for the current month.
5	Priority	Medium
6	Source	Imran bin Mohammed(User)
7	Description	The user selects an individual or group expense record to display its details. User requests to view the individual expenses summary report of the current month generated by the system.
8	Actors	User.
9	Pre-conditions	Individual accounts must exist. An expense record must exist. An expense group must exist for group expense.
10	Post-conditions	Selected expense records or expenses summary reports must be displayed.
11	Result	Unable to display expense details. Incorrect expense details displayed. Unable to access the records to generate a report. Incorrect data in the report. Unable to display expenses summary report
12	Main scenario	User requests to display expenses details or expense summary of the current month.
13	Alternative scenarios	None.
14	Exception scenarios	None.
15	Qualities	In the Quality Requirements part.

3.3.8.Manage expenses

No.	Section	Content
1.	Designation	UC-8
2.	Name	Manage expenses
3	Authors	Ahmed
4	Triggering event:	The user wants to add, update or delete a record of individual or group expenses.
5	Priority	Medium
6	Source	Imran bin Mohammed(User)
7	Description	Users can select to add, update or delete an individual expense record or a group expense record. An expense group can be created or selected from an existing group. User enter date, category, cost, and descriptions to add a new individual or group expenses record. Users select an existing record and enter the new date, category, cost, and description to update the individual or group expense record. User selects an existing individual or group expense record to delete it
8	Actors	User.
9	Pre-conditions	Individual accounts must exist. An expense group must exist for group expense.
10	Post-conditions	Expense records must be added, updated, or deleted. The expense must be associated with the expense limit recorded in the account and the remainder.
11	Result	Expense record unable to create. Unable to display expense details. Incorrect expense details displayed. Changes not saved.
12	Main scenario	User manages their expenses.
13	Alternative scenarios	None.
14	Exception scenarios	None.
15	Qualities	In the Quality Requirements part.

3.4 Quality Requirements

Specifications for the quality of goods, services, processes, or environments are known as quality requirements. Any factor, physical or intangible, that adds worth to objects beyond their functioning and characteristics is referred to as quality. The quality criteria listed below are only a few examples. A condition or capacity that must be included in a requirement is referred to as a quality requirement. They represent the information required to verify that a project deliverable was completed successfully.

3.4.1.Performance

Performance criteria specify how well a system operates under certain circumstances like reaction time and throughput.

The number of devices that can be supported: The application should be able to handle up to 10,000 concurrent users.

Application generation time: The system app shall be fully loaded and functional after starting up within 5 seconds 99% of the time with an internet speed of 500kb/s.

Response time: The Application shall respond to any input and queries within 3 seconds 99% of the time.

GUI and function: The system GUI and functions shall be simple, convenient, and easy to operate by any type of user.

3.4.2.Reliability

In this system, the quality attributes of the system are performance, confidentiality, reliability, capacity, recoverability, usability, and portability.

Performance specifies the system's properties that allow the system to respond to users in a shorter time and increase user satisfaction. Reliability specifies the duration of the system run without failure in a period.

Confidentiality is an important requirement that must be included in the application to control the accessibility of sensitive data and protect the privacy of the users. Reliability requirements are one of the most important aspects in quality requirements as it defines the reliability of the application or products need to achieve.

The application stores unfinished options, input, or data in the database and easily recovered upon any unexpected termination of the session. The application will not crash upon any invalid input 99% of the time. If there is a fatal error, the system will shut down without damaging or crashing the devices.

3.4.3.Availability

The minimum uptime and performance level of system services are defined by availability criteria.

The application will be accessible 24 hours a day, seven days a week. (With the exception of maintenance or upgrades). also, the application must be linked to the Internet, and users must have access to the internet in order to utilize it. In addition, the user may restart the application at any

moment and continue to use it, but any unsaved data will be deleted. Furthermore, every day, data is automatically backed up.

3.4.4.Security

Security control is to maintain our system operating environment for users and also protect user's private info from various types of security attacks such as hacker attacks and viruses and worms. It should be able to protect information and transactions during transmission across the Internet and other insecure environments from intercepted, destroyed and modified.

Access control allows a specific user to access specific resources. For example, only that particular user can access their private info in this system. A privileged user is the admin and system developer, they can access all user account and also executable code.

3.4.5. Maintainability

Application extendibility

It should be simple to add new features to the application. The code should be designed in such a manner that it is easy to add additional functions. In order for future features to be added to the application with ease.

Applicability testing

To enable testing of the application's many functionalities, test environments should be created for it. In order to put the app through its paces.

The system shall implement specific code, written in a specific way to favor developers to further extend and improve the system. The system shall implement testing applications and software to test and check the server periodically.

3.4.6. Portability

Application portability

The app should be compatible with both iOS and Android. as well as the flexible platform on which the program will operate.

The ability of the system and services to be transferred from one environment to another is described by portability criteria. It also has to do with the ease with which the server may be moved from one hardware and software environment to another. The system shall support and compatible with operating systems such as iOS, Android.

3.5 Constrains

The mobile application is limited by the mobile phone's system interface mechanism. The application's Internet connection is also a limitation. Because the application retrieves data from the database via the Internet, it is essential that the user has access to the Internet.

Smart life Manager financial assistance data may only be modified by administrators and authorized users. In addition, the graphical user interface (GUI) should follow industry standards and have a consistent look across all operating systems. The system must operate on smartphones that use Android or iOS as its operating system. Furthermore, the system must be built to allow for the switching of interfaces and databases without disrupting the operation.

4. Analysis Models

4.1 Requirement Modelling in Data Perspective

4.1.1. UML Class Diagram

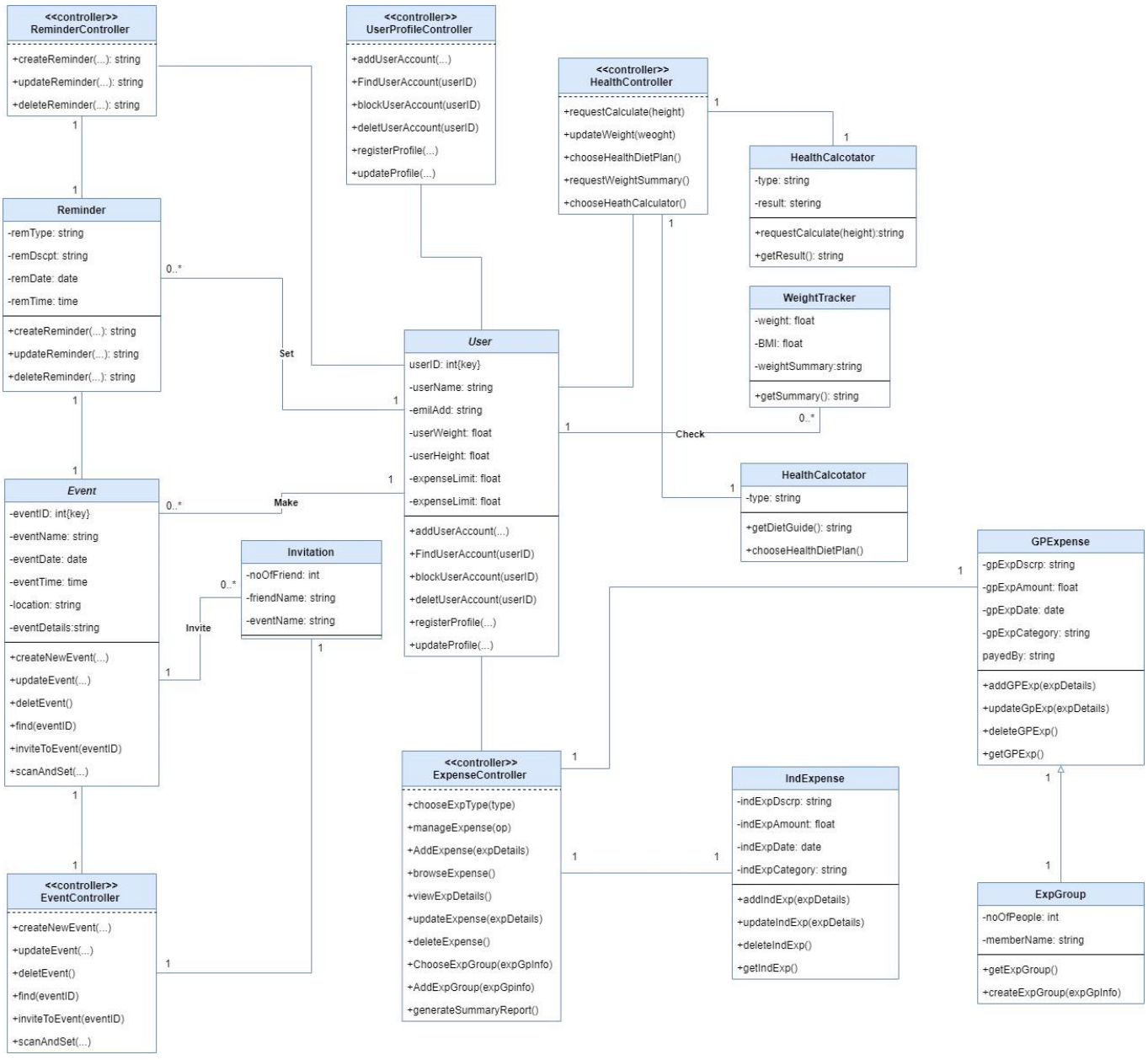


Figure 3: Class Diagram.

4.2 Requirement Modelling in Functional Perspective

4.2.1. Data Flow Diagram

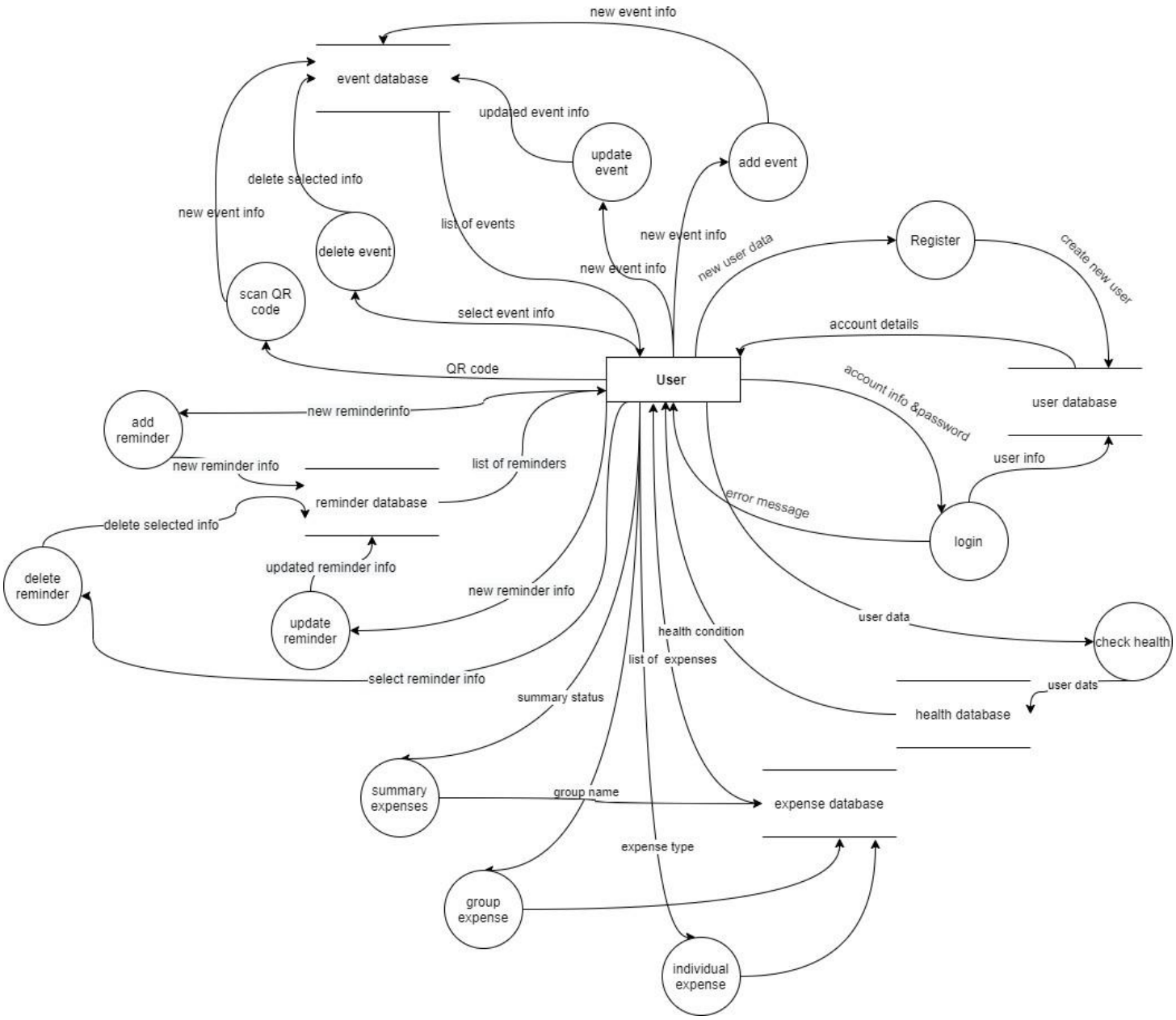


Figure 4: Data flow diagram.

4.2.2.Activity Diagram

4.2.2.1. User login

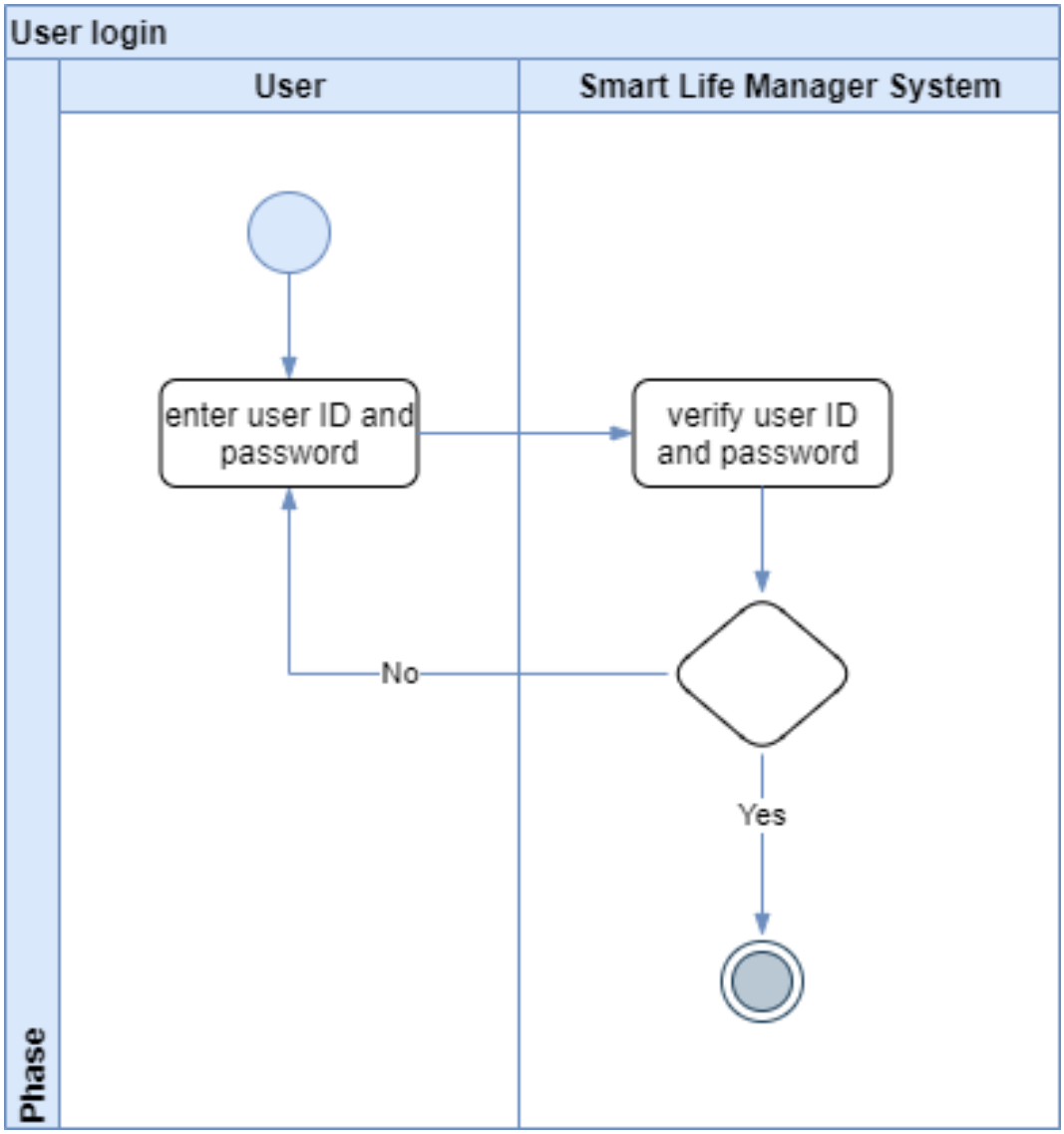


Figure 5: User login activity diagram.

4.2.2.2. Add event

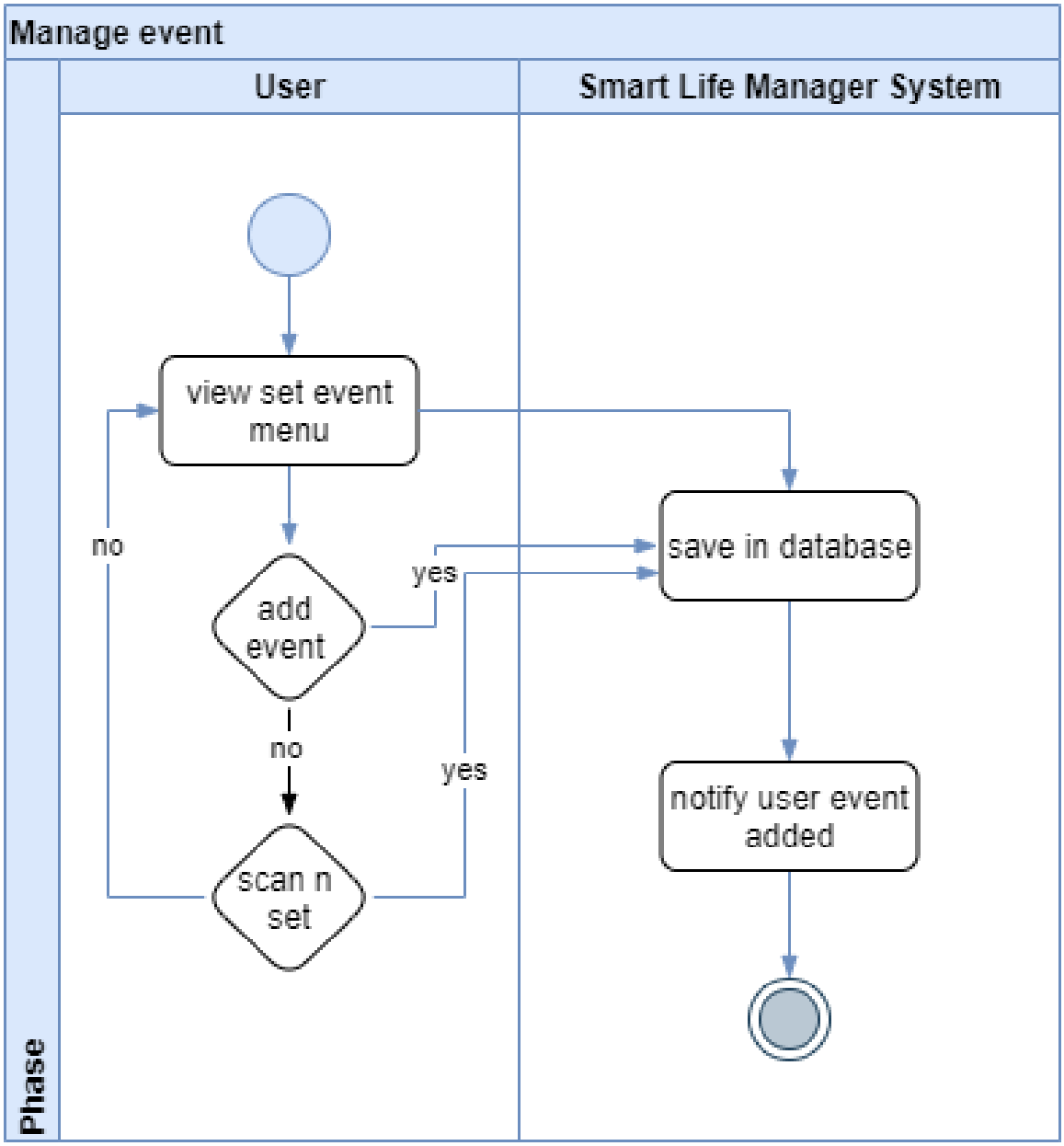


Figure 6: Add event activity diagram.

4.2.2.3. Update event

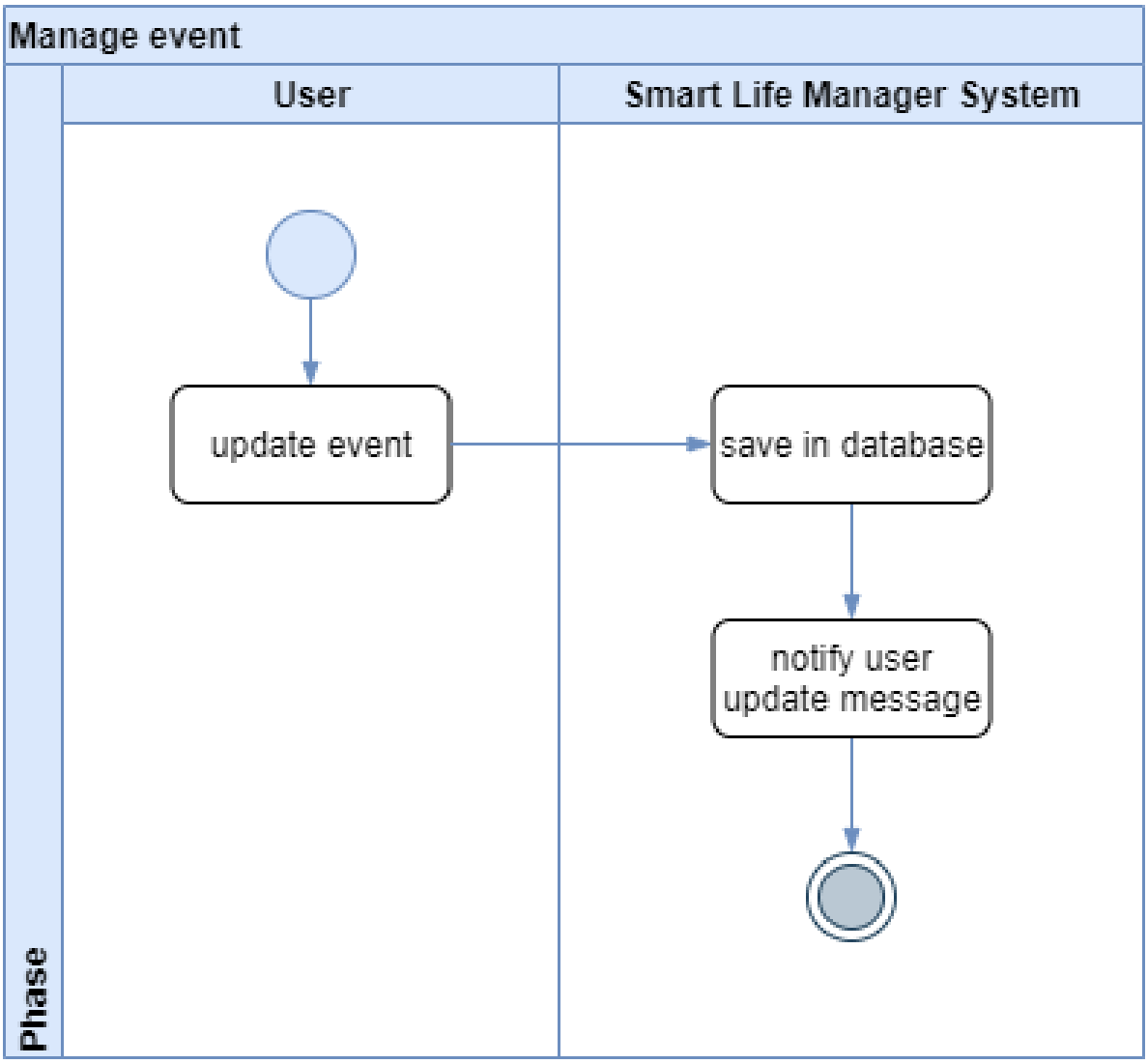


Figure 7: Update event activity diagram.

4.2.2.4. Delete event

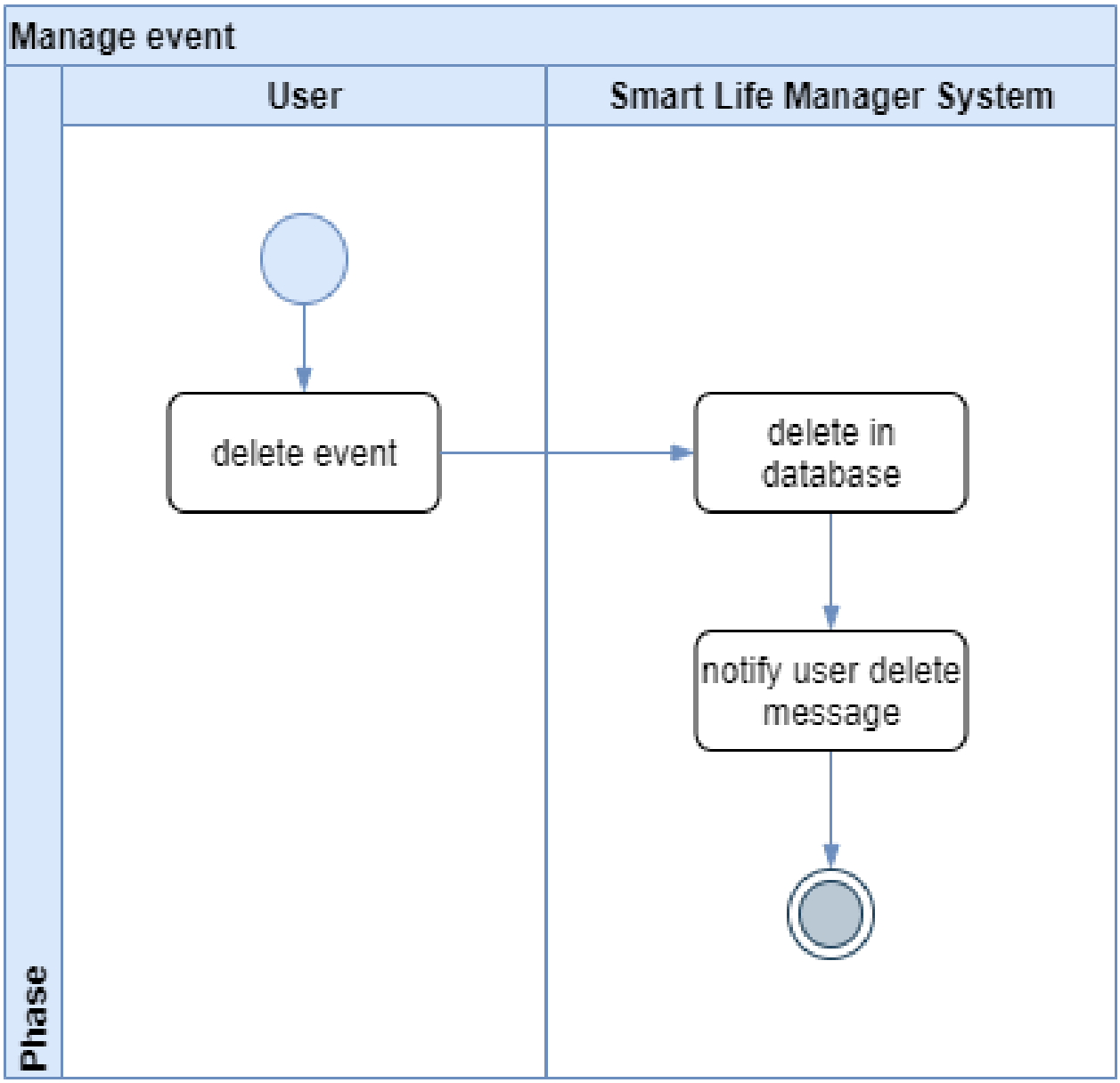


Figure 8: Delete event activity diagram.

4.2.2.5. Add reminder

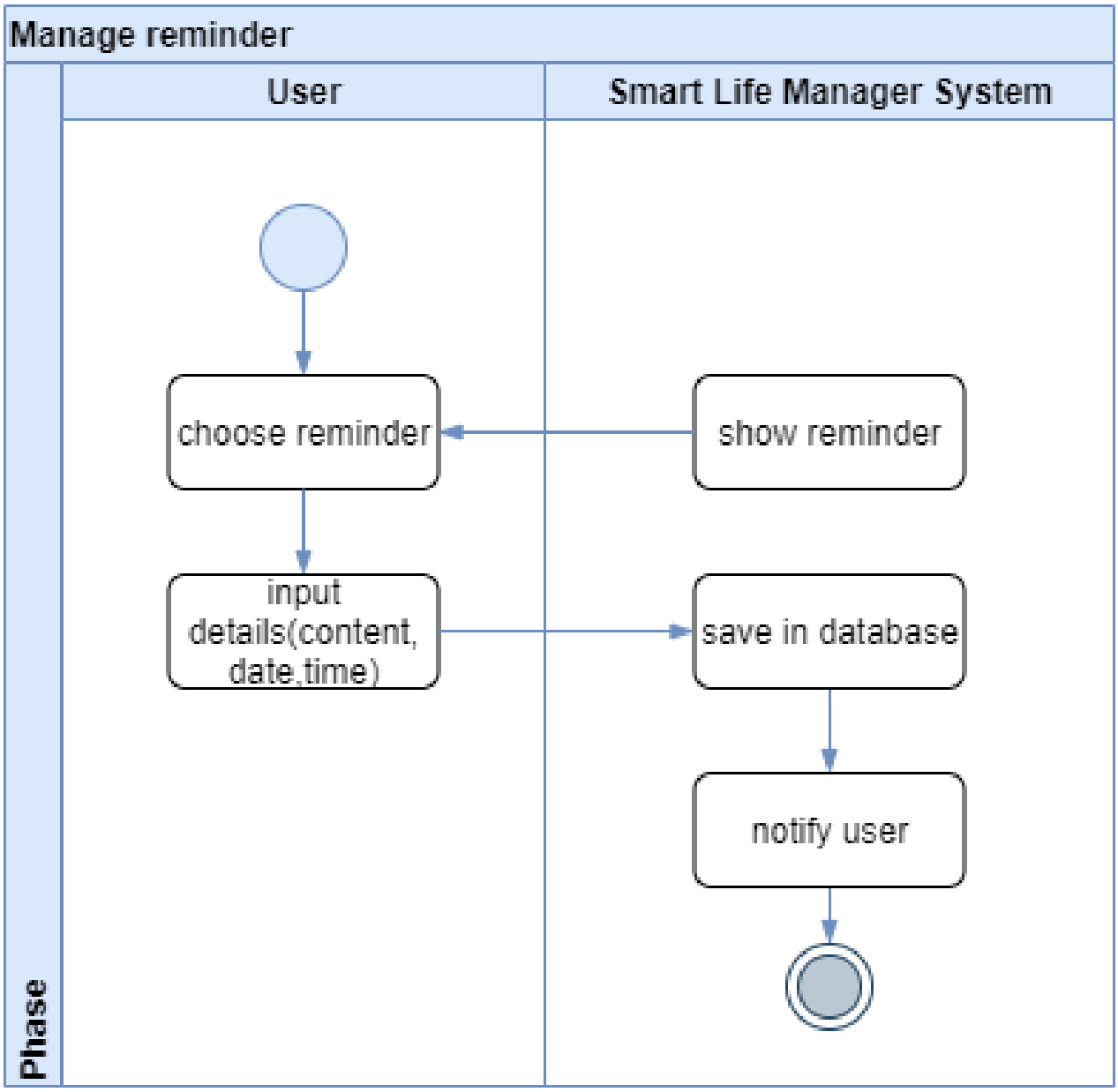


Figure 9: Add reminder activity diagram.

4.2.2.6. Update reminder

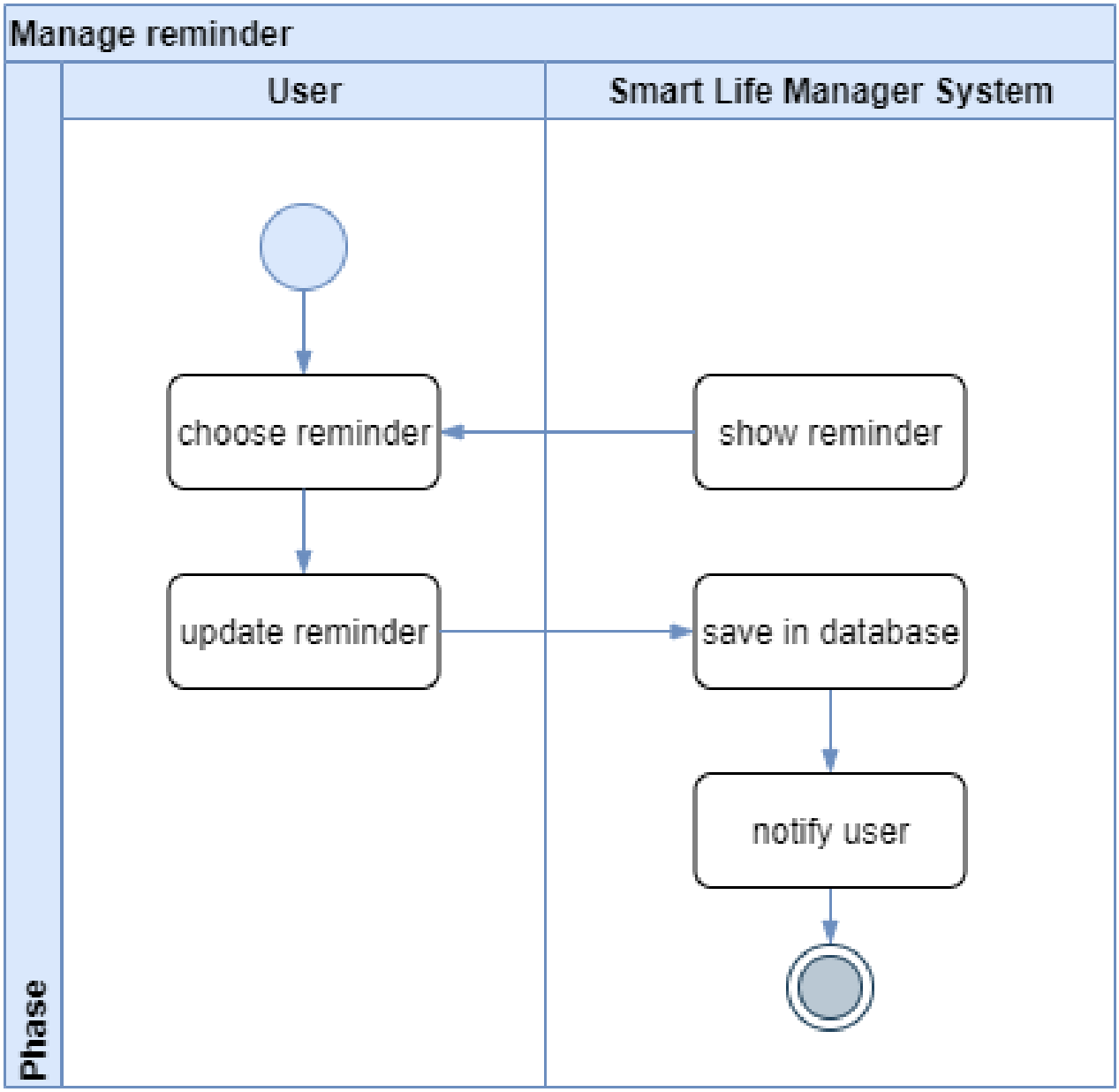


Figure 10: Update reminder activity diagram.

4.2.2.7. Delete reminder

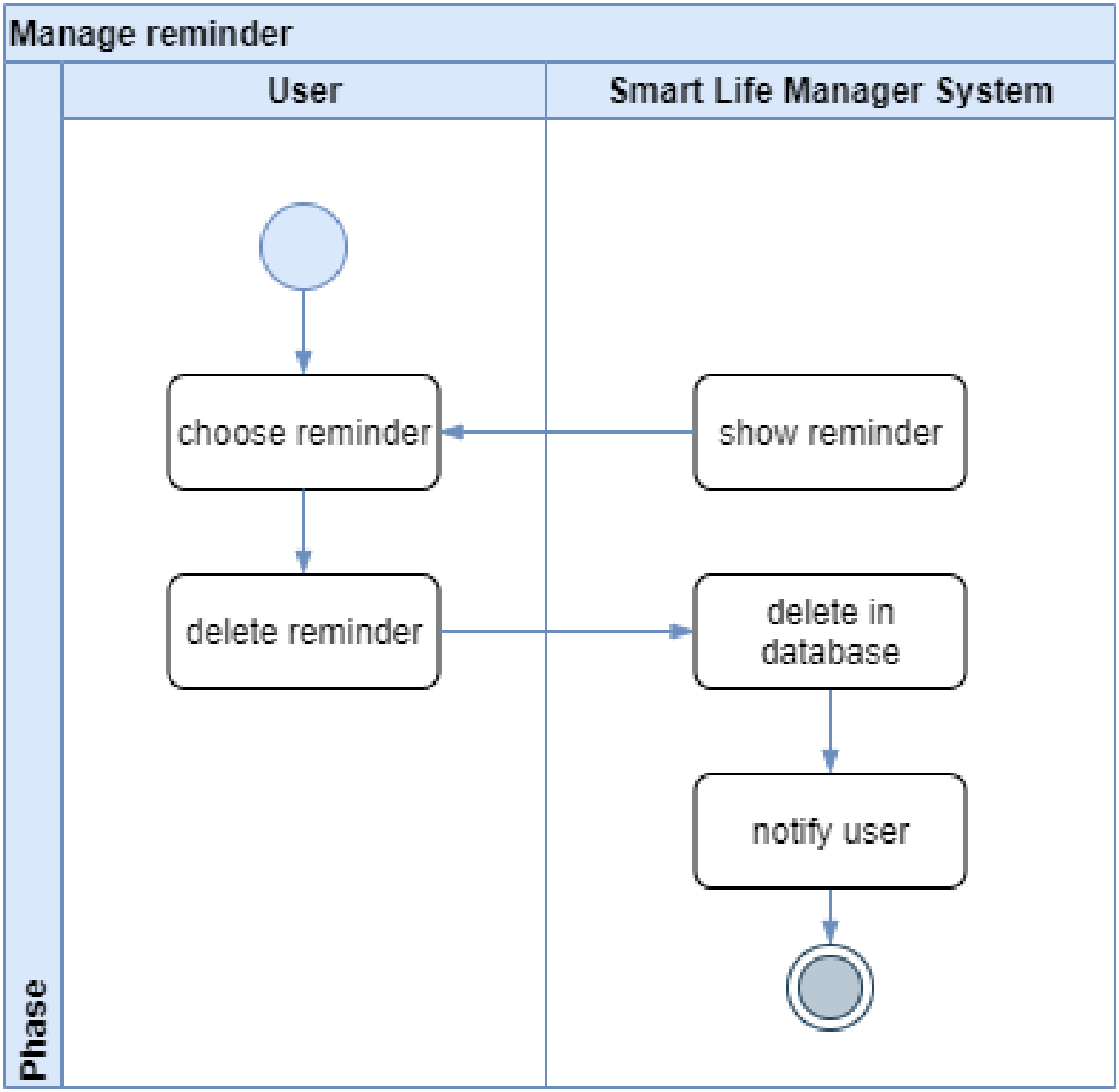


Figure 11: Delete reminder activity diagram.

4.2.2.8. Check health

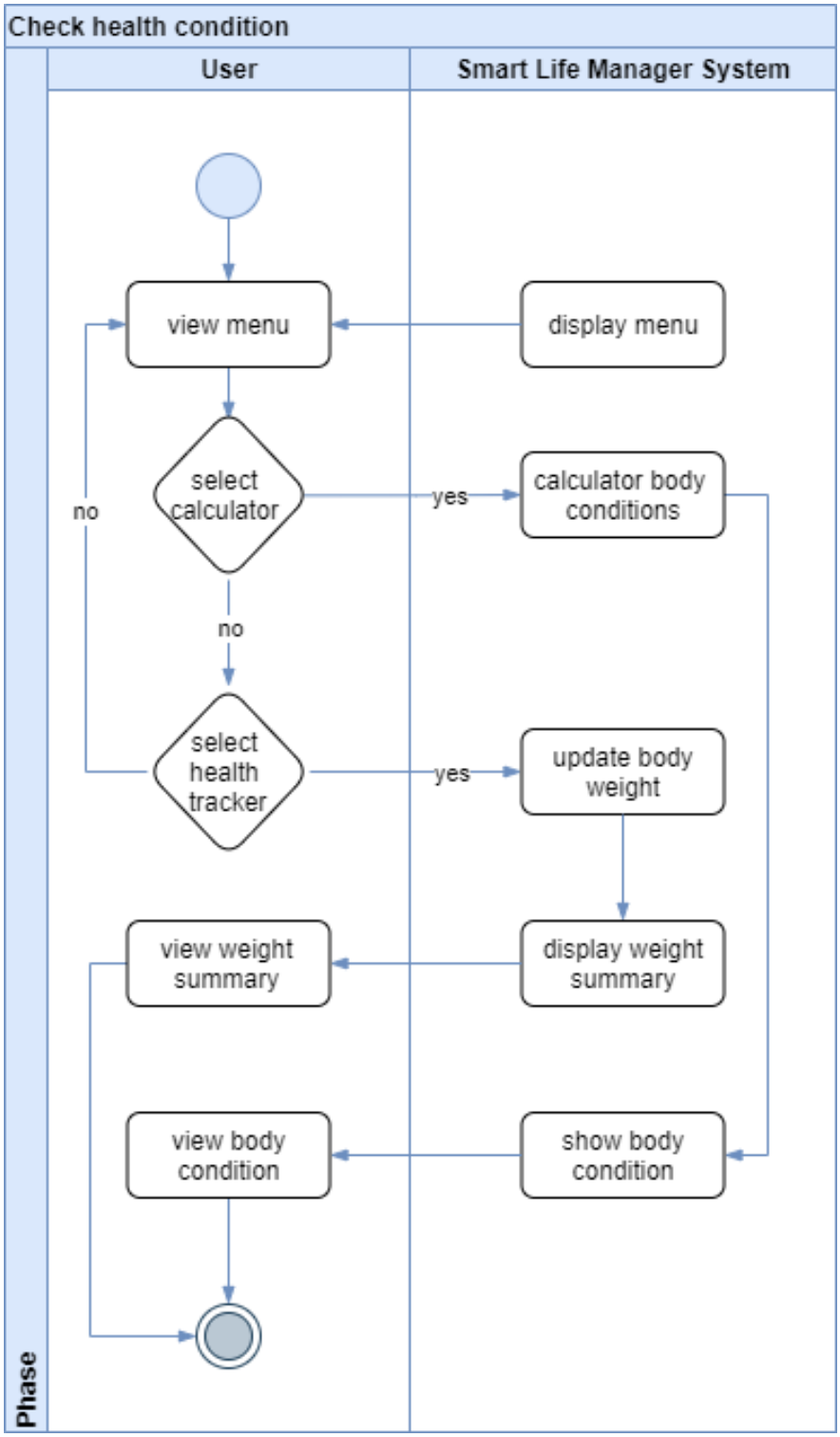


Figure 12: Check health activity diagram.

4.2.2.9. Manage expenses

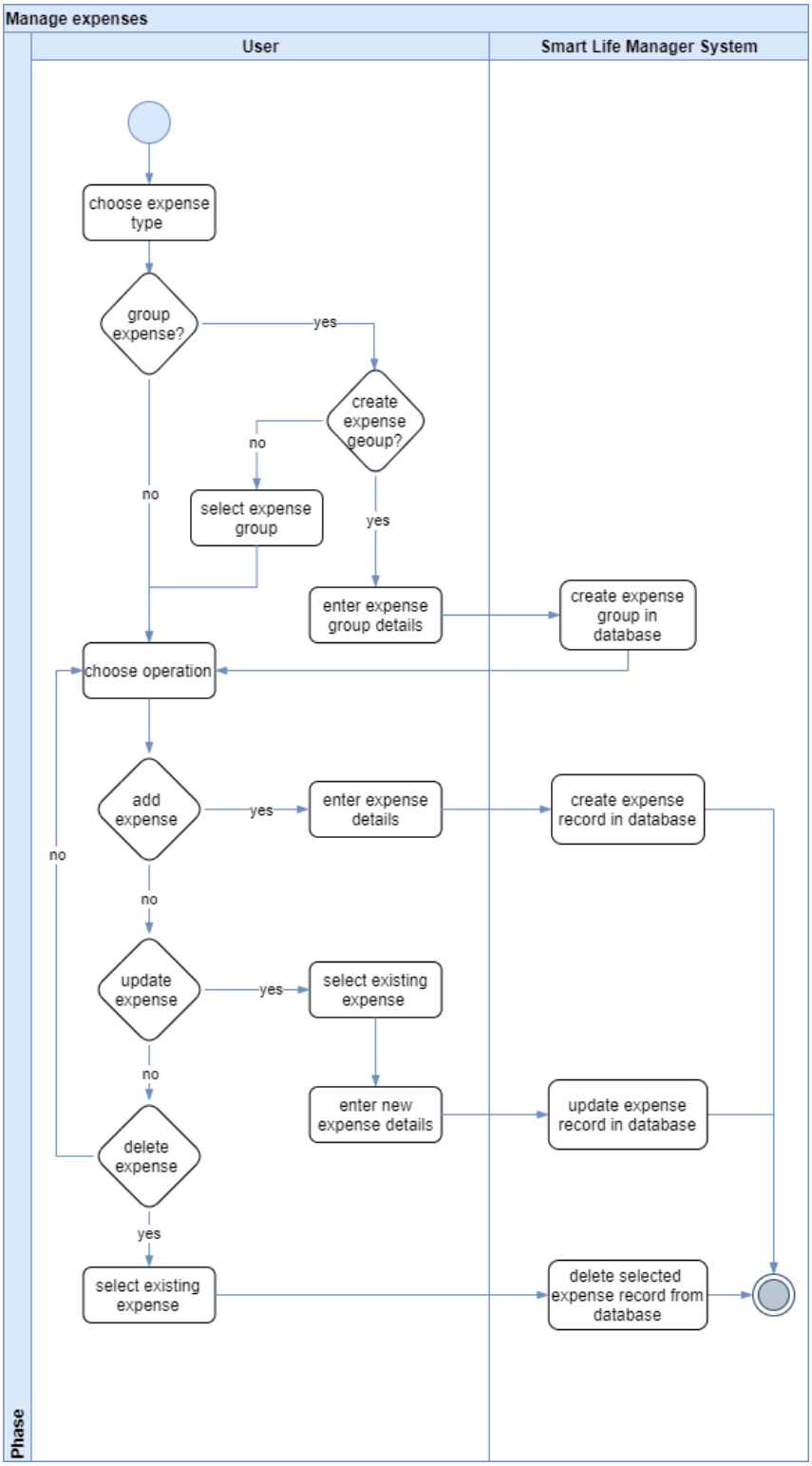


Figure 13: Manage expense activity diagram.

4.3 Requirement Modelling in Behavioral Perspective

4.3.1. Sequence Diagram

4.3.1.1. Manage user account

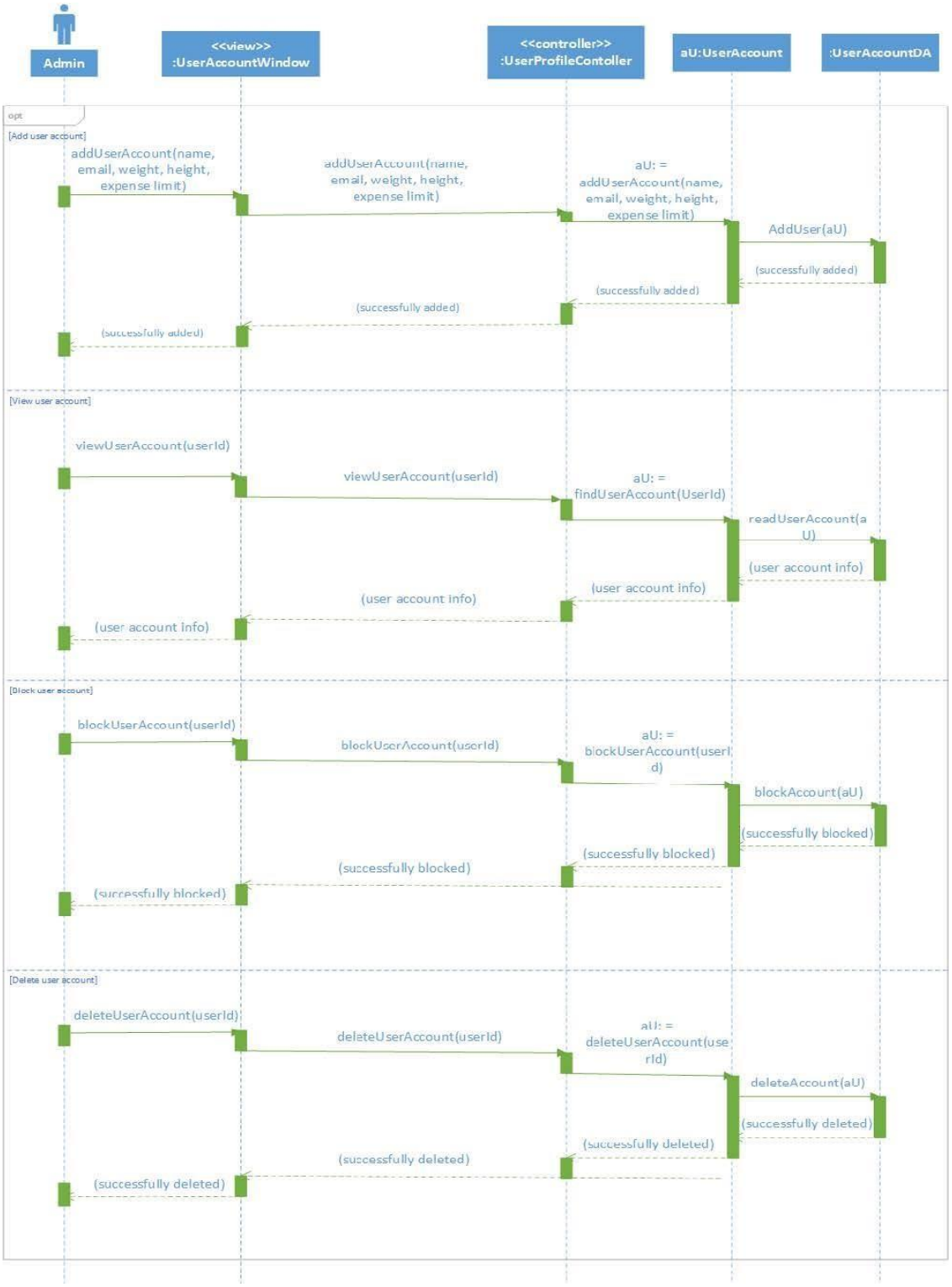


Figure 14: Manage user account sequence diagram.

4.3.1.2. Manage profile

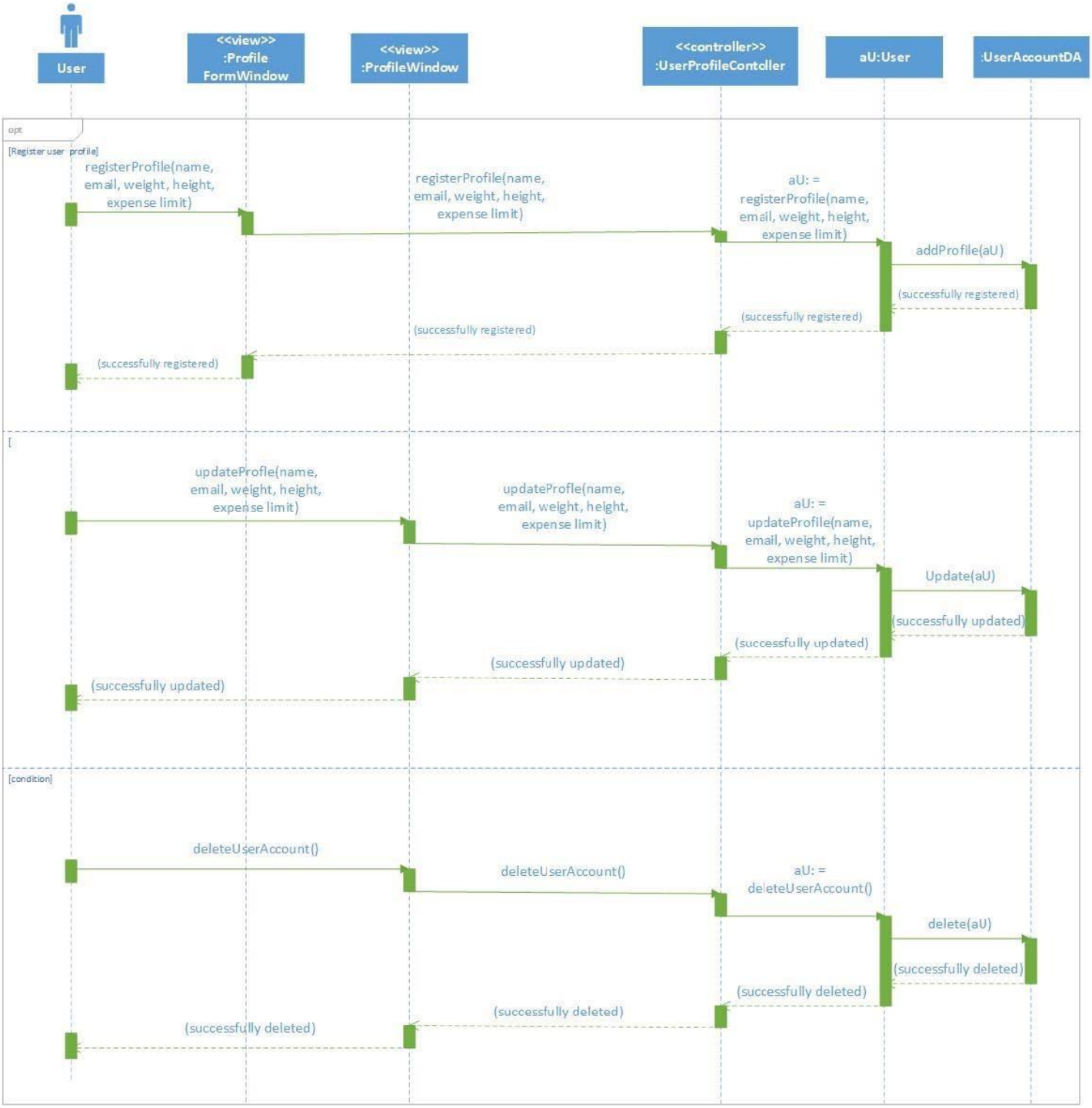


Figure 15: Manage profile sequence diagram.

4.3.1.3. Register event

Figure 16: Register event sequence diagram.

4.3.1.4. Check health condition

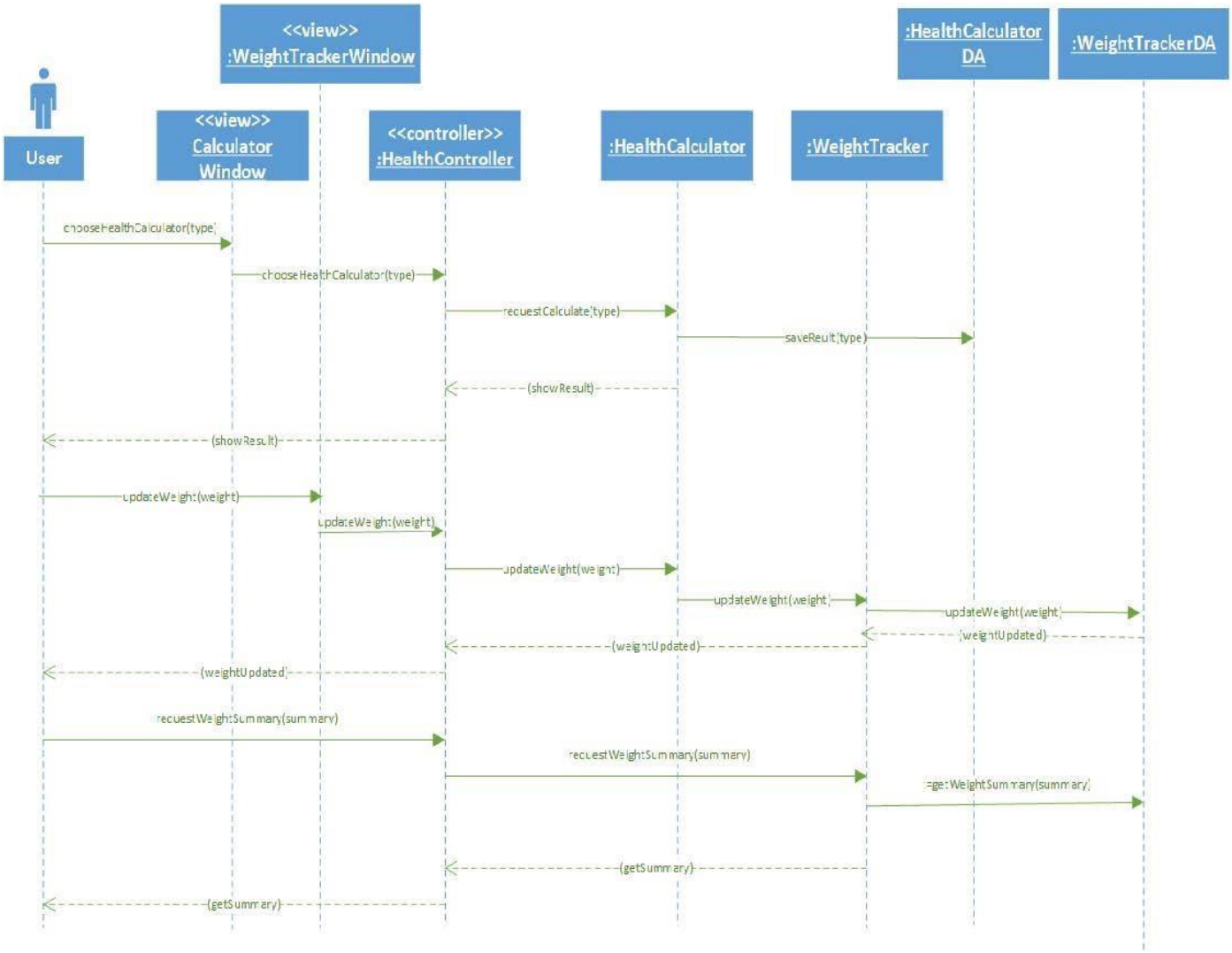


Figure 17: Check health condition sequence diagram.

4.3.1.5. Request healthy diet guide

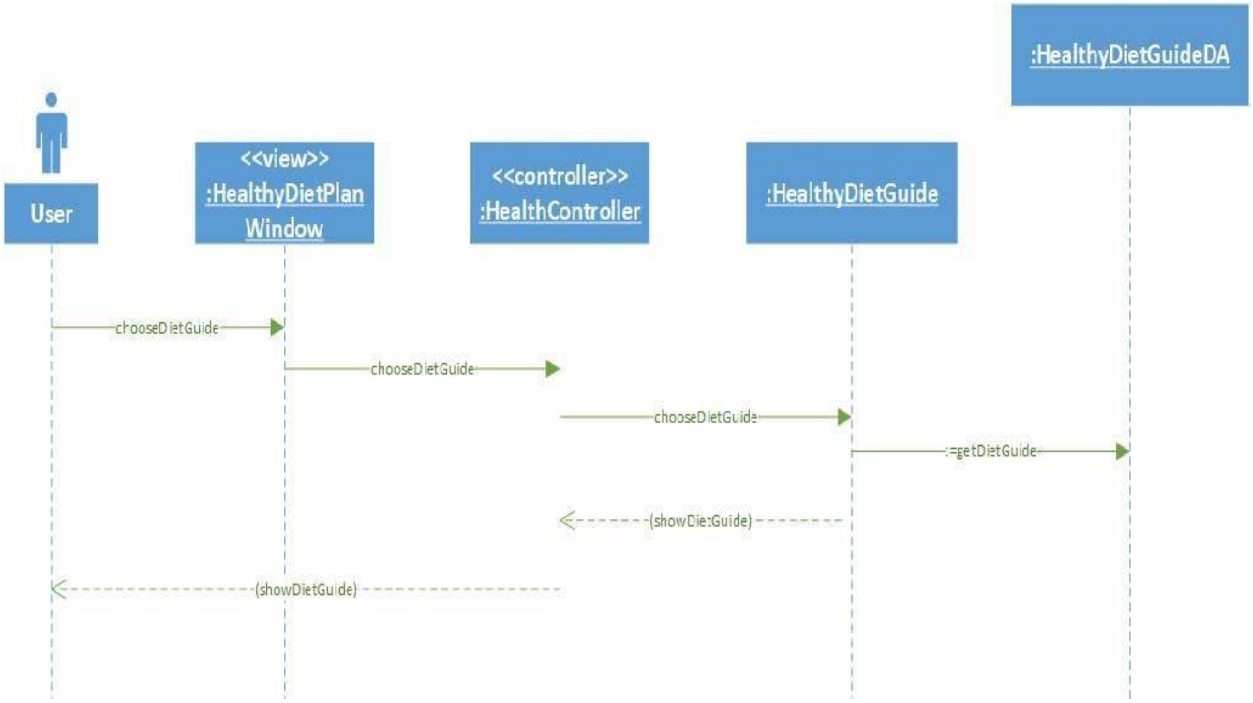


Figure 18: Check health condition sequence diagram.

4.3.1.6. Manage reminder

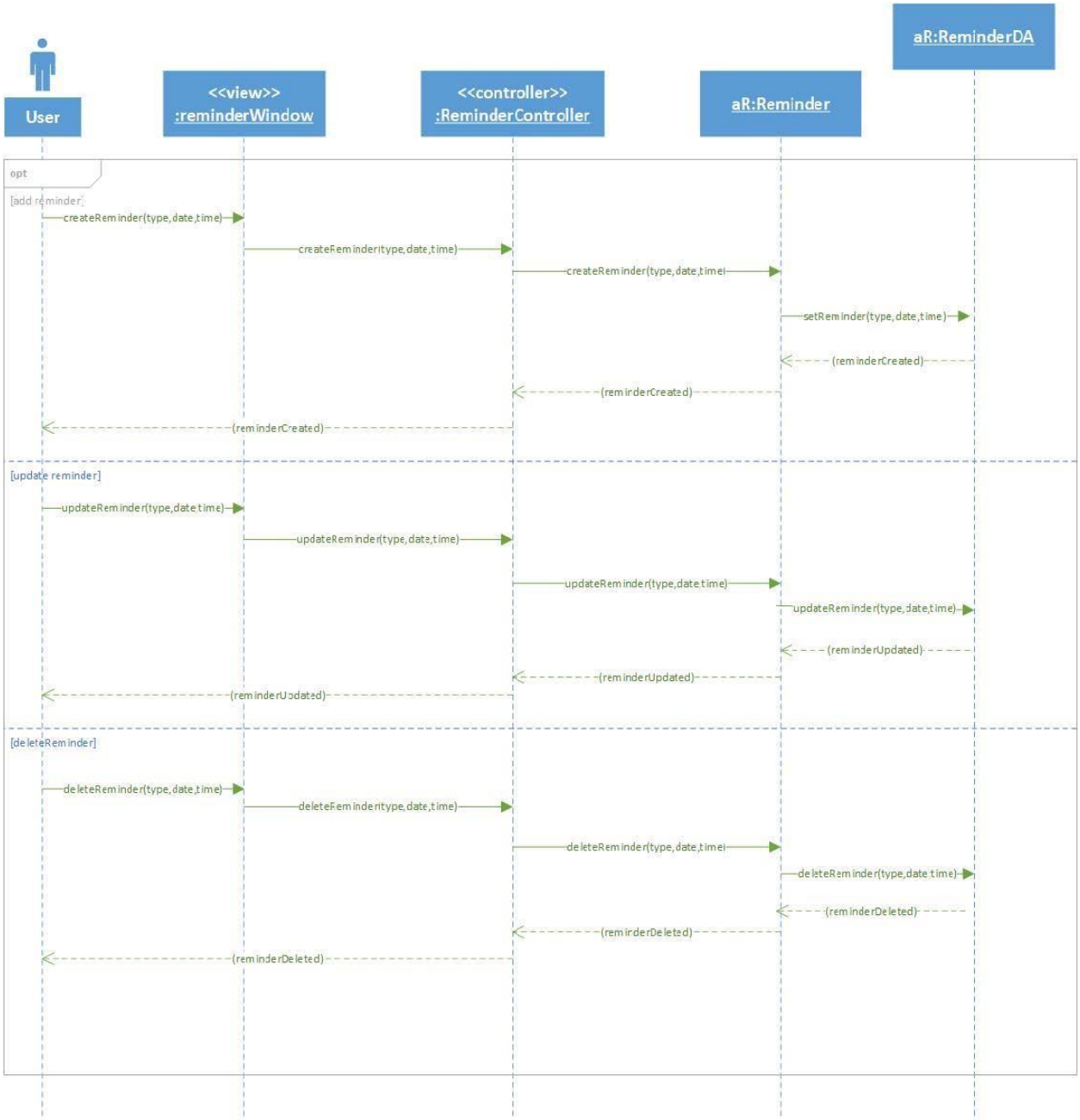


Figure 19: Mange reminder sequence diagram.

4.3.1.7. Manage expenses

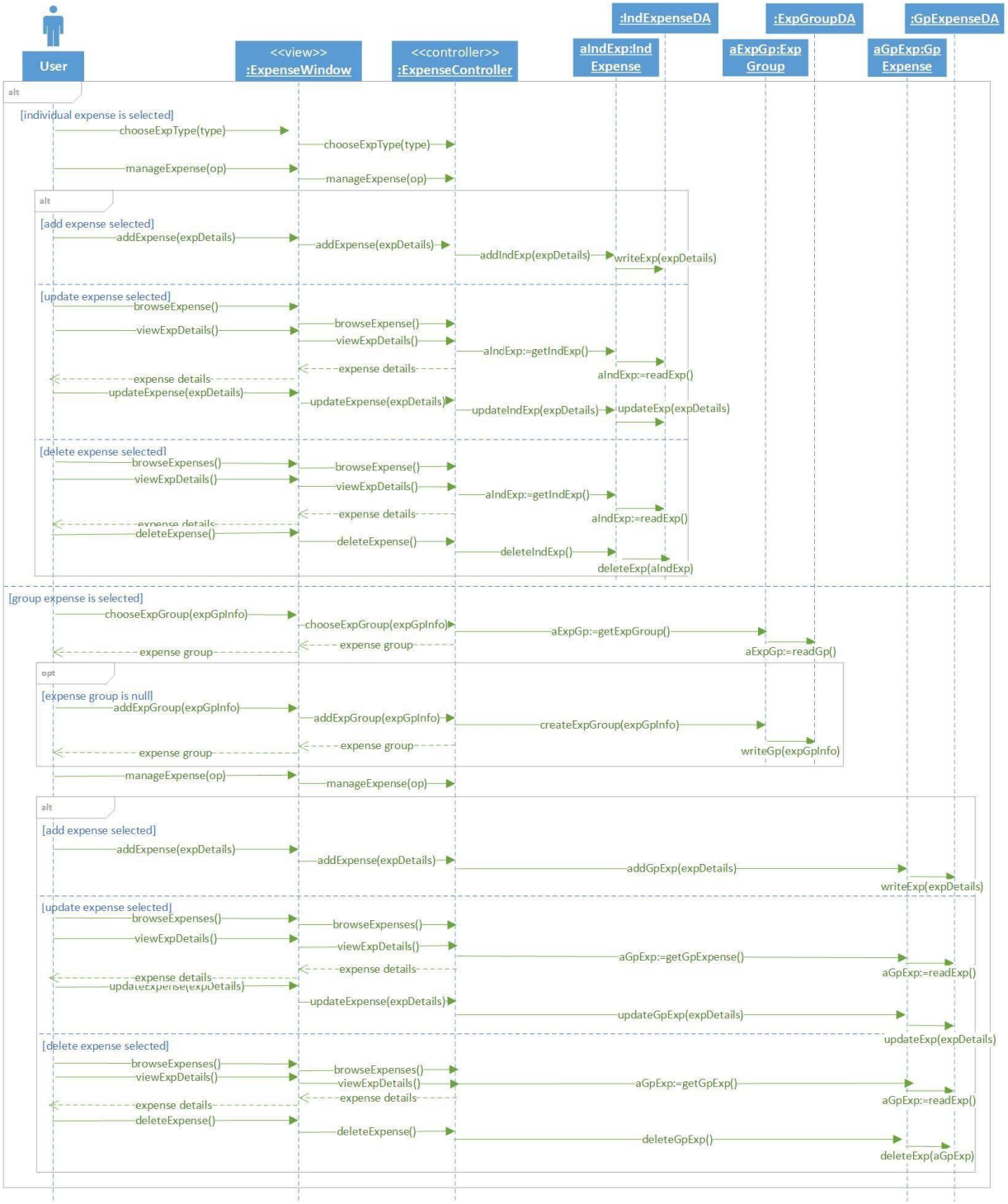


Figure 20: Manage expense sequence diagram.

4.3.1.8. View expense summary

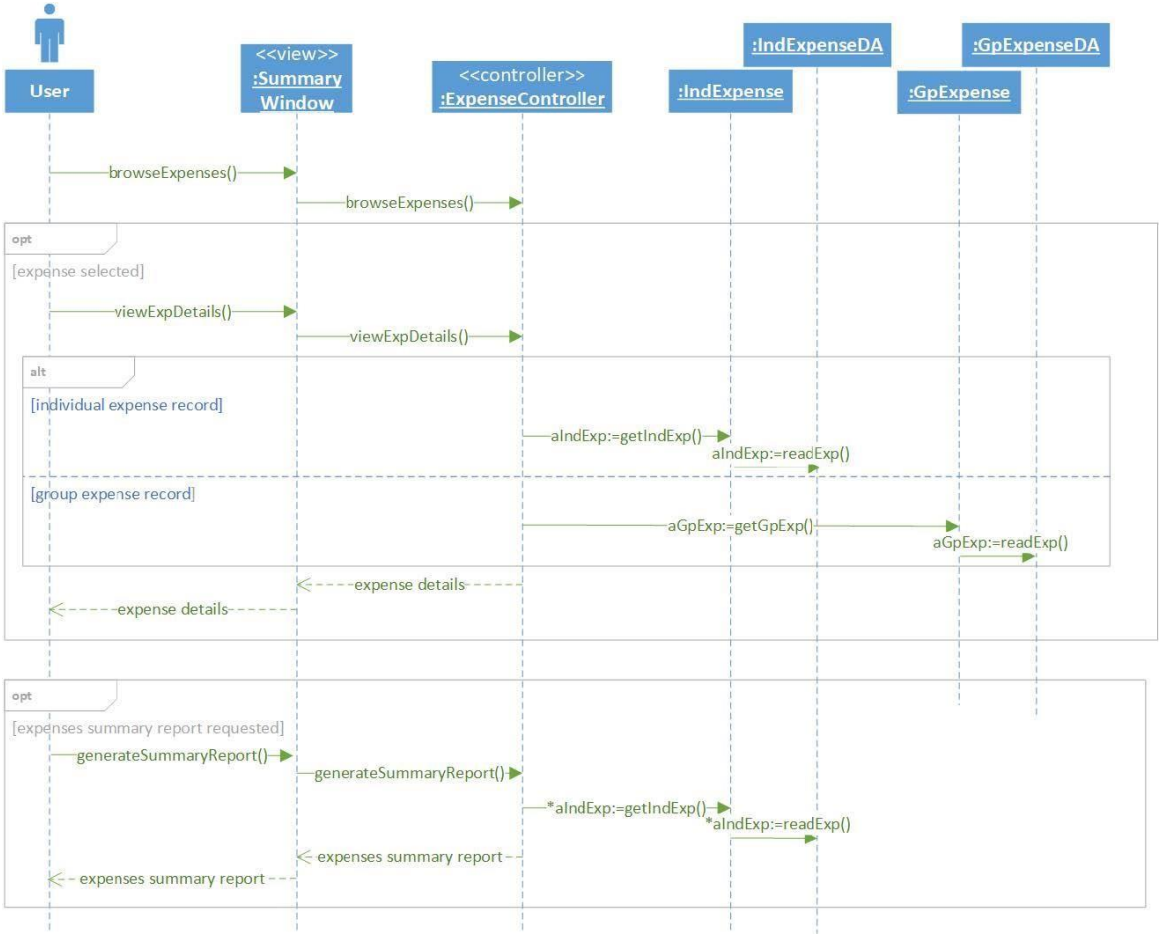


Figure 21: View expense summary sequence diagram.

5. System Implementation and Testing

5.1 Implementation Strategy

5.1.1. Development order

The order in which software components are developed, tested, and deployed is known as development order. The development order for this project is Input, Process, and Output (IPO). Because input programs and modules are created first, they may be used to enter test data for process and output programs and modules, making testing easier.

5.2 Testing Strategy

5.2.1. Integration Testing

Unit testing ensures that the smallest pieces of code, such as a method or function, perform as anticipated. Unit testing was used in this project to split down the code into separate units and verify whether the delivered value matches the anticipated outcome. There are also unit testing generators that may be used to test numerous bits of code, depending on the environment and the language being used.

5.2.2. Usability Testing

This test covers both functional and non-functional interfaces, as well as the interface's quality. It should be carried out to ascertain if the systems operate as anticipated and are user-friendly.

5.2.3. Acceptance Testing

Acceptance Testing ensures that the deliverable satisfies agreed-upon user criteria and is capable of supporting all possible business and user situations. The user may highlight issues that may have been overlooked during a unit or integration testing. Acceptance testing will be conducted on the system by the respective user to ensure it meets their needs.

5.3 Deployment Strategy

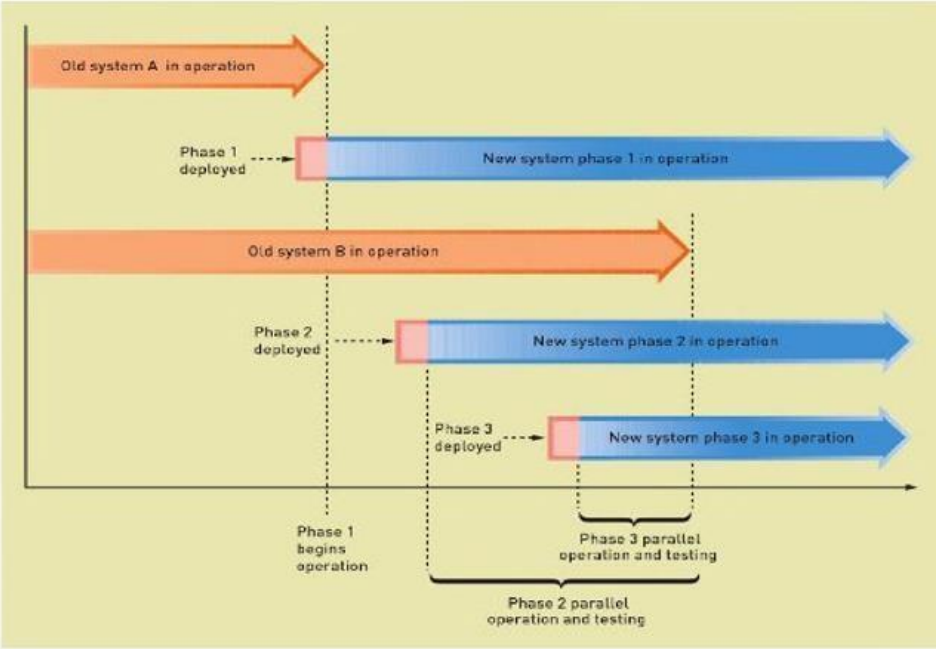


Figure 22: Deployment Strategy.

The system is gradually deployed through several stages or phases. Each step augments the operating system with new components or functionalities. We chose this deployment method because it requires testing at each step to verify that the system is ready for the next phase. Parallel deployment may be coupled with phased deployment, especially when the new system is tasked with taking over the operation of several old systems. As shown above, the new system supersedes the two current systems. The main benefit of phased deployment is risk reduction since the failure of a single step is less concerning than system failure. The downside of this deployment strategy is that it adds complexity to the system.

6. Conclusion

Our system can offer users intelligent features to make life easier in this project. This method enables the user to manage events inside the program while also reminding them of everyday tasks such as significant events, payment due dates, and medicine reminders. Additionally, users may invite their friends to certain events.

Users may manage individual or group expenditures in the expense tracker subsystem. Additionally, users may keep track of their daily spending in several areas. The user will see an expenditure summary record. Users may set an expenditure limit to remind them how much money they have remained in their budget.

Finally, the health management subsystem offers a health calculator that allows users to verify their health status while also tracking their weight and receiving a summary report. Additionally, users may get a healthy eating plan as part of the healthy consultation bundle. This initiative has three goals. The first goal is to offer intelligent advice to consumers, such as reminders or notification alerts about daily activities such as events, medicine, and spending

restrictions. The second goal is to organize users' everyday activities and assist them in managing their time effectively. The third goal is to educate the user toward a better lifestyle, and finally, this system is designed to aid users in tracking daily expenditures and resolving personal financial issues as well as those involving a third party.

The benefit of our system is that we integrated numerous elements from various types of existing applications and added new ones. For instance, the reporting and reminder modules offer reminders for many aspects of life, such as events, medications, and expenses. Following that is the online invitation function, which allows users to invite friends to events they create. This is a new feature in our system. However, the drawbacks include that the system must be constantly updated in comparison to other current systems, which is necessary given the market's high level of competitiveness. Suggestion for future work is that we may identify more real-world issues in order to suggest remedies in our system and so enhance our performance while also assisting individuals in their lives.

References

[1]. Satzinger, J. W., Jackson, R. B. and Burd, S. D., ***Introduction to Systems Analysis and Design: An Agile, Iterative Approach***, 6th edition, Course Technology CENGAGE Learning, 2012.

Appendices

Appendix A: Interview Questions

Facts about Stakeholder

1. What is your name?
2. Which company and department are you from?
3. What title and role do you hold in your organization?
4. What is the primary responsibility for your job?

Identify Problem

Open-end question

1. What problem did you faced that prevent you from performing your duties?
2. Why is it a problem?
3. How does the system help in your work?
4. What is the ideal solution for the system?
5. What primary function did you expect from the system?
6. What are the expected benefits of the system for you?

Close-end question

1. Do you find it difficult to manage your time?
2. Do you find it hard to manage your expenses and how financially secured you are?
3. How often do you use your mobile calendar?
4. How often do you exercise?
5. Do you think the Health tracker function is good? Why?
6. Have you heard of Budget Planner?
7. Have you heard of Health Pal Fitness?
8. Do you prefer the use of separate systems like Budget Planner, Health Pal Fitness, and mobile calendar or using all these features in one app?
9. Can you elaborate on why?

Summarize the problem captured to make sure you've understood correctly and captured everything relevant.

1. Are there any other problem you are facing?
2. How do you solve the problem?

3. How do you think we can further improve the system?
4. What else is a problem for you?

Quality requirements

1. What are your expectations for system availability?
2. What are your expectations for system performance?
3. Who will manage system support and maintenance?
4. What are the organization's support requirements?
5. What are the organization's system maintenance requirements?
6. What are the organization's security requirements?
7. What are the organization's requirements for installation and configuration?
8. Are there any legal requirements or other regulatory requirements that need to be met?
9. Are there any additional requirements that we should know about?

Closing question

1. Are there any other questions you and I should be asking or missing?
2. Anything else you want to tell me?
3. Can I contact you again if I need to ask some follow-up questions?
4. Would you want to participate in a review of the requirements?

Appendix B: Questionnaire

Smart Life Manager system

Greetings
We are students from school of Computer Science. We are conducting a survey about life management in Malaysia. The result of this survey will be used to implemented in the system we are going to develop, which is Smart Life Manager System. This system aims to improve users lifestyle by introducing a time management system, health tracker, and budget planner.

* Required

Age? *

- ◯ < 18
- ◯ 18 - 25
- ◯ 26 - 35
- ◯ 36 - 45
- ◯ > 46

Do you have Problem Managing your time? *

- ◯ Yes
- ◯ No
- ◯ Maybe

How often do you exercise? *

- ◯ > 1 hour a week
- ◯ 1 - 3 hours a week
- ◯ 3 - 5 hours a week
- ◯ > 5 hours a week

Do you keep track of your health? *

- ◯ Yes
- ◯ No
- ◯ Maybe

Do you use apps to help you manage your daily life? *

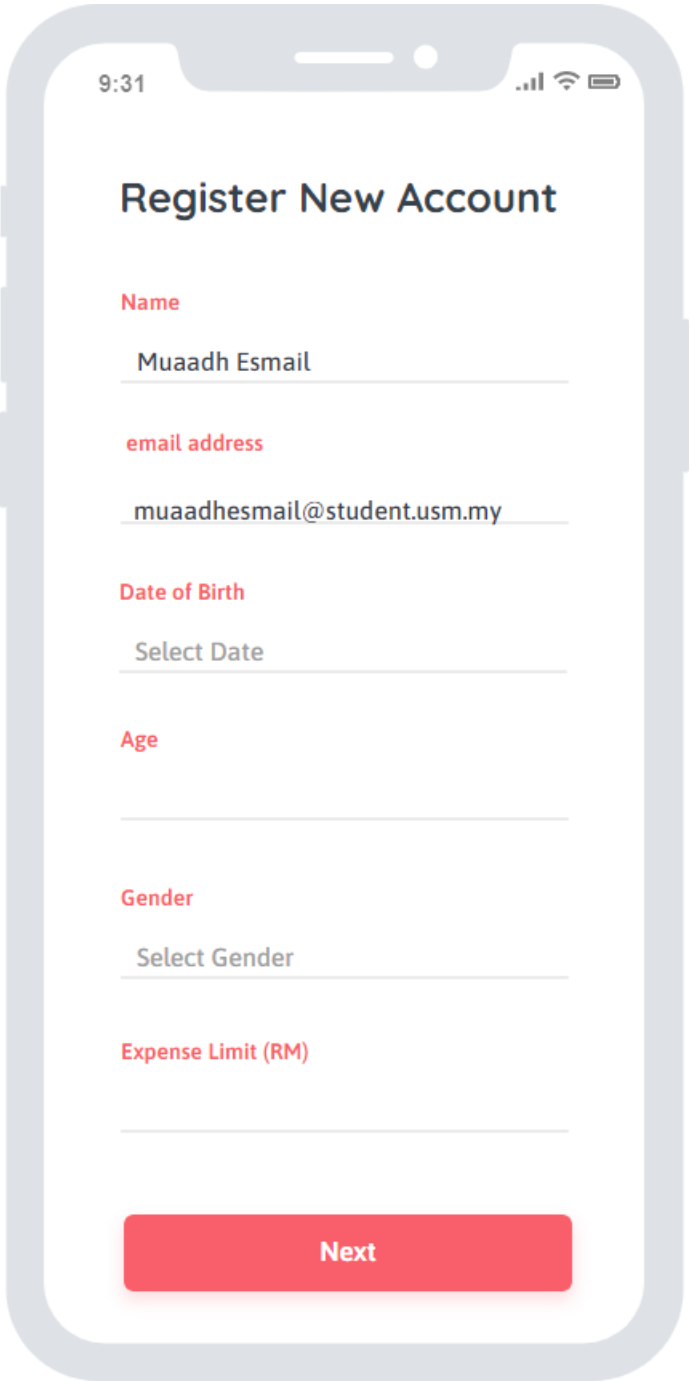
- ◯ Yes
- ◯ No
- ◯ Maybe

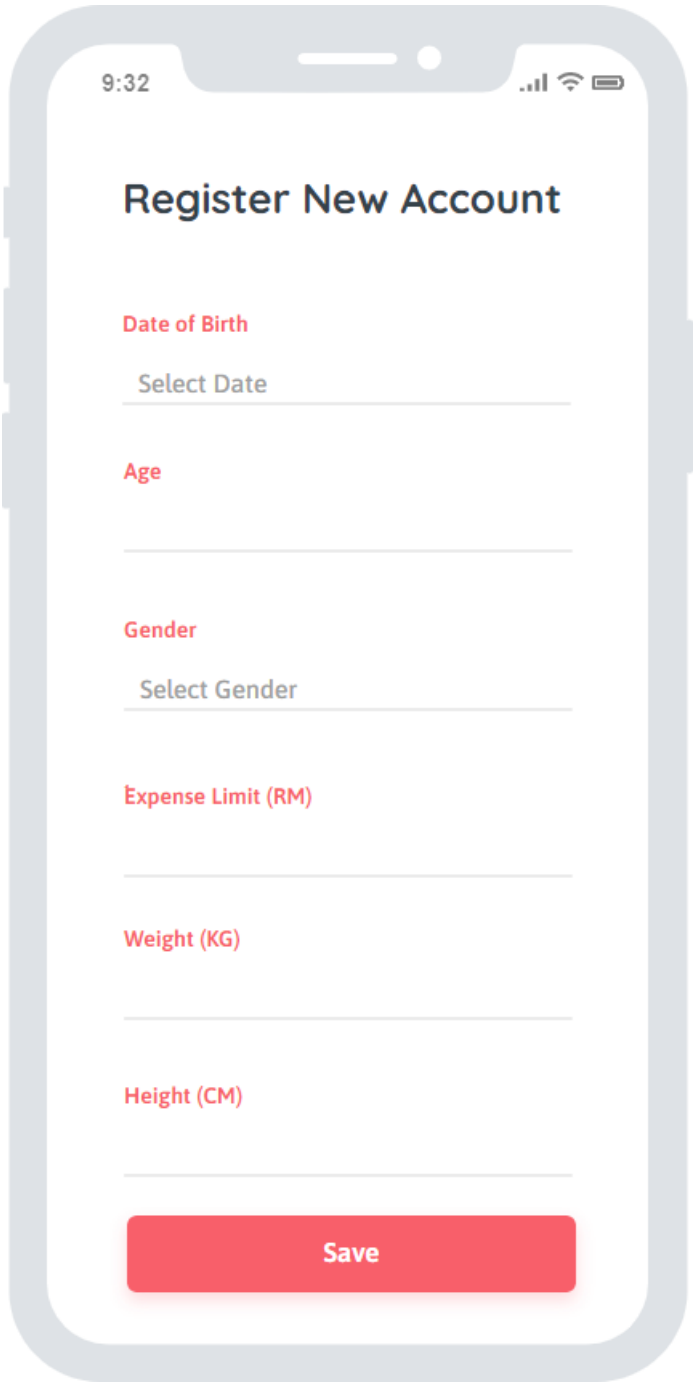
Would you like to use an app that has all the features mentioned above? *

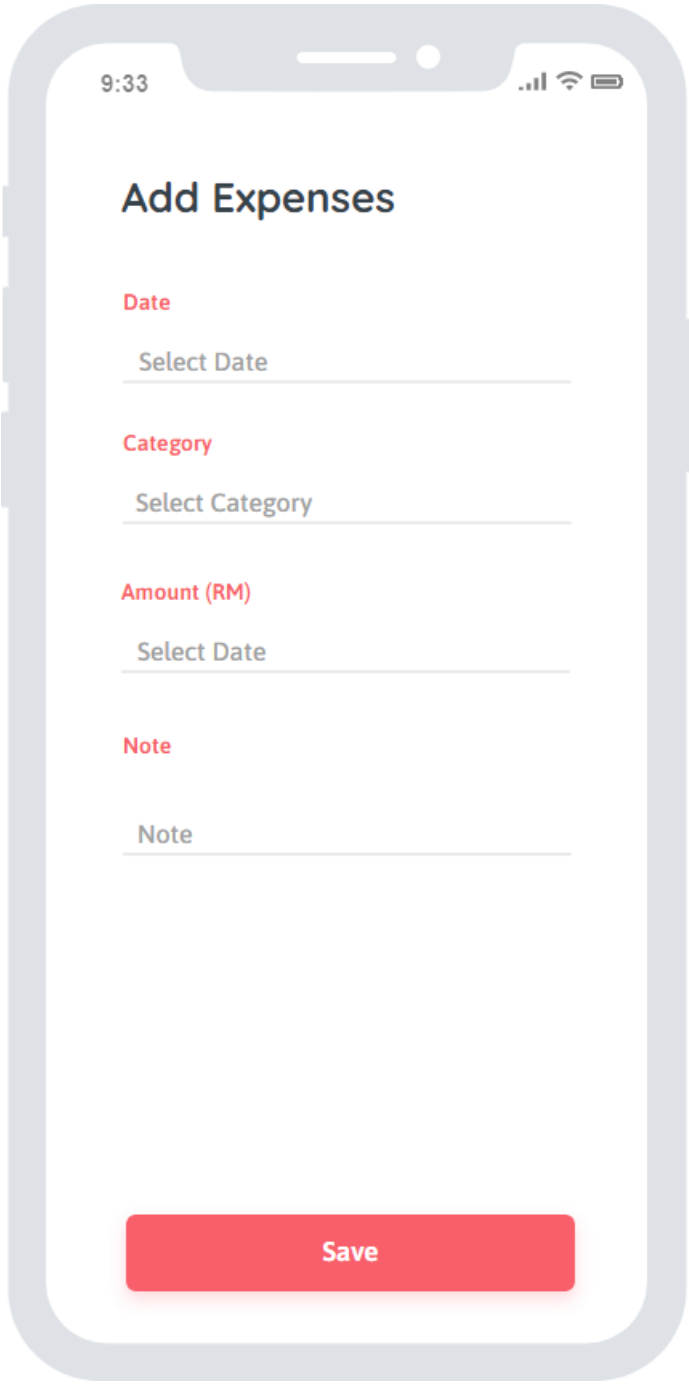
- ◯ Yes
- ◯ No
- ◯ Maybe

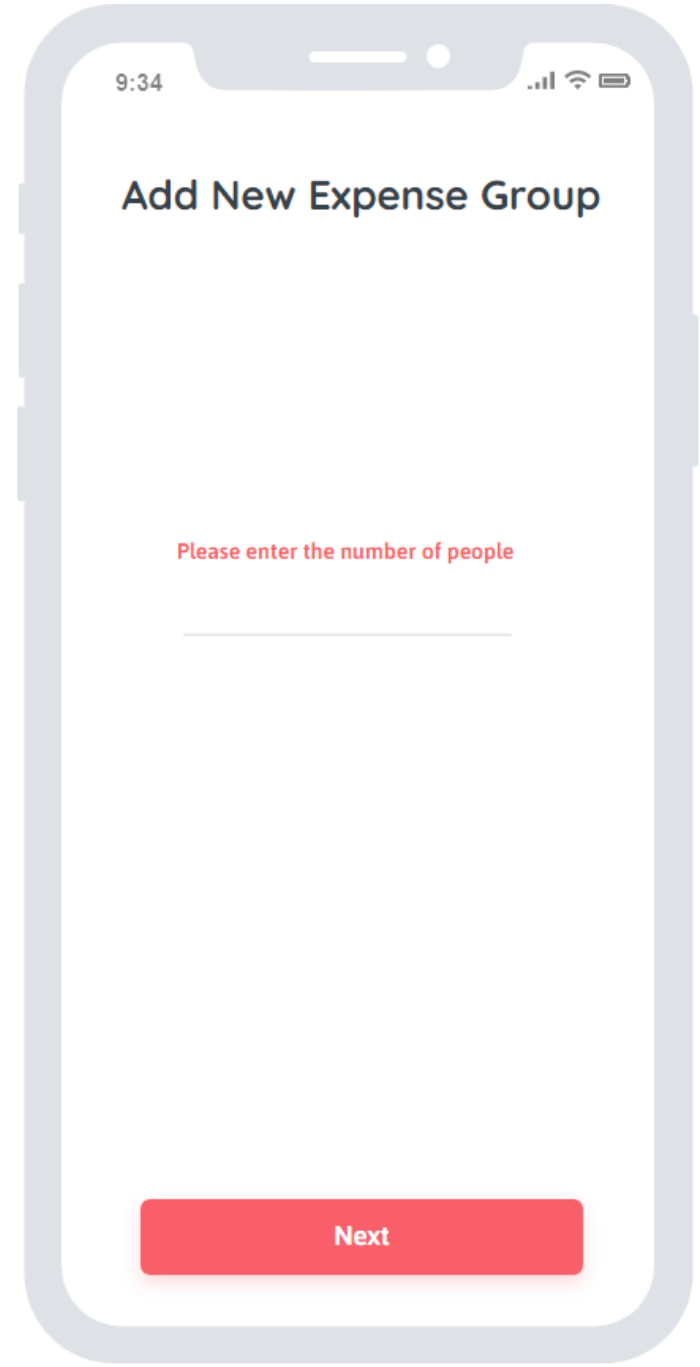
Submit

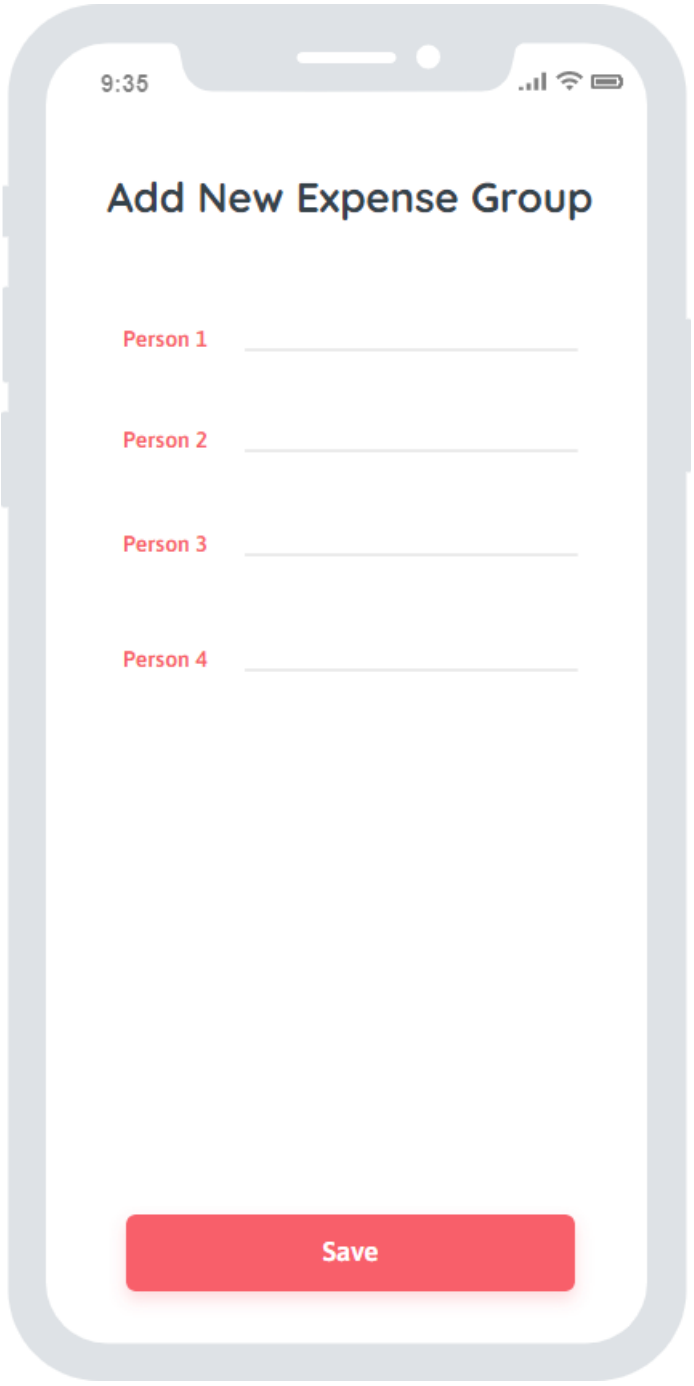
Appendix C: User Interface

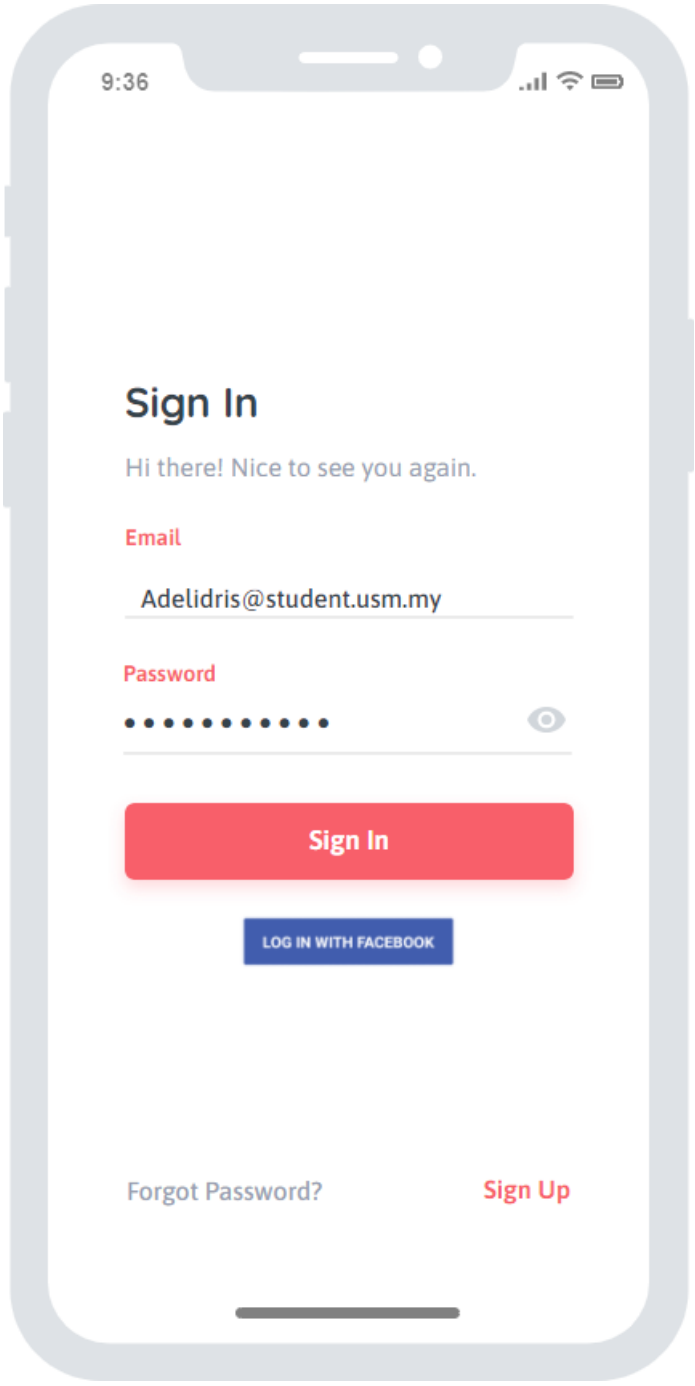


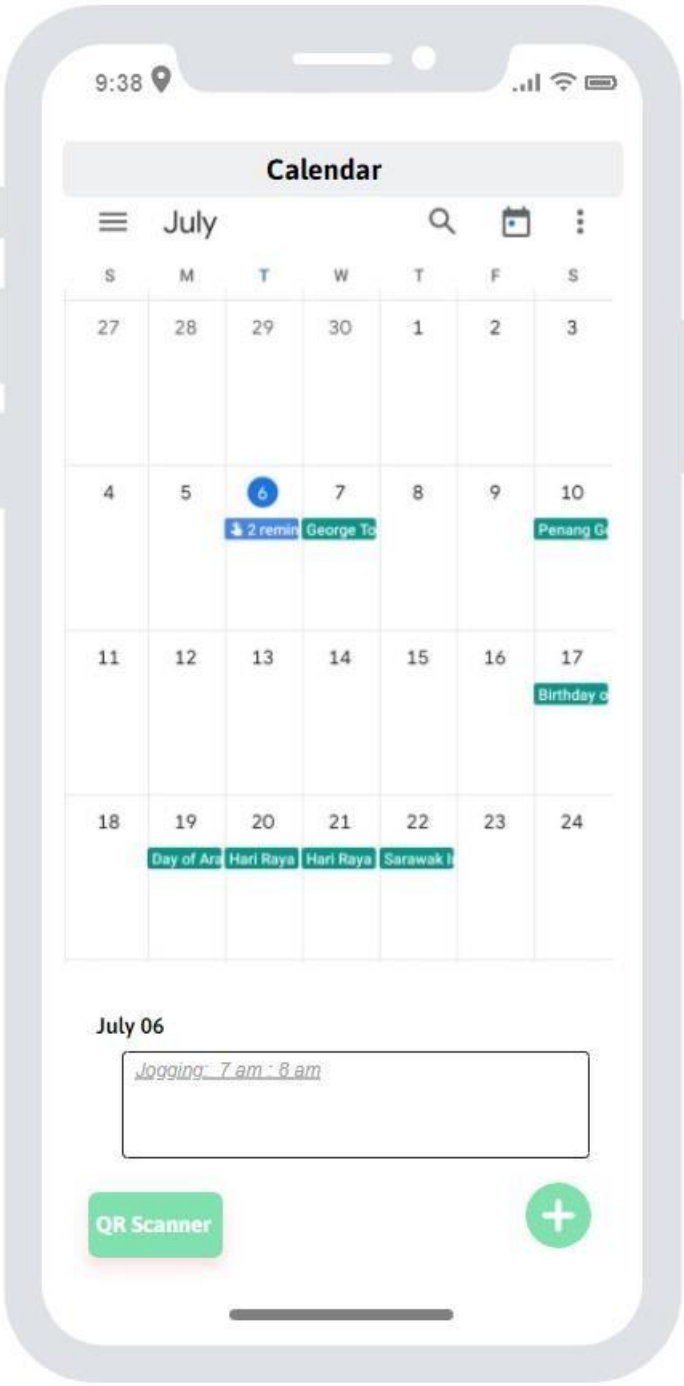

9:32
Register New Account
Date of Birth
Select Date
Age
Gender
Select Gender
Expense Limit (RM)
Weight (KG)
Height (CM)
Save

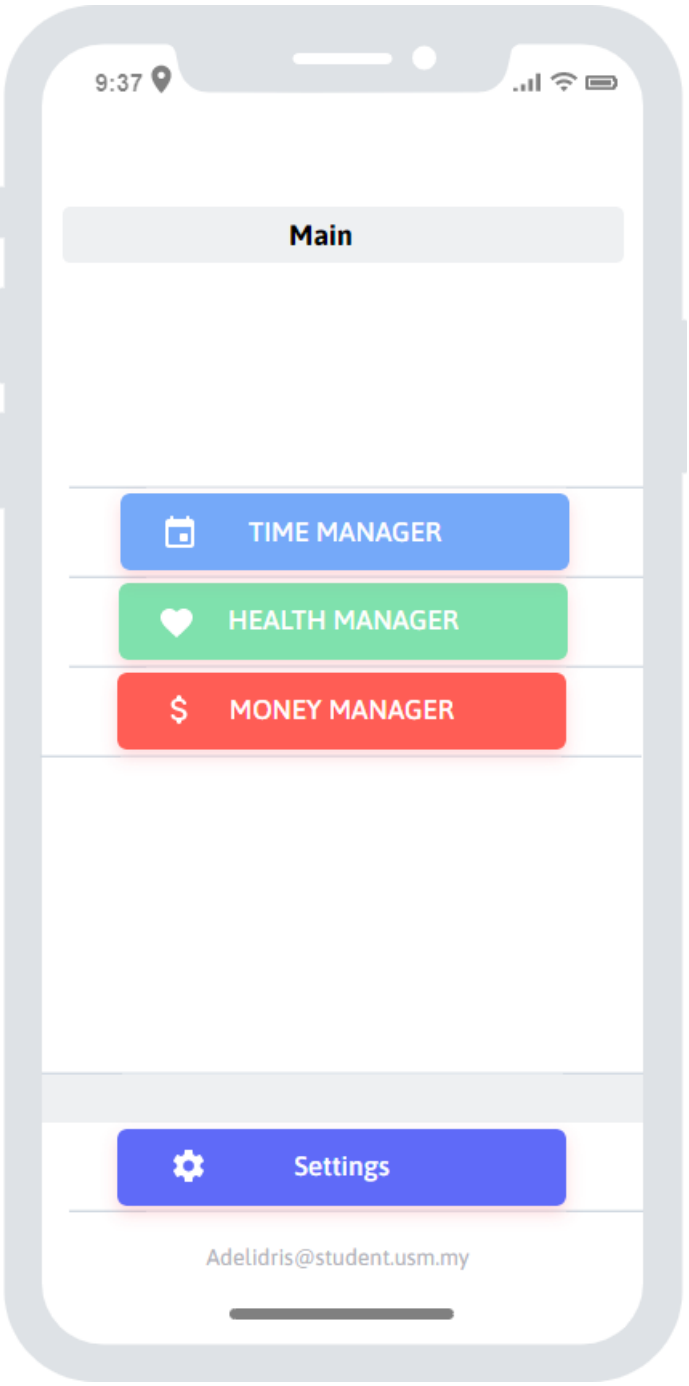

9:33
Add Expenses
Date
Select Date
Category
Select Category
Amount (RM)
Select Date
Note
Note
Save

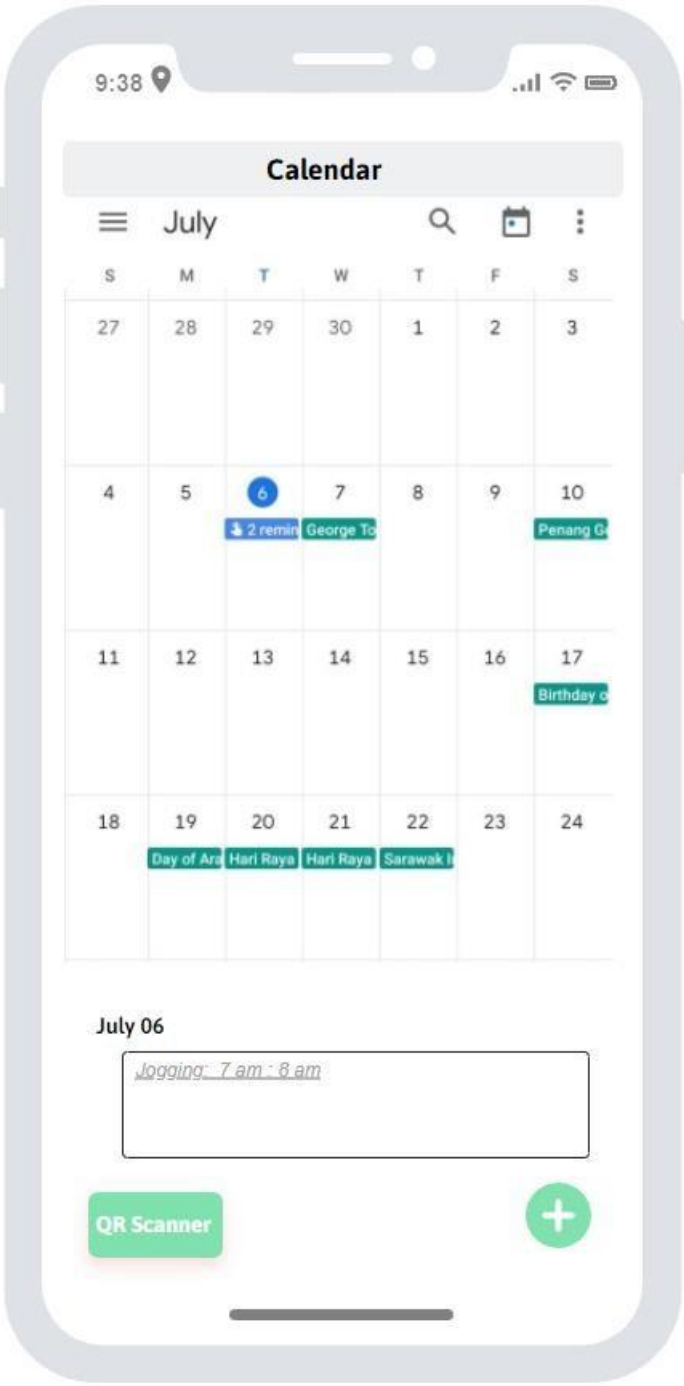

9:34
Add New Expense Group
Please enter the number of people
Next

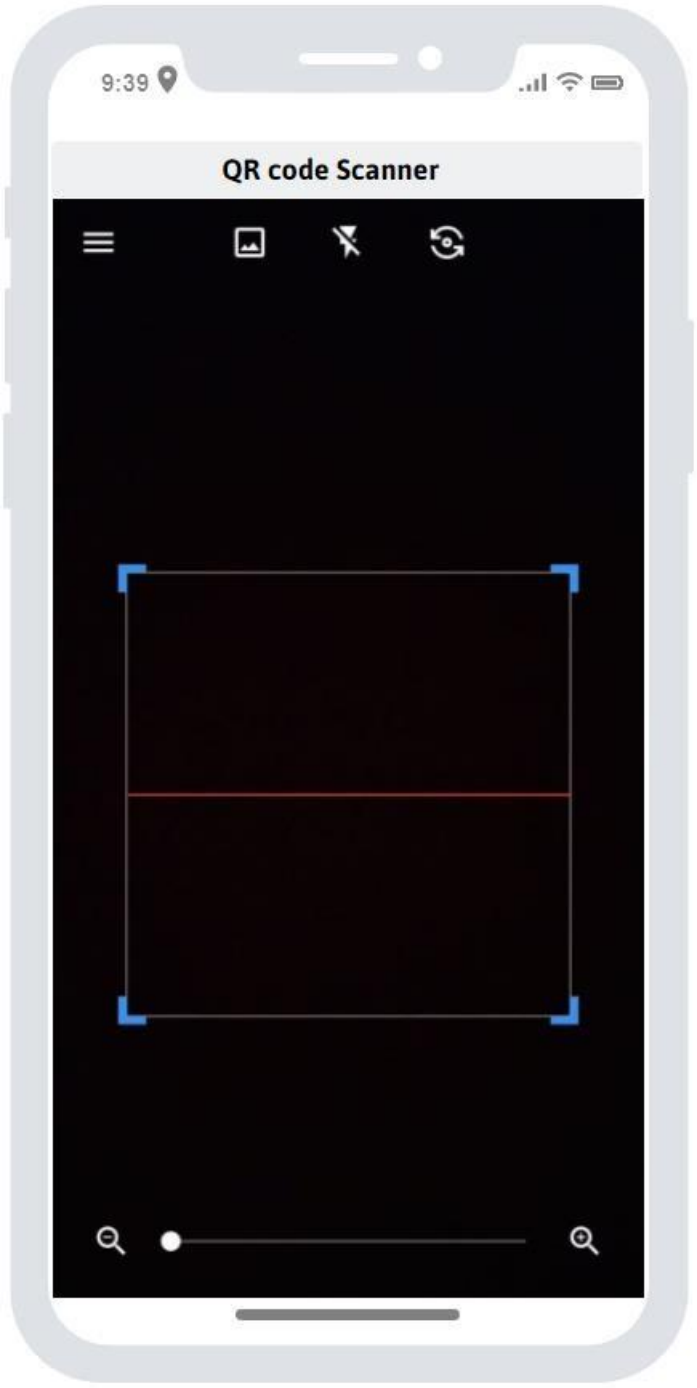

9:35
Add New Expense Group
Person 1
Person 2
Person 3
Person 4
Save

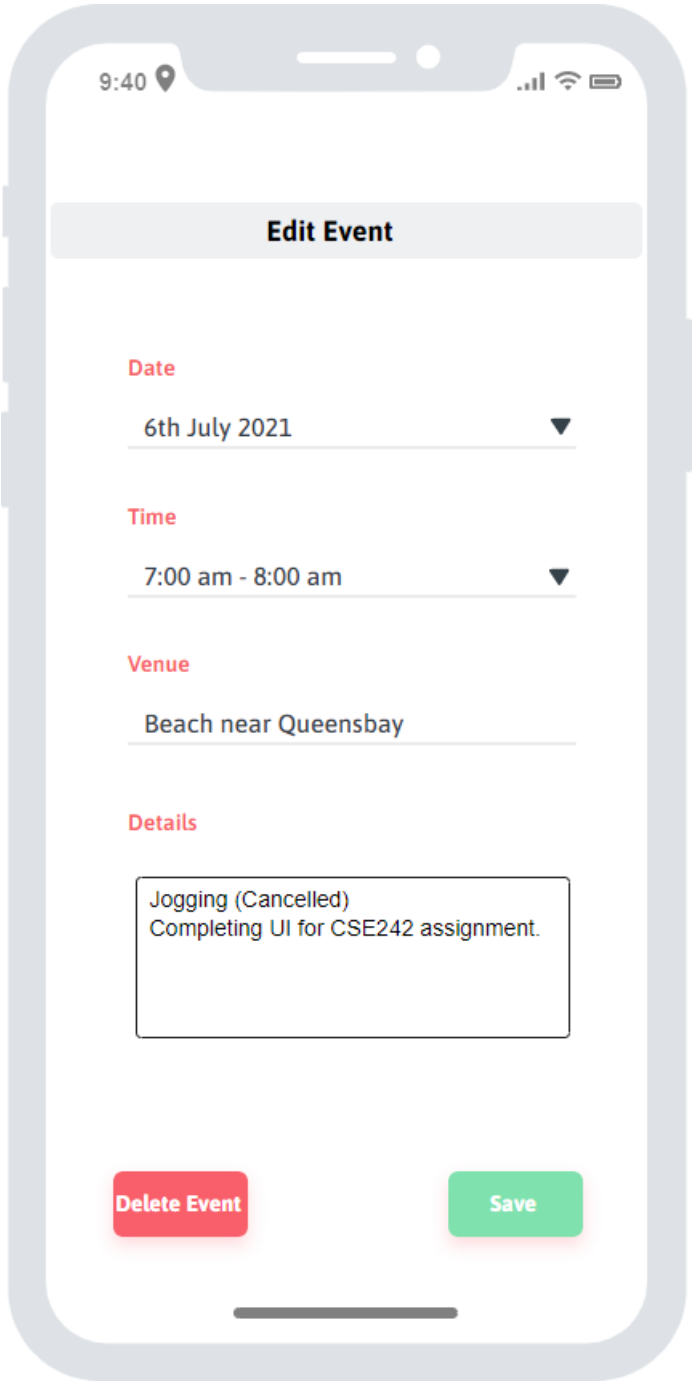

9:36
Sign In
Hi there! Nice to see you again.
Email
Adelidris@student.usm.my
Password
Sign In
LOG IN WITH FACEBOOK
Forgot Password?
Sign Up

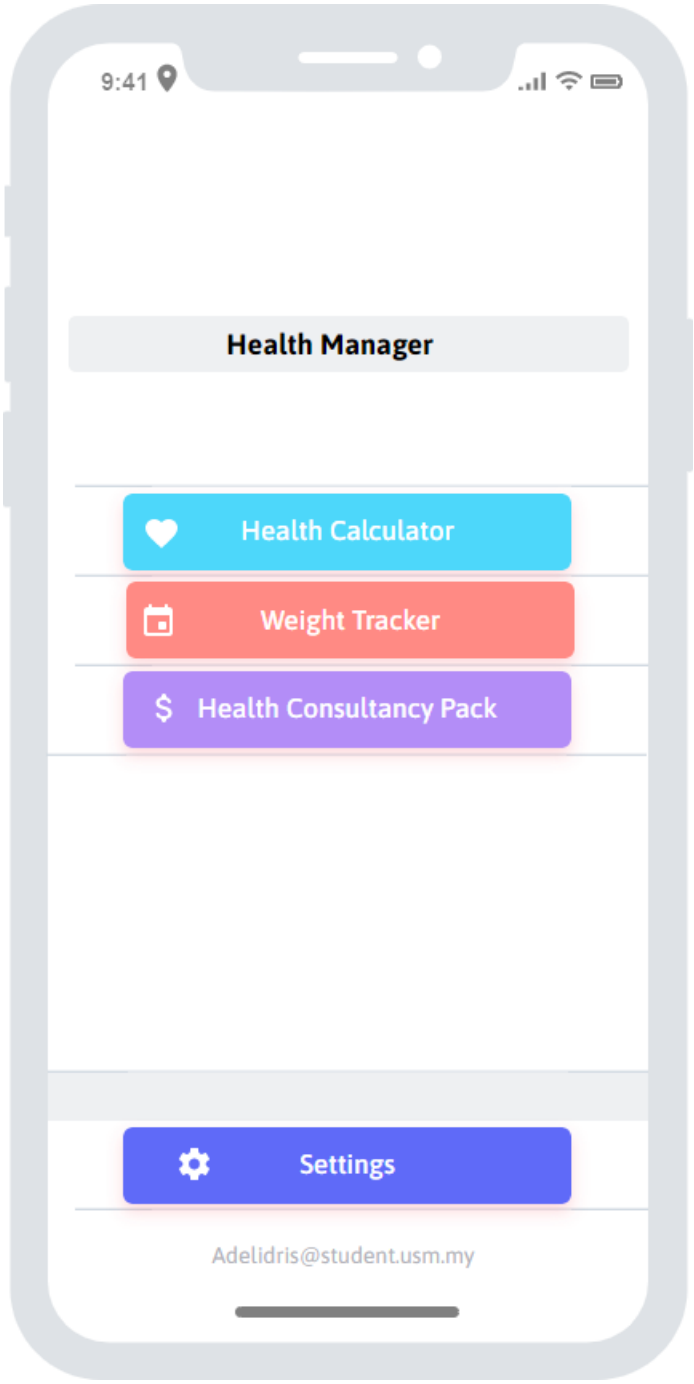

9:38
Calendar
July
S M T W T F S
27 28 29 30 1 2 3
4 5 6 7 8 9 10
2 remin
George To
Penang G
11 12 13 14 15 16 17
Birthday c
18 19 20 21 22 23 24
Day of Ar
Hari Raya
Hari Raya
Sarawak
July 06
Jogging: 7 am : 8 am
QR Scanner


9:37
Main
TIME MANAGER
HEALTH MANAGER
MONEY MANAGER
Settings
Adelidris@student.usm.my

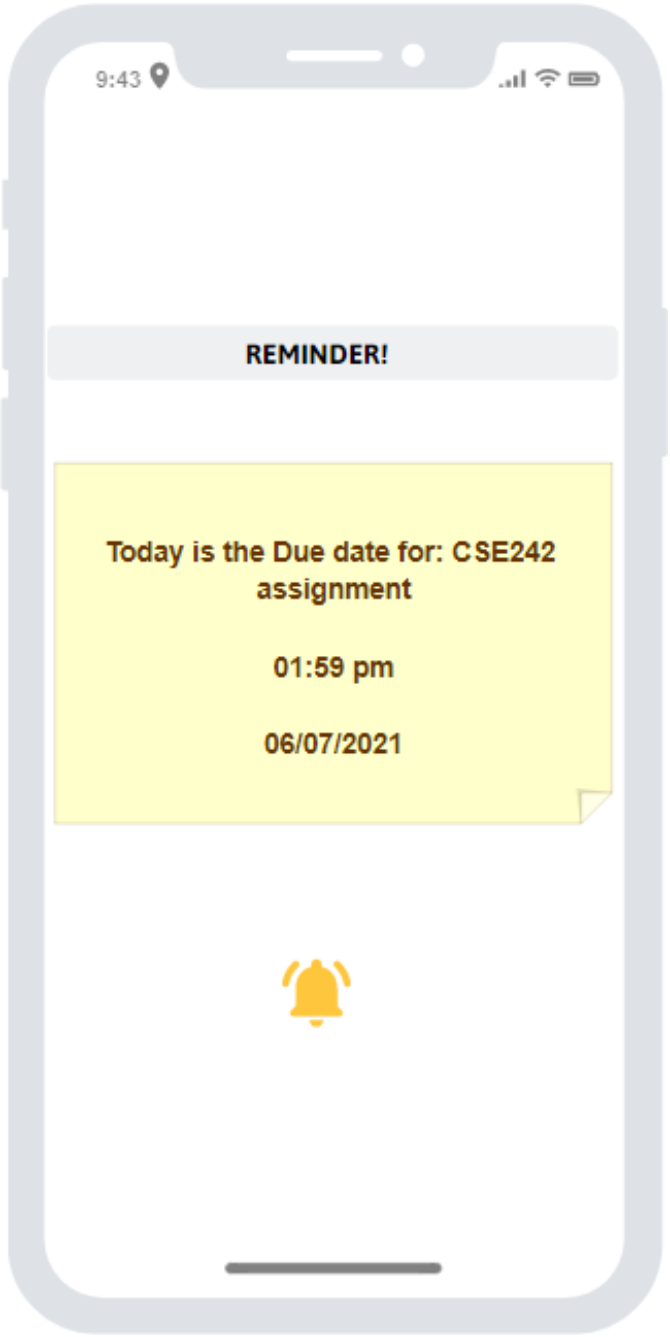

9:38
Calendar
July
S M T W T F S
27 28 29 30 1 2 3
4 5 6 7 8 9 10
2 remin
George To
Penang G
11 12 13 14 15 16 17
Birthday c
18 19 20 21 22 23 24
Day of Ar
Hari Raya
Hari Raya
Sarawak
July 06
Jogging: 7 am : 8 am
QR Scanner


9:39
QR code Scanner


9:40
Edit Event
Date
6th July 2021
Time
7:00 am - 8:00 am
Venue
Beach near Queensbay
Details
Jogging (Cancelled)
Completing UI for CSE242 assignment.
Delete Event
Save


9:41
Health Manager
Health Calculator
Weight Tracker
Health Consultancy Pack
Settings
Adelidris@student.usm.my


9:42
Expense Summary
Education
24.1 %
3.4 %
Entertainme
27.1 %
Summary of Expenses
ving
15.3 %
6.9 %
Clothes
3.0 %
Transport
20.2 %
Food
Food
Transport
Clothes
Living
Education
Entertainment
Others
Settings


9:43
REMINDER!
Today is the Due date for: CSE242 assignment
01:59 pm
06/07/2021