Software Requirements Specification

for

Module 2 : MyKhairat Application Project

Version 1.0 approved

Prepared by

Nor Syafiqa binti Abd Rahman (202422)

Safraa Khairunnisa binti Rahim (202375)

Nurul Aina binti Ariffin (204309)

Muhammad Aiman bin Mohd Rahimi (204235)

Mohamad Ramzuzzhini bin Mohamad Ramlee (201542)

Faculty of Computer Science And Information Technology

Universiti Putra Malaysia

14 April 2022

Table of Contents

1. Introduction

1.1. Purpose

The aim of this Software Requirement Specification (SRS) document is to outline the requirements for the MyKhairat application to be developed for Masjid Raja Muda Musa, Semenyih, Selangor. This document will give a detailed description of the requirements needed in the application including diagrams.

1.2. Document Conventions

This document is formatted to be in A4 Paper with Arial font. The normal text is with the font size of 11, while the section headings has the 20 font size and is bolded. For the subheadings, has the font size of 16 and bolded, and the third subheadings are bolded with the 14 font size. The page numbers are on the bottom right corner of every page with 11 font size and coloured with light grey.

Each requirement has a unique identification number that consists of a string of letters followed by a number, for example, REQ1. These unique identification numbers are assigned when the requirement is created according to the use case diagram and its relation. The requirement numbers are intended to be permanent and must not change when sections or requirements are added to or deleted from this document. Each requirement is also assigned a short title that allows it to be identified in the table of contents.

1.3. Intended Audience and Reading Suggestions

This SRS document is intended for individuals that are directly involved with the development of the MyKhairat application and also for the stakeholders. This includes software developers, team managers, project coordinator, and committee members of the Masjid Raja Muda Musa. All the sections are independent and readers may jump to any section they find relevant, however the reader can read this document sequentially for better understanding.

- Section 1 (Introduction)

  This section offers a summary of the MyKhairat Application project, including the purpose, document conventions, intended audience, product scope and references.

- Section 2 (Overall Description)

  This section describes the perspective and general functionality of the application. It also explains about the actors involved, the environment in which the system operates, design and implementation, assumptions and dependencies of the application.

- Section 3 (External Interface Requirements)

  This section provides illustrations of user interfaces of each function in the application and also describes the hardware, software and communication interfaces that will be used for the application.

- Section 4 (System Features)

  This section describes all the eight functional requirements that will be implemented in the application in detail; Edit Profile, View Dependent Record, View Dependent Grave, Validate Info, Add Dependent and Search Members. The details of these functions are included with use cases, flow of events (basic flow and alternative flows), activity diagrams and sequence diagrams (in Appendix B) to help the readers understand how the application works in a specific manner.

- Section 5 (Other Non-Functional Requirements)

  This section describes the application properties which are known as quality attributes rather than the product features that focuses on user expectations rather than users requirements. The main ones are performance, safety and security. This section also has details about the business rules which state the conditions for specific users to perform certain functions.

- Section 6 (Other Requirements)

This section describes other requirements; legal requirements and database requirements that are not covered elsewhere in this document.

1.4. Product Scope

This module of MyKhairat Application aims to facilitate the process of handling Kariah members' personal information and their dependents which include editing profile, viewing their dependent record along with grave location and reporting new death when one of the dependents died. The information provided by the Kariah members will be used for the mailing process and for distribution for the death benefit. Apart from that, the masjid committee can view the recent death record list of Kariah members' dependents and also make a quick search for Kariah members.

These functions are important as it will help the masjid committee to keep on track with the new reported death in the real-time and help the Kariah members to know their dependents' grave location easily. This application will enhance the existing management system making it more efficient, more organised and time effective.

1.5. References

1. "IEEE Guide for Software Requirements Specifications," in IEEE Std 830-1984 , vol., no., pp.1-26, 10 Feb. 1984, doi: 10.1109/IEEESTD.1984.119205.

2. Software Engineering Standards Committee of the IEEE Computer Society. (2009, December). *IEEE Recommended Practice for Software Requirements Specifications*. Michigan State University. https://cse.msu.edu/~cse870/IEEEXplore-SRS-template.pdf

3. E. Wiegers, K. (1999). *IEEE Software Requirements Specification Template*. Dalhousie University. https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc

2. Overall Description

2.1. Product Perspective

This module is made for editing profile details of MyKhairat Member to allow easier procedures for not only the Masjid Raja Muda Musa committee but as well as all masjids who handle Khairat throughout Malaysia to ensure the details they have about the Members are correct and updated, for Members to view grave details of their dependents and submit new death records of their dependents in other to report the death and claim the money of islamic death benefit collection and for Masjid Committee to search the registered Members and view their profile. The module will be part of the three modules, each handled by each developing team, and to be merged together to produce the system.

As there are different roles in MyKhairat app that limit the functions and actions for what each user type can do, this module specifies some of the operations that are specified for each role. The finished product will be a mobile-based system.

2.2. Product Functions

The functions below are included in Module 2 of MyKhairat project:

F1 - Members can edit their profile.

F2 - Members can view their dependent record.

F3 - Members can add new dependents.

F4 - Masjid Committee can validate Member’s dependents info.

F5 - Members can view the location of the grave of their dependent.

F6 - Members can report the death of their dependents.

F7- Members can view records of their dependents.

F8-Masjid Committee can search Members based on their Id, Ic or name.

2.3. User Classes and Characteristics

There are only three user types for MyKhairat mobile app:

User Class	Characteristics
U1 - User	User is a person who is not validated and acts as a guest on the MyKhairat app who is only able to do limited actions.
U2 - Member	Member is a validated person that will use MyKhairat app and only allowed to do operations that are specified for ordinary users without knowing how the backend of this app operates. For this module, Members are able to edit their profile, submit a new death record and view dependent records, either to add a new dependent or view the grave information of their dependents.
U3 - Masjid Committee	Masjid Committee is an authorised person with access to perform administrative operation and handle the backend of MyKhairat. For this module, Masjid Committee are able to search registered Members to view their profile, view their death records and validate any new additional dependents.

2.4. Operating Environment

The operating environment for the MyKhairat application will be in the mobile environment.

OE1 - MySQL database will be used to store new and existing data.

OE2 - Flutter technologies will be used to build the MyKhairat application.

OE3 - Laravel will be used to handle the backend of MyKhairat application.

OE4 - Android Studio/Visual Studio Code, XAMPP and GitHub Desktop will be used as the programming environments.

2.5. Design and Implementation Constraints

C1 - The mobile application should be finished, fulfilling all client's requirements, within the 14 weeks given.
C2 - The mobile application is displayed only in Malay language.
C3 - Most developers for MyKhairat never have experience building the application using Flutter with Laravel backend.

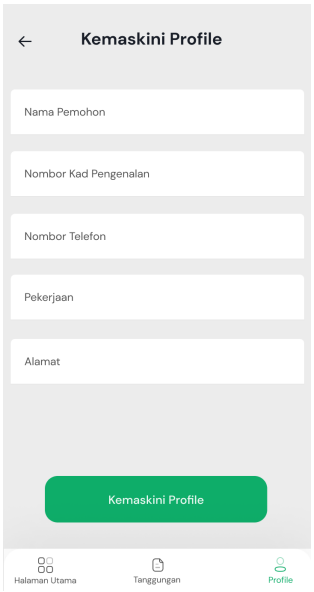
2.6. Assumptions and Dependencies

A1 - We assume that the users use a stable internet connection in order to access the application and its features.
A2 - We assume that the users have basic knowledge of technologies and know how to use MyKhairat without a user manual.
D1 - We depend on the servers to be running and functioning always.

3. External Interface Requirements

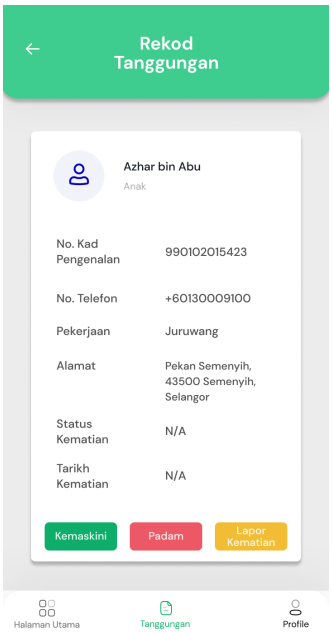
3.1. User Interfaces

3.1.1. Edit Profile (UI10)



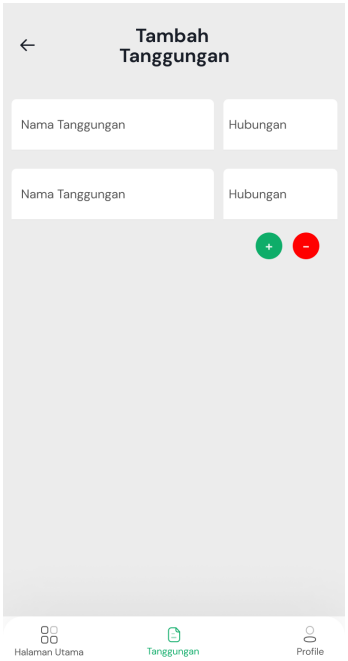
Edit Profile UI, members can edit their profile.

3.1.2. View Dependent Record (UI11)



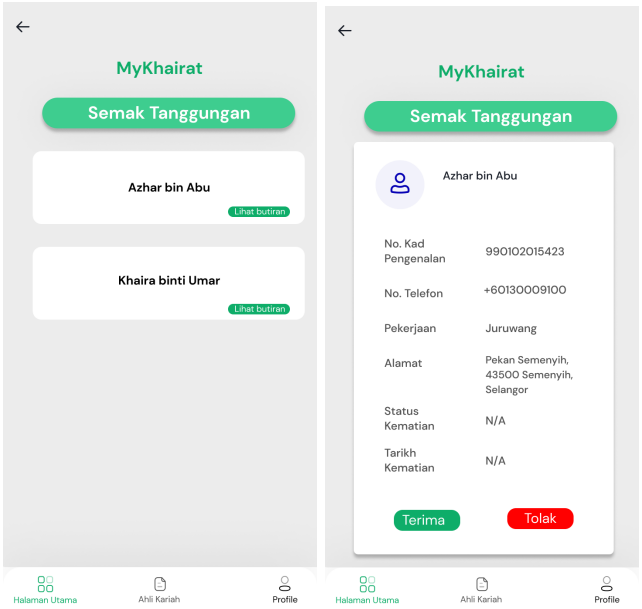
View Dependent Record UI, members can view the information of any dependent that has already been added.

3.1.3. Add Dependent (UI12)



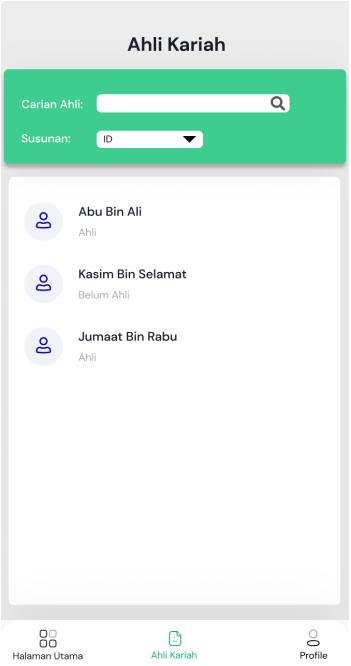
Add Dependent UI, members can add their dependents.

3.1.4. Validate Info (UI13)



Validate Info UI, Masjid Committee can validate users' dependents after they have registered them.

3.1.5. Search Members (UI18)



Search Members UI, masjid committee can search members of the committee by their ID, IC or name.

3.2. Hardware Interfaces

This application works on Android mobile devices, and as such, will be designed to interface with the hardware present on Android devices. In theory the application will be able to run by other devices that can emulate the Android, but this will not be a consideration during design. This application will be using the cellular network or WiFi to connect to the Internet, which will allow it to communicate with database servers. This means that it will be using the infrastructure, be it wireless communication points or physical lines, of the network in order to perform properly.

3.3. Software Interfaces

The software used in developing this application are Visual Studio Code and Android Studio for programming. The android emulator that comes along with Android Studio software, Android Virtual Device (AVD) will be used to stimulate the application in the mobile mode and the application will be running in Android

as its operating system. For the database, this project will be using MySQL relational database to organise, store, and fetch data.

3.4. Communications Interfaces

As described above, there will be communication with the database so will be making use of the Android network and HTTPS in order to communicate. The primary forms of communication will be database transactions or requests.

4. System Features

This section describes the features of the system in diagram representations. It will include UML use case diagram, flow of events, activity diagram, and sequence diagram for each feature in module 2. Diagram below shows the application's use case diagram as a whole.

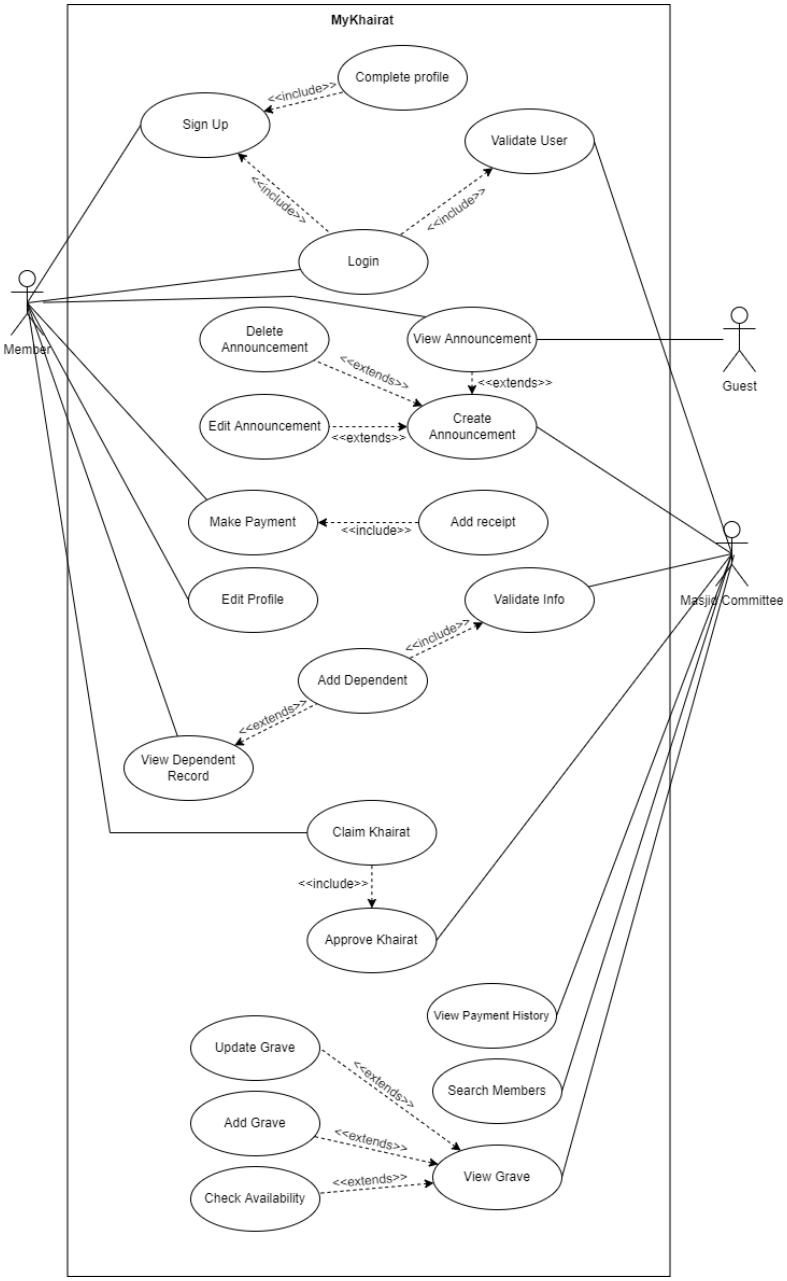


Diagram 4.1: System Use Cases

4.1. Edit Profile (REQ10)

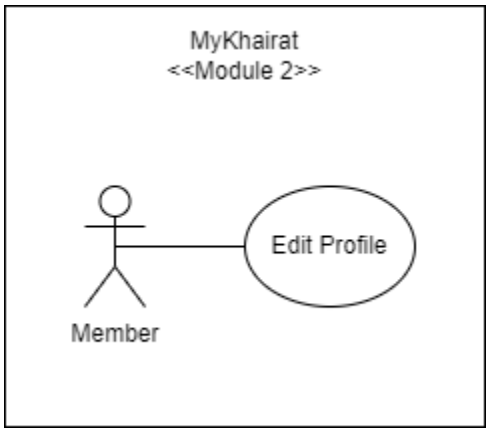


Diagram 4.1.1: Edit Profile Use Case

4.1.1. Description and Priority

This use case is initiated by the actor Member. This use case enables the Member to edit their profile by entering required details. This use case is of medium priority.

4.1.2. Flow of Events

4.1.2.1. Basic Flow

4.1.2.1.1. The Member shall click the ‘Kemaskini’ button on the Profile Page.

4.1.2.1.2. The Member shall fill in all required details.

4.1.2.1.3. The Member shall click on the ‘Kemaskini Profile’ button.

4.1.2.1.4. System shall verify the validity of input.

4.1.2.1.5. System shall direct the member to Profile Page with edited details. The use case ends.

4.1.2.2. Alternative Flow - None

4.1.2.3. Exception Flow - Start at 4.1.2.1.3.

4.1.2.3.1. Details inputs are invalid.

4.1.2.3.2. System shall display an error message and prompt the user to fill in valid details.

4.1.3. Precondition(s)

The Member has logged in.

4.1.4. Post condition(s)

The Profile page will display details that have been edited.

4.1.5. Notes/Issues

None.

4.1.6. Activity Diagram

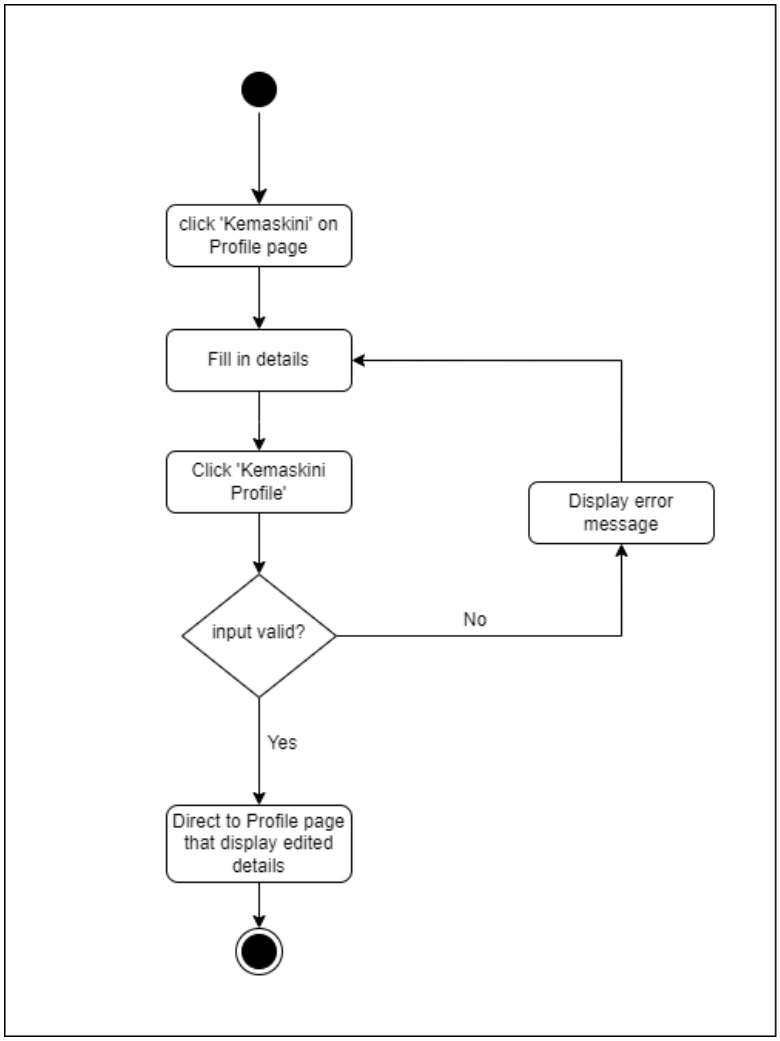


Diagram 4.1.6.1: Edit Profile Activity Diagram

4.2. View Dependent Record (REQ11)

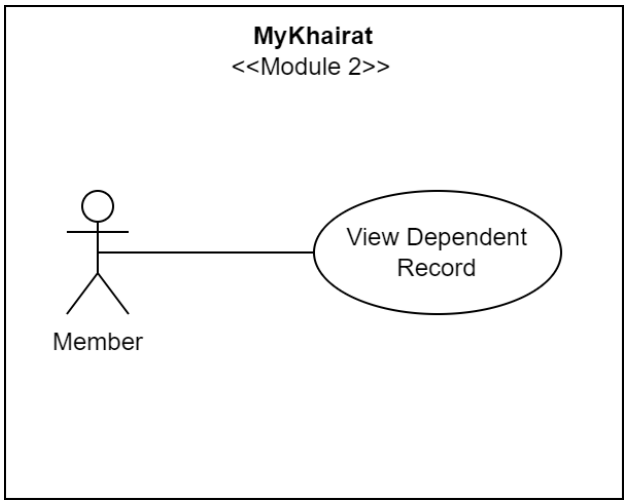


Diagram 4.2.1: View Dependent Record Use Case

4.2.1. Description and Priority

This use case is initiated by the actor Member. It enables the member to view the record of their dependents. In this record, it shows the name of the member's dependents and relationship between them. This use case has high priority.

4.2.2. Flow of Events

4.2.2.1. Basic Flow

4.2.2.1.1. The Member shall click on the menu navigation 'Tanggungan'.

4.2.2.1.2. The System shall display the list of the member's dependents name and the relationship.

4.2.2.1.3. The Member shall click on one of the dependents.

4.2.2.1.4. The System shall display the selected dependent's information in detail including buttons for editing, deleting and reporting for death of the selected dependent.

4.2.2.2. Optional Flow A - Start at 4.2.2.1.4.

4.2.2.2.1. The Member shall click the 'Kemaskini' button.

4.2.2.2.2. The System shall display the details form.

4.2.2.2.3. The Member shall fill in the form that needs to be edited.

4.2.2.2.4. The Member shall click the ‘Kemaskini Info’ button to submit.

4.2.2.2.5. The System shall save the updated information into a respective dependent database.

4.2.2.2.6. The System shall display the updated information in the dependent profile.

4.2.2.3. Optional Flow B - Start at 4.2.2.1.4.

4.2.2.3.1. The Member shall click the ‘Padam’ button to delete the dependent profile.

4.2.2.3.2. The System shall run a confirmation check.

4.2.2.3.3. The Member shall click the ‘Ya’ button.

4.2.2.3.4. The System shall delete the dependent profile from the database.

4.2.2.3.5. The System shall direct the member to the ‘Tanggungan’ page or also known where it lists all the member’s dependent(s).

4.2.2.4. Alternative Flow A - Start at 4.2.2.1.3.

4.2.2.4.1. The System shall display the selected dependent’s information in detail including the grave location.

4.2.2.5. Alternative Flow B - Start at 4.2.2.3.3.

4.2.2.5.1. The Member shall click the ‘Tidak’ button to cancel the action.

4.2.2.5.2. The System shall direct the member to the current dependent’s profile page.

4.2.2.6. Exception Flow

None.

4.2.3. Precondition(s)

The Member has logged in.

4.2.4. Post condition(s)

- If the selected member's dependent is not deceased, the 'Rekod Tanggungan' page will display the dependent's details and display three buttons: editing dependent's information, removing dependent and reporting dependent's death.
- If the selected member's dependent is deceased, the 'Rekod Tanggungan' page will display the dependent's details and the grave location.

4.2.5. Notes/Issues

Make sure the Member has added their dependent(s)'s details and getting approved by the Masjid Committee.

4.2.6. Activity Diagram

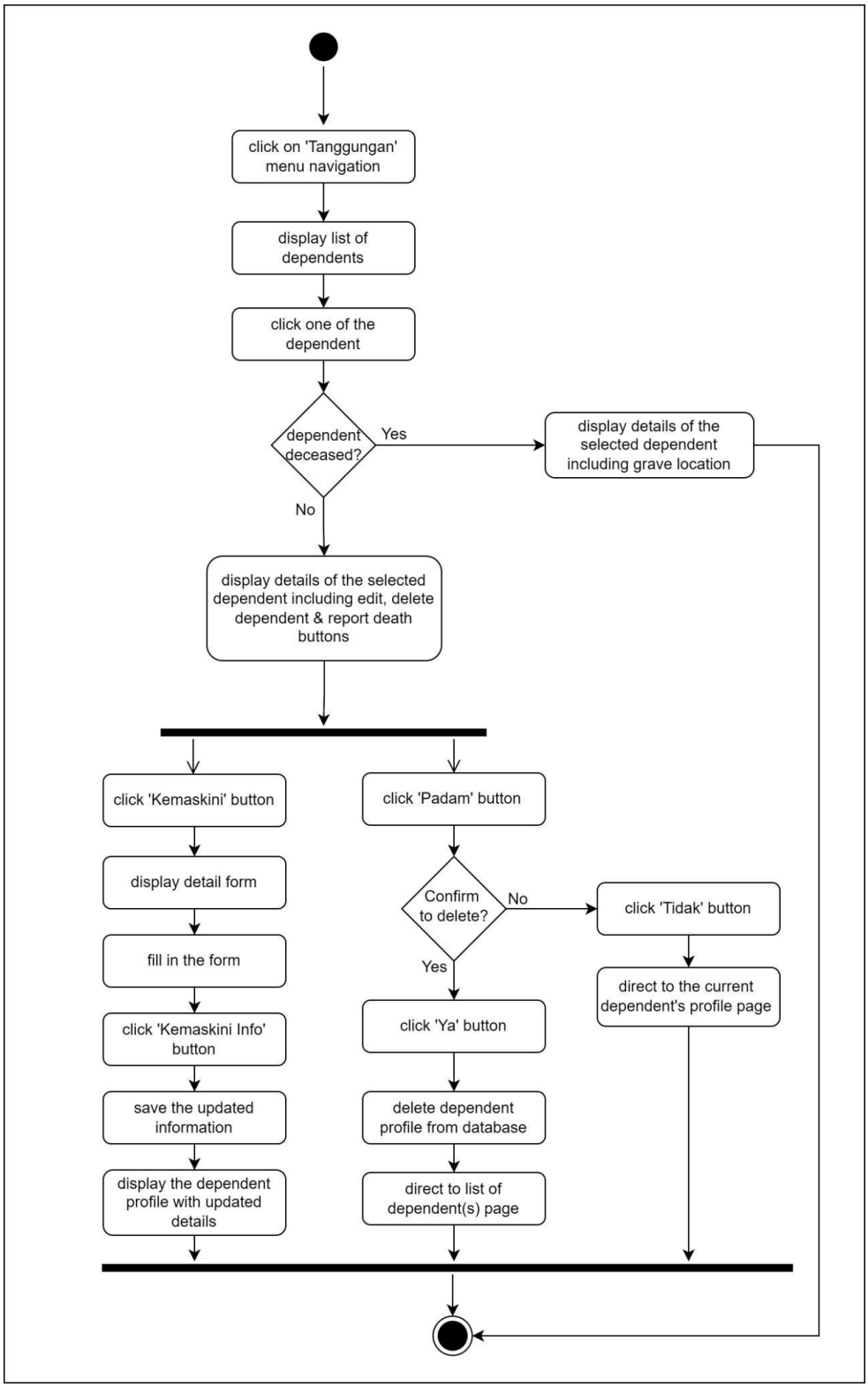


Diagram 4.2.6.1: View Dependent Record Activity Diagram

4.3. Add Dependent (REQ12)

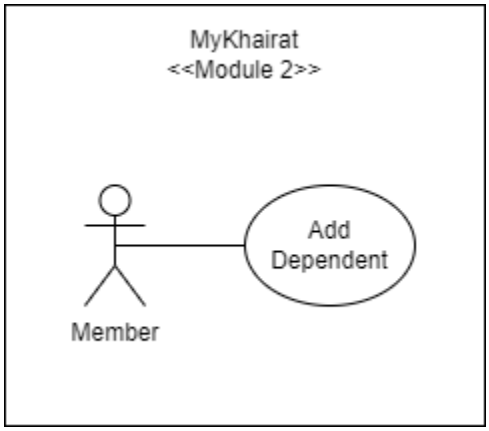


Diagram 4.3.1: Add Dependent Use Case

4.3.1. Description and Priority

This use case is initiated by the actor Member. It enables the member to add new dependents by filling required details. This use case is of high priority.

4.3.2. Flow of Events

4.3.2.1. Basic Flow

4.3.2.1.1. The Member shall click the Add Symbol (+) button on the ‘Tanggungan’ Page.

4.3.2.1.2. The Member shall fill in all required details.

4.3.2.1.3. The Member shall click on the ‘Tambah Tanggungan’ button.

4.3.2.1.4. System shall verify the validity of input.

4.3.2.1.5. System shall display a message stating that info will be verified after a few days. The use case ends.

4.3.2.2. Alternative Flow - None

4.3.2.3. Exception Flow - Start at 4.3.2.1.3.

4.3.2.3.1. Details inputs are invalid.

4.3.2.3.2. System shall display an error message and prompt the member to fill in valid details.

4.3.3. Precondition(s)

The Member has logged in.

4.3.4. Post condition(s)

The Member will be directed to the ‘Tanggungan’ Page after clicking ‘OK’ on the message displayed. No changes shown on the Dependent Page until the information is verified.

4.3.5. Notes/Issues

None.

4.3.6. Activity Diagram

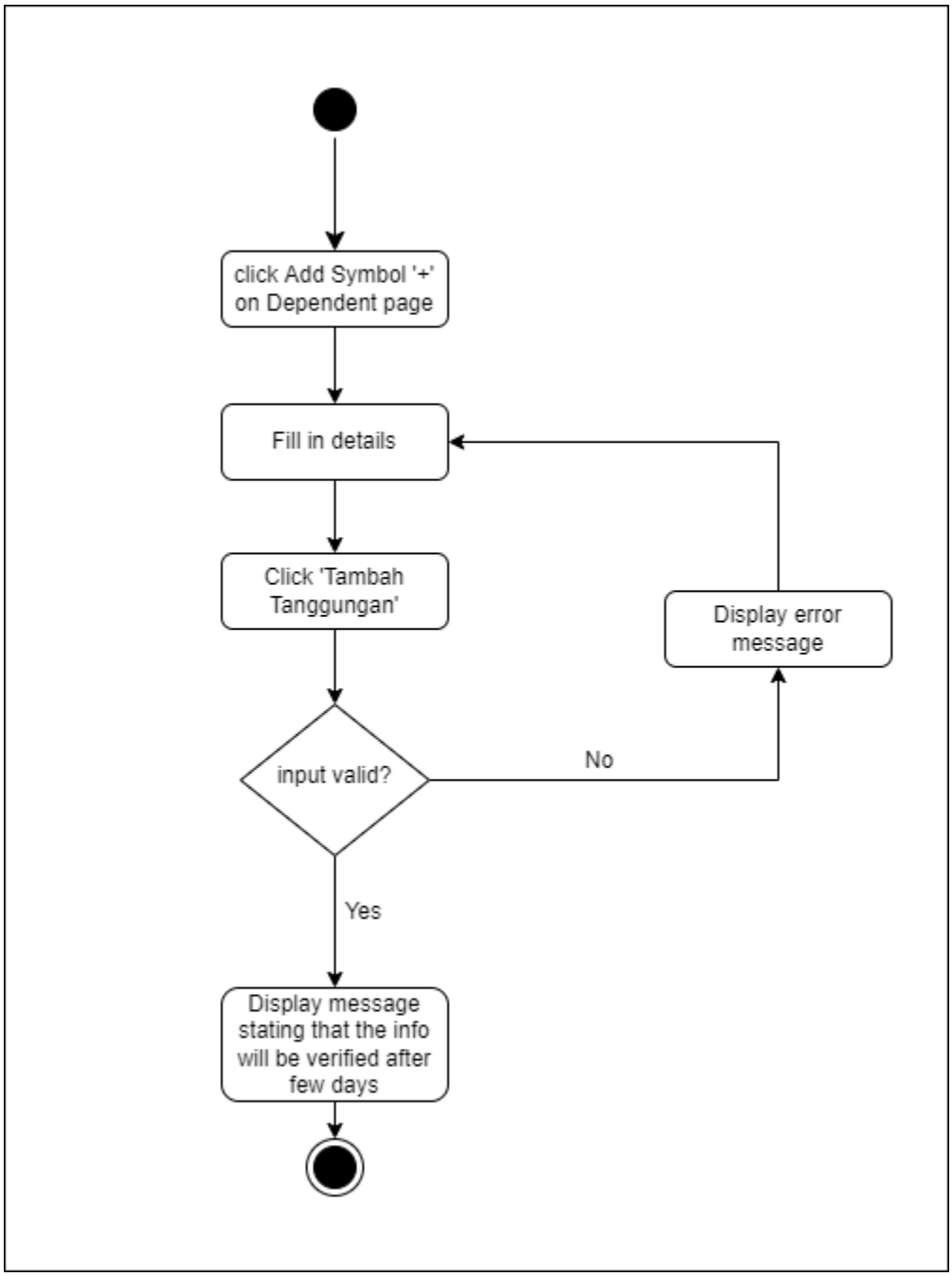


Diagram 4.3.6.1: Add Dependent Activity Diagram

4.4. Validate Info (REQ13)

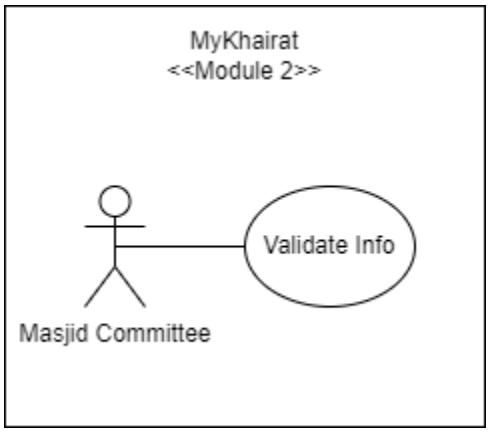


Diagram 4.4.1: Validate Info Use Case

4.4.1. Description and Priority

This use case is handled by the actor Masjid Committee. It enables the committee to validate the Member's dependent information. This use case is of high priority.

4.4.2. Flow of Events

4.4.2.1. Basic Flow

4.4.2.1.1. The Masjid Committee shall click 'Semak Tanggungan' on 'Halaman Utama' page

4.4.2.1.2. The Masjid Committee shall click on 'Lihat Butiran' button on each list of names

4.4.2.1.3. System shall display dependent details with buttons 'Terima' and 'Tolak'

4.4.2.1.4. System shall update and add the new details in Dependent Data

4.4.2.1.5. System shall displays message state that dependent details accepted

4.4.2.1.6.

4.4.2.2. Alternative Flow - Start at 4.4.2.1.3. ('Tolak' button is clicked)

4.4.2.2.1. System shall delete the new details. No changes in Dependent Data will be made

4.4.2.2.2. System shall displays message state that dependent details rejected

4.4.2.3. Exception Flow - none

4.4.3. Precondition(s)

New ‘Tanggungan’ is added by the Member.

4.4.4. Post condition(s)

New ‘Tanggungan’ details are shown on the Member ‘Tanggungan’ page.

4.4.5. Notes/Issues

State here.

4.4.6. Activity Diagram

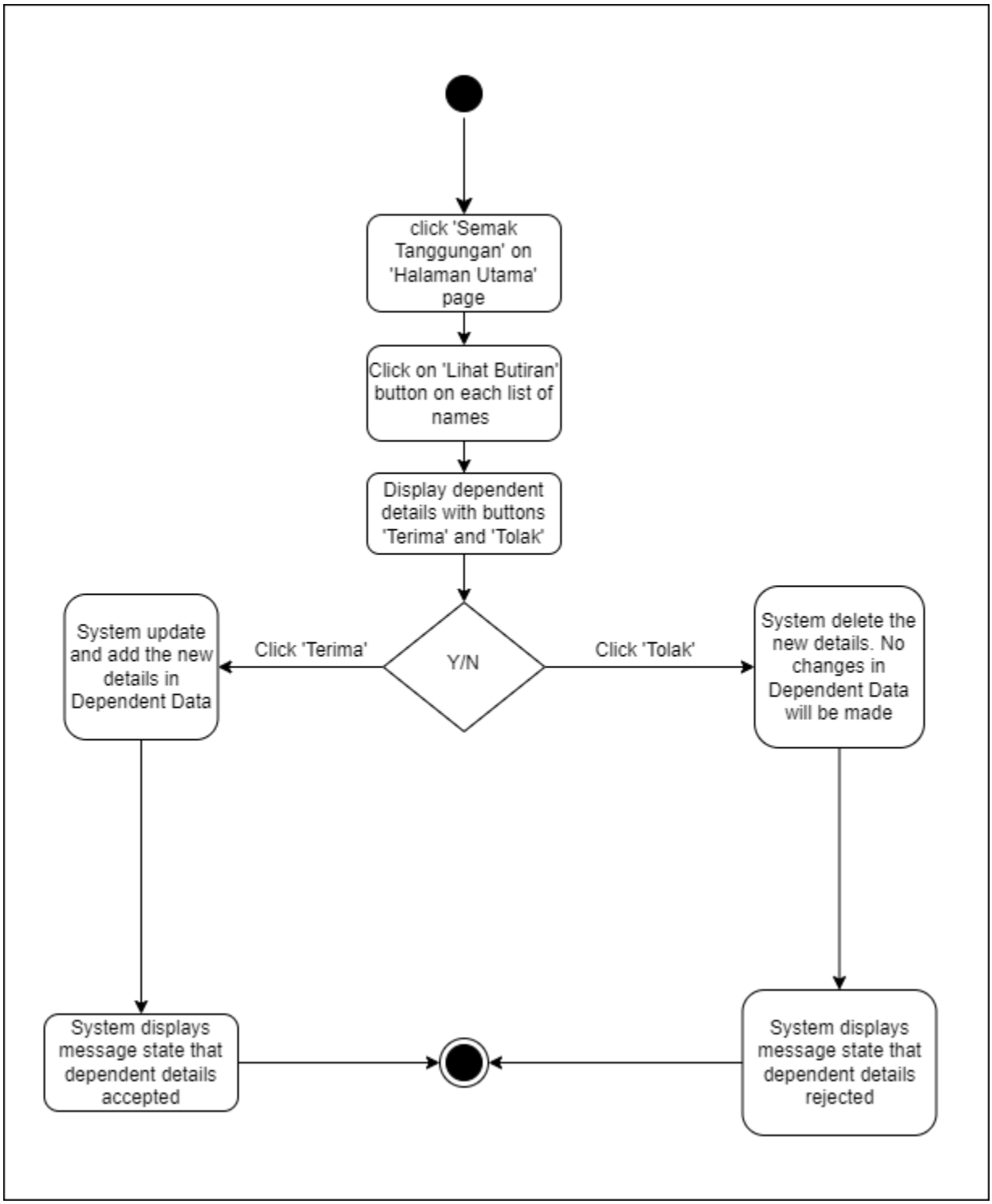


Diagram 4.3.6.1: Validate Info Activity Diagram

4.5. Search Members (REQ18)

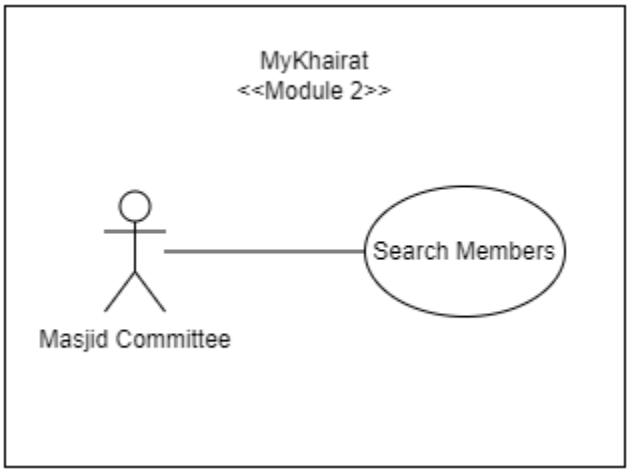


Figure 4.6.1: Search Members Use Case

4.5.1. Description and Priority

This use case is initiated by the actor Masjid Committee. This use case enables the Masjid Committee to search for a specific member by entering the member's key information. The list of related members will be displayed upon completion of this use case. This use case is of low priority.

4.5.2. Flow of Events

4.5.2.1. Basic Flow

4.5.2.1.1. The Masjid Committee shall insert the member's name, Identity Card (IC) number or user ID on the textfield.

4.5.2.1.2. The Masjid Committee shall click on the Search button.

4.5.2.1.3. System shall display the list of members which match the inserted information. The use case ends.

4.5.2.2. Alternative Flow: No matches found

4.5.2.2.1. The Masjid Committee shall insert the member's name, Identity Card (IC) number or user ID on the textfield.

4.5.2.2.2. The Masjid Committee shall click on the Search button.

4.5.2.2.3. System shall display an error message. The use case ends.

4.5.2.3. Exception Flow - Not Applicable

4.5.3. Precondition(s)

The Masjid Committee has logged in.

4.5.4. Post condition(s)

List of related members (if available) or an error message will be displayed.

4.5.5. Notes/Issues

None.

4.5.6. Activity Diagram

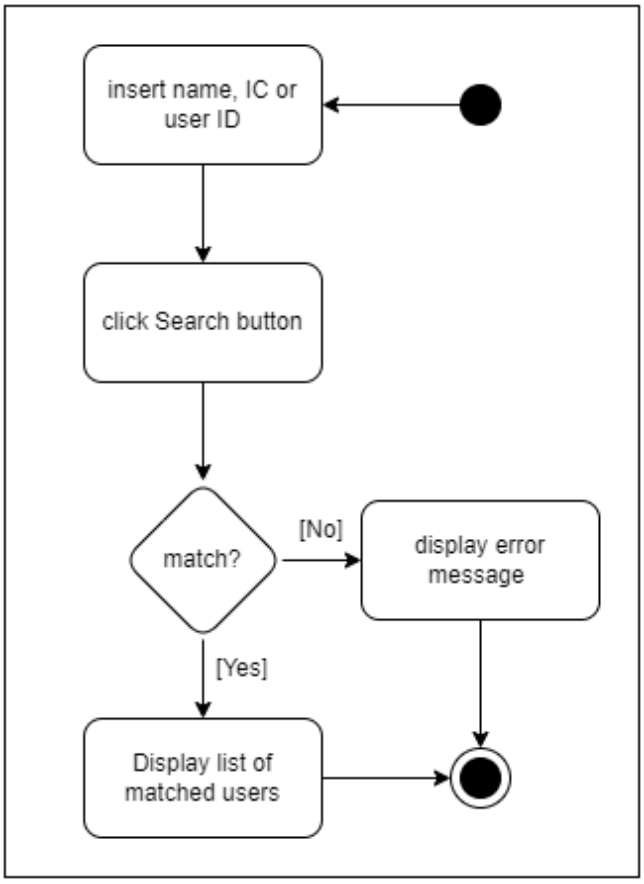


Figure 4.3.6.1: Search Members Activity Diagram

5. Other Non-functional Requirements

5.1. Performance Requirements

The system needs to show the result of the search within 2 seconds.

5.2. Safety Requirements

The system will ask for confirmation each time the members update their profile.

5.3. Security Requirements

Only the users can access and edit their own information.

5.4. Software Quality Attributes

5.4.1. Confidentiality

Only users know the password to their account and only them can change their own password.

5.4.2. Accountability

User is responsible for each information that they have given.

5.4.3. Usability

The software is user-friendly and easy to be learned by the adults with a simple and easy to understand UI.

5.4.4. Flexibility

The software can be used using any Android smartphone and tablet .

5.4.5. Integrity

Users must give real information during registration and fake accounts are not allowed. Each registered account will be checked for its authenticity.

6. Other Requirements

6.1. Legal requirement

All the information is private and confidential, masjid can only know the basic information of the user.

7. Appendix A: Glossary

Terms	Definitions
Android Studio	The official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ software and designed specifically for Android development.
Flutter	An open source framework by Google to create beautiful, natively compiled applications for mobile, web, and desktop from a single codebase.
Github	A code hosting platform that allows hosting and collaborations of software development and version control.
Laravel	A PHP web application framework with expressive, elegant syntax. It eases common tasks (authentication, routing, etc.) used in web projects.
MySQL	An open source relational database management system. It creates a database for storing and manipulating data, defining the relationship of each table.
Visual Studio Code	An efficient code editor that is optimised for performing application development tasks such as building, debugging and others.
Windows	A graphical operating system, developed by Microsoft Corporation, which allows the use of a computer.
XAMPP	A popular cross-platform web server which helps programmers have the ability to serve web pages on the World Wide Web while developing and testing their code on a local web server.

8. Appendix B : Sequence Diagram

Edit Profile (REQ10)

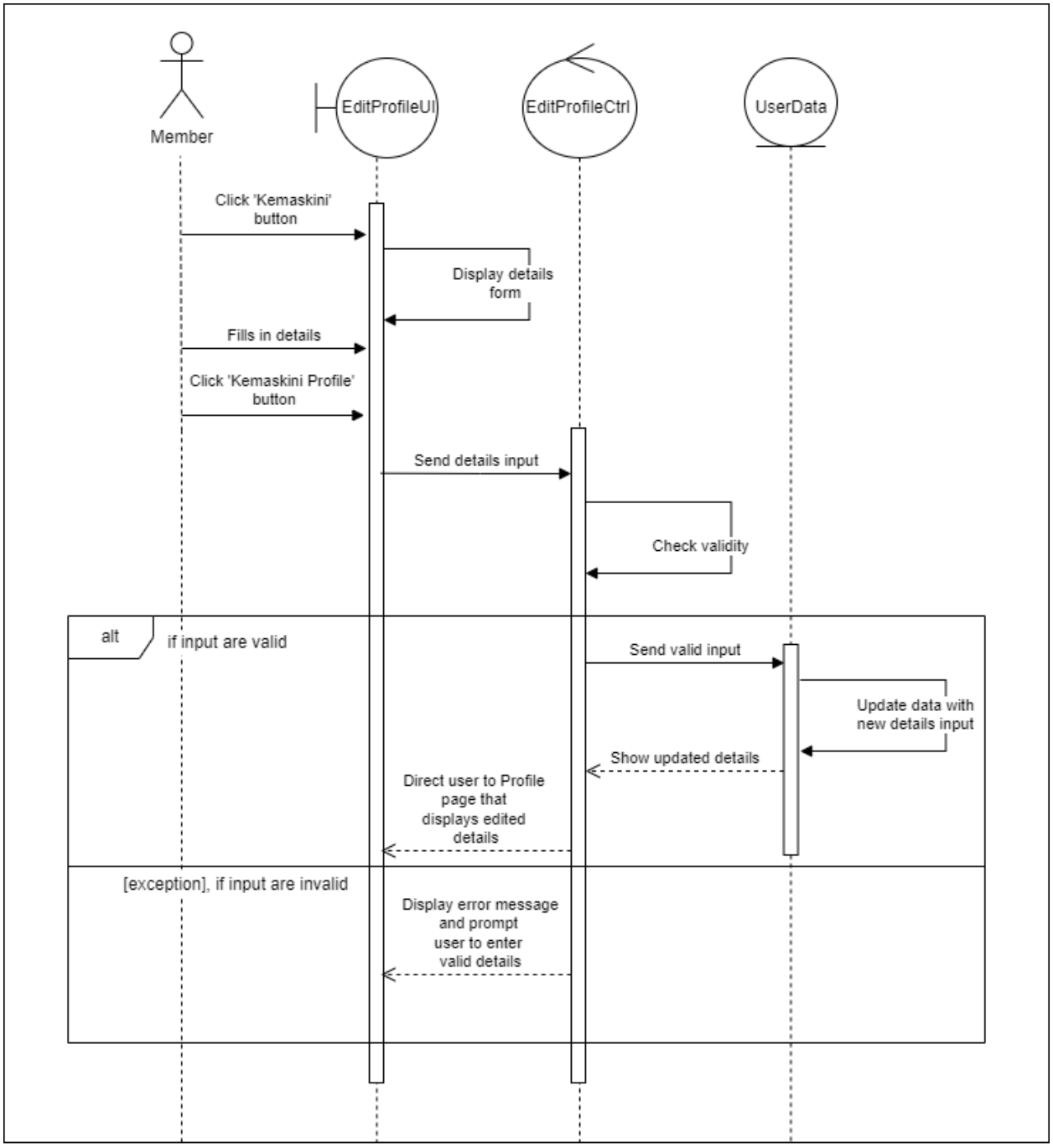


Diagram 8.1

View Dependent Record (REQ11)

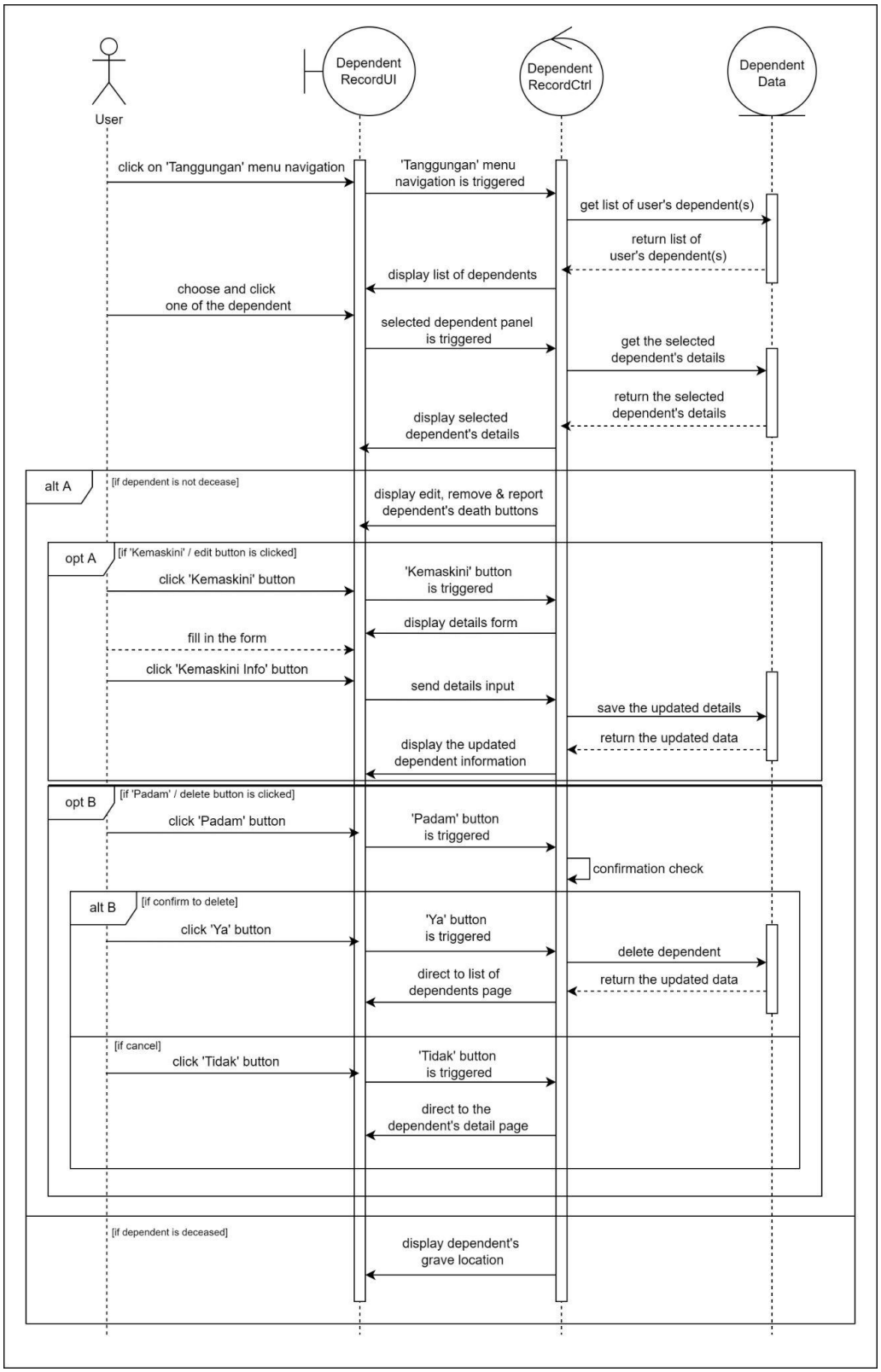


Diagram 8.2

Add Dependent (REQ12)

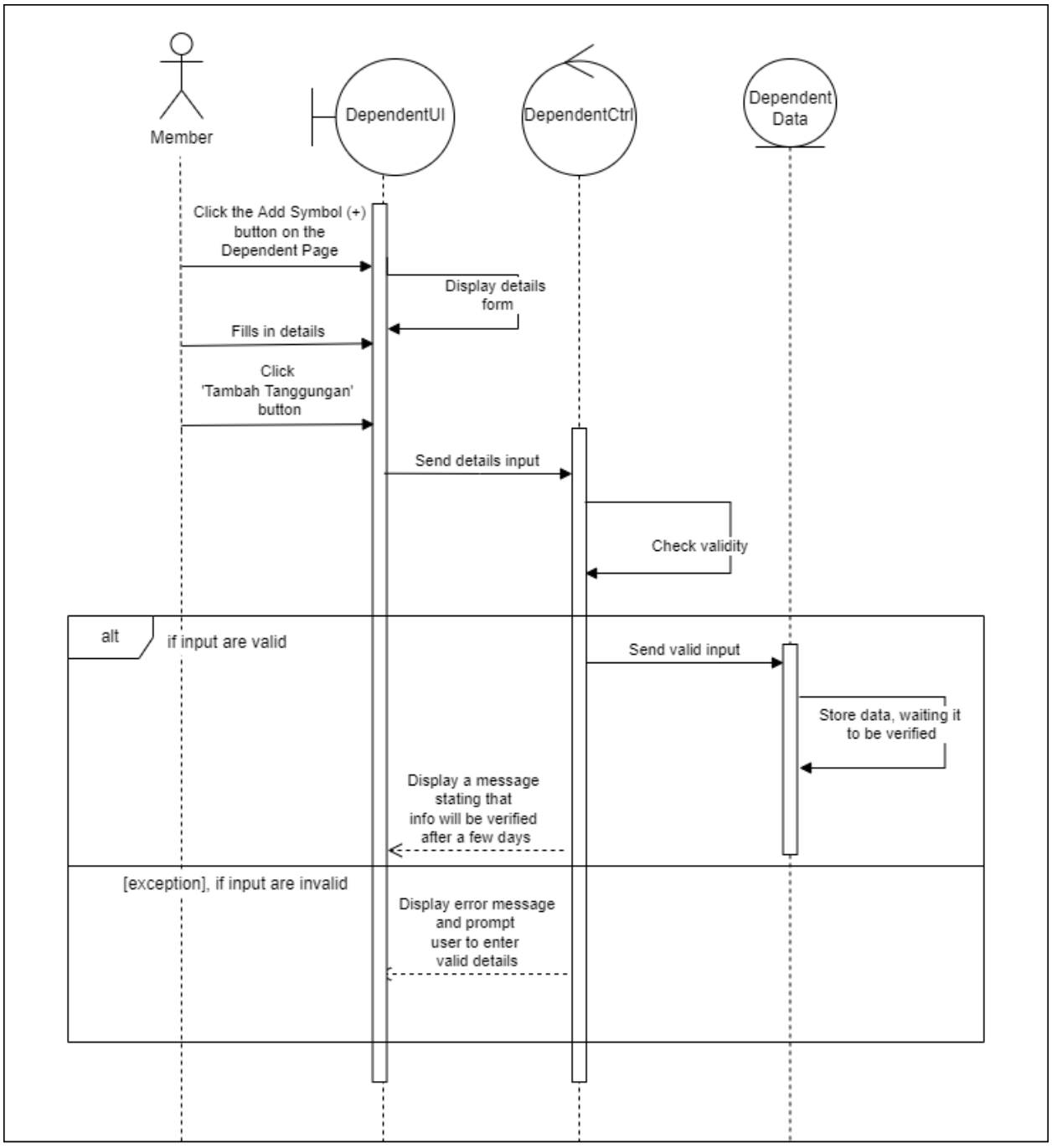


Diagram 8.3

Validate Info (REQ13)

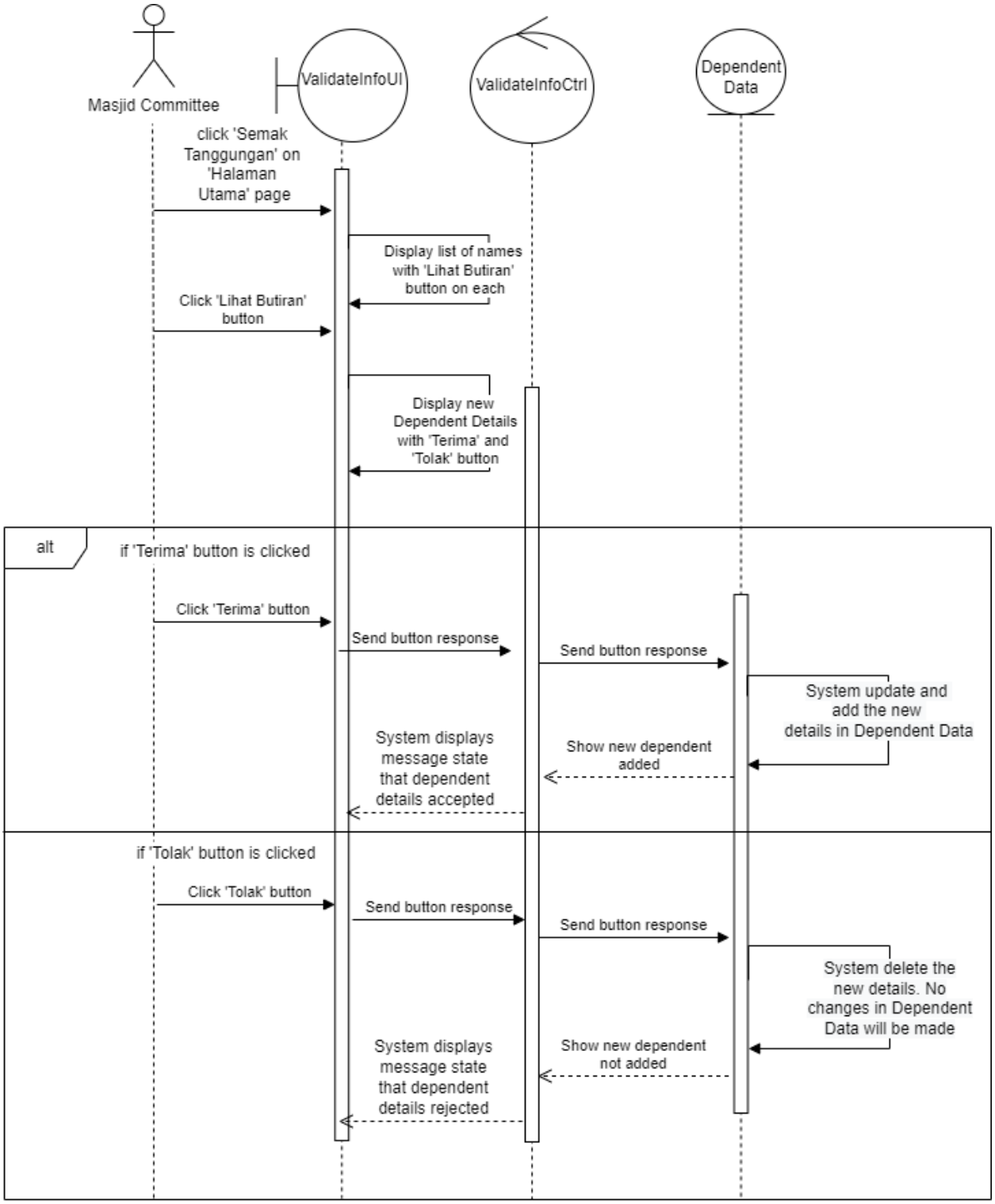


Diagram 8.4

Search Members (REQ18)

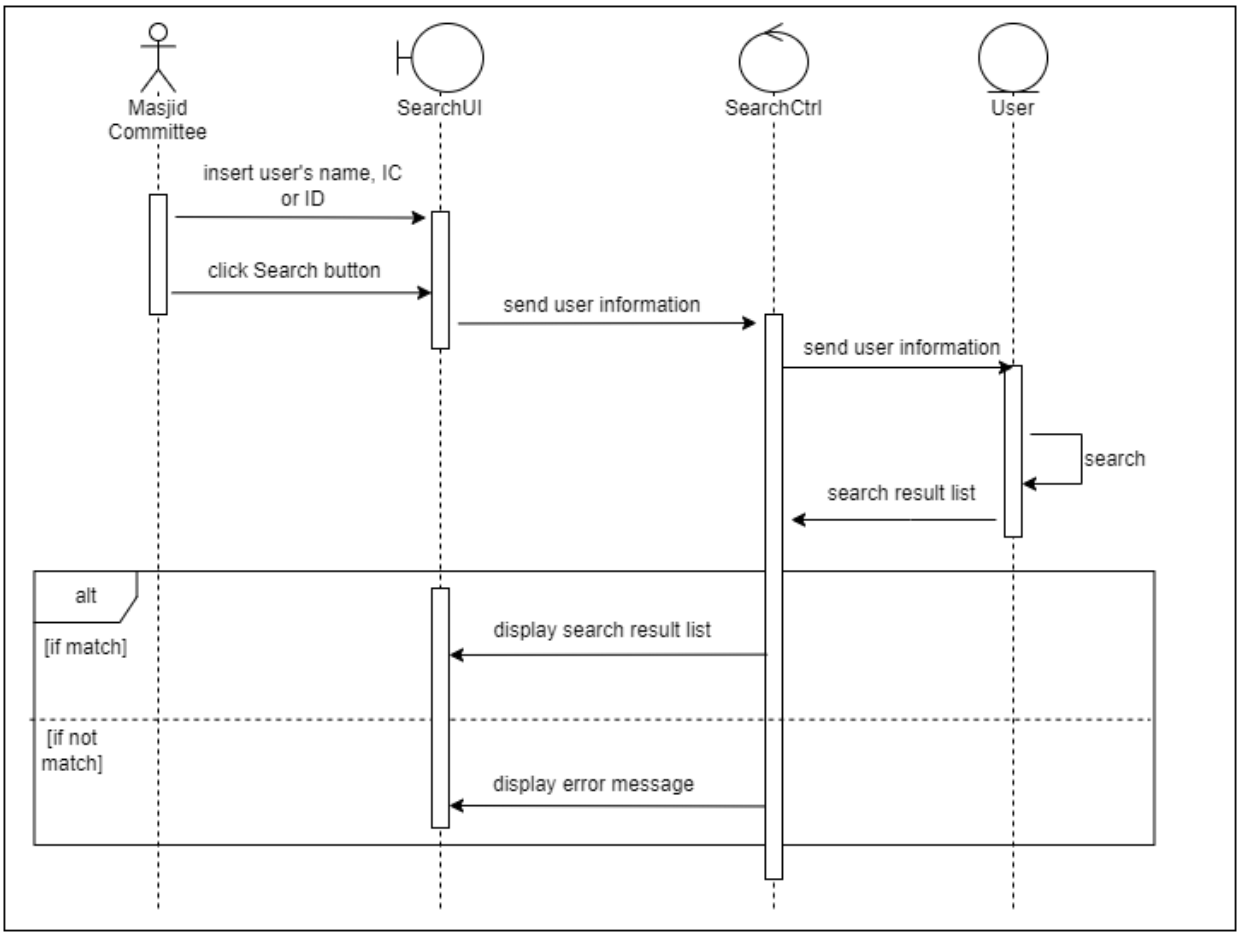


Diagram 8.6

9. Appendix C : Task Distribution

No.	Task	Name
1	Introduction	Nurul Aina binti Ariffin
2	Overall Description	Nor Syafiqa binti Abd Rahman Mohamad Ramzuzzhini bin Mohamad Ramlee
3	External Requirements	Safraa Khairunnisa binti Rahim

		Muhammad Aiman bin Mohd Rahimi
4	System Features	Nor Syafiqa binti Abd Rahman Nurul Aina binti Ariffin Safraa Khairunnisa binti Rahim
5	Other Non-functional Requirements	Muhammad Aiman bin Mohd Rahimi
6	Appendix	Mohamad Ramzuzzhini bin Mohamad Ramlee