

**Bachelor of Computer Science**

**(Digital System Security)**

CSCI321 – Final Year Project

SOFTWARE REQUIREMENT

SPECIFICATION

**Project Particulars**

|  |  |
| --- | --- |
| Supervisor | Dr Ta Nguyen Binh Duong |
| Project Group | SS18/1F |
| Project Title | Two Factor Authentication |

**Project Team’s Particulars**

|  |  |  |
| --- | --- | --- |
| Student Number | Student Name | Email Address |
| 5363536 | Koh Hong Wei | hwkoh003@mymail.sim.edu.sg |
| 5710923 | Chua Han Ming Adler | hmachua002@mymail.sim.edu.sg |
| 5711356 | Ong Wei Hao | whong012@mymail.sim.edu.sg |

****

**A secure file locking application**

**Document Control**

Title: Project Proposal

Document Name: FYP\_ProjectProposal

**Distribution List**

|  |  |  |
| --- | --- | --- |
| Name | Title/Role | Where |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Record of Revision**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Revision Date | Description | Section Affected | Changes Made By | Version after Revision | Notes |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Table of Contents

[1. Introduction 5](#_Toc508888841)

[1.1. Background 5](#_Toc508888842)

[1.2. Project Scope 5](#_Toc508888843)

[1.3 Intended Audience and Reading Suggestions 5](#_Toc508888844)

[2. Product Overview Description 5](#_Toc508888845)

[2.1 Product Perspective 5](#_Toc508888846)

[2.2 Product Features 5](#_Toc508888847)

[2.2.1 Locking File 5](#_Toc508888848)

[2.2.2 Unlocking file 5](#_Toc508888849)

[2.2.3 Secure OTP generation 5](#_Toc508888850)

[2.3 User Classes and Characteristics 5](#_Toc508888851)

[2.3.1 Physical Actors 5](#_Toc508888852)

[2.3.2 System Actors 5](#_Toc508888853)

[2.4 Operating Environment 5](#_Toc508888854)

[2.5 Implementation Constraints 5](#_Toc508888855)

[2.6 User Documentation 5](#_Toc508888856)

[2.7 Assumptions and Dependencies 5](#_Toc508888857)

[3. System Features 5](#_Toc508888858)

[Main Use Case 6](#_Toc508888859)

[Use Case 1.1 Create Account 6](#_Toc508888860)

[Use Case 1.1 Create Account 6](#_Toc508888861)

[Use Case 1.1 Create Account 6](#_Toc508888862)

[Use Case 1.1 Create Account 6](#_Toc508888863)

[Use Case 1.1 Create Account 6](#_Toc508888864)

# 1. Introduction

## 1.1. Background

The purpose of this document is to provide a detailed overview of our software product. This includes the description of our project’s target audience, system features, user interface and both hardware and software requirements.

## 1.2. Project Scope

This project aims to provide users with a way to enhance security on file management on desktop applications. We will be including a two factor authentication mechanism to secure our files from unauthorised usage.

## 1.3 Intended Audience and Reading Suggestions

# 2. Product Overview Description

## 2.1 Product Perspective

Go2FA is an application running on both Windows(PC) and Mobile(Android) platform written in JAVA. It is intended for providing a secure method of protecting a user’s file or folder in their local PC with our provided 2 factor authentication

## 2.2 Product Features

### 2.2.1 Locking File

### 2.2.2 Unlocking file

### 2.2.3 Secure OTP generation

## 2.3 User Classes and Characteristics

### 2.3.1 Physical Actors

### 2.3.2 System Actors

#### 2.3.2.1 SQL Database

## 2.4 Operating Environment

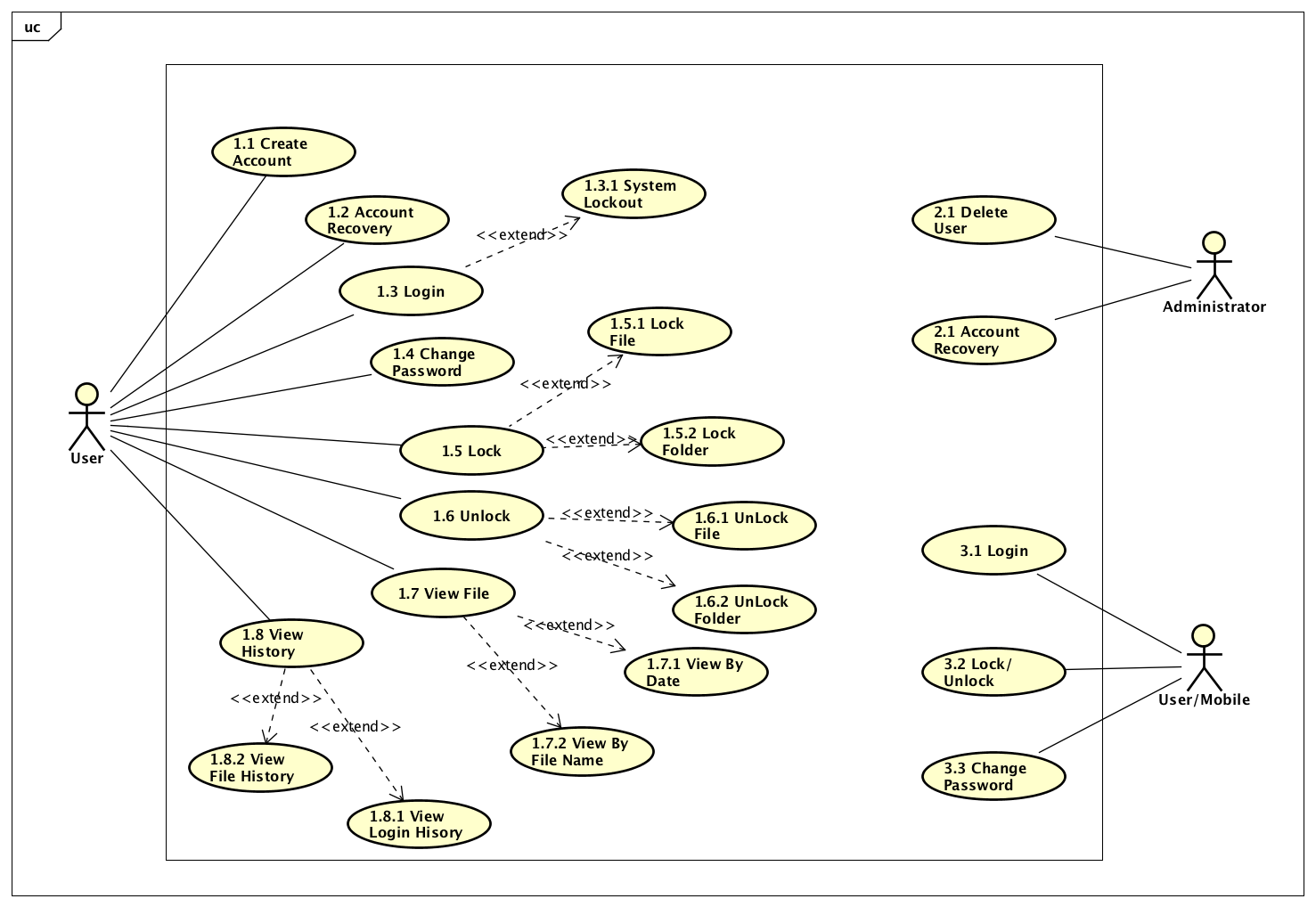
## 2.5 Implementation Constraints

## 2.6 User Documentation

## 2.7 Assumptions and Dependencies

# 3. System Features

## Main Use Case

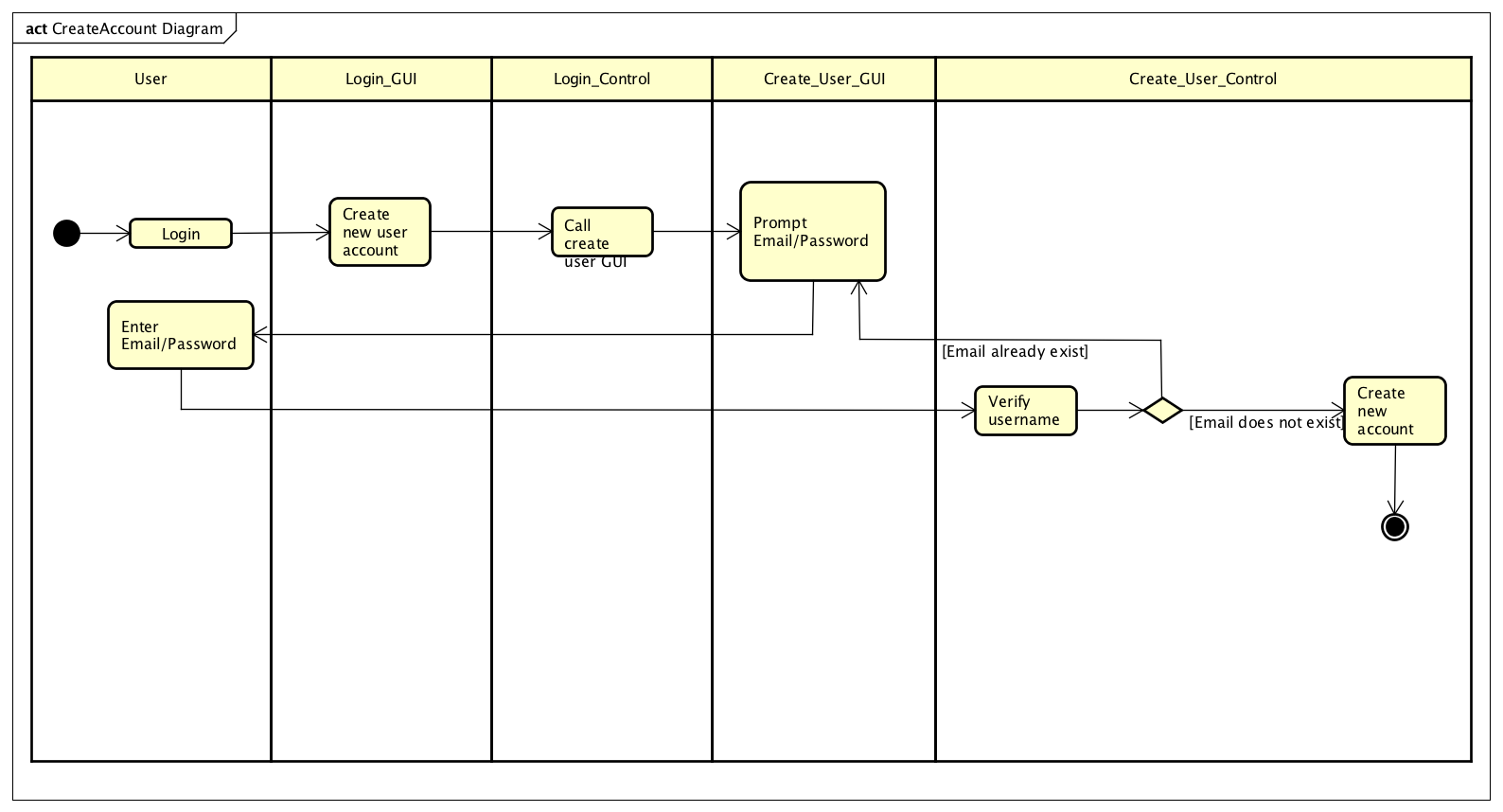


## Create Account

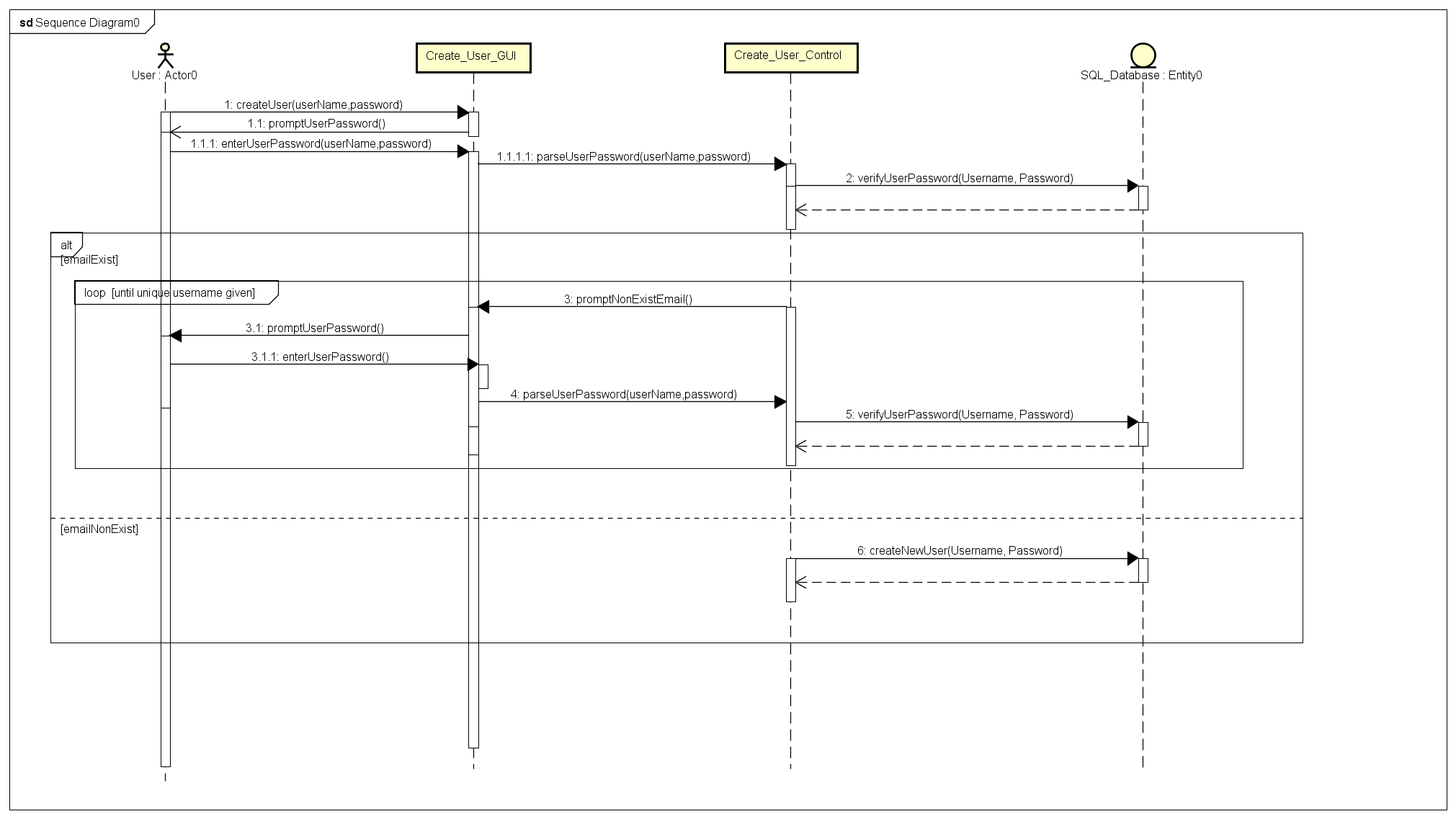
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 1.1 |
| Name | Create Account |
| Description | To allow user to create new account after he download the application for the first time |
| Actor(s) | User |
| Precondition | A user has installed the application and wish to create a new account |
| Main Scenario | Step 1: System prints out a sign-up form with fields to ask for user email address and desired password  Step 2: User submits form and system does validation  Step 2(alternate): If data is invalid, an error message will be displayed, and user is sent back to form for re-entry of right credentials  Step 3: User password will be hashed  Step 4: System updates the SQL Database  Step 5: User is redirected to log in screen |

### Activity Diagram



### Sequence Diagram

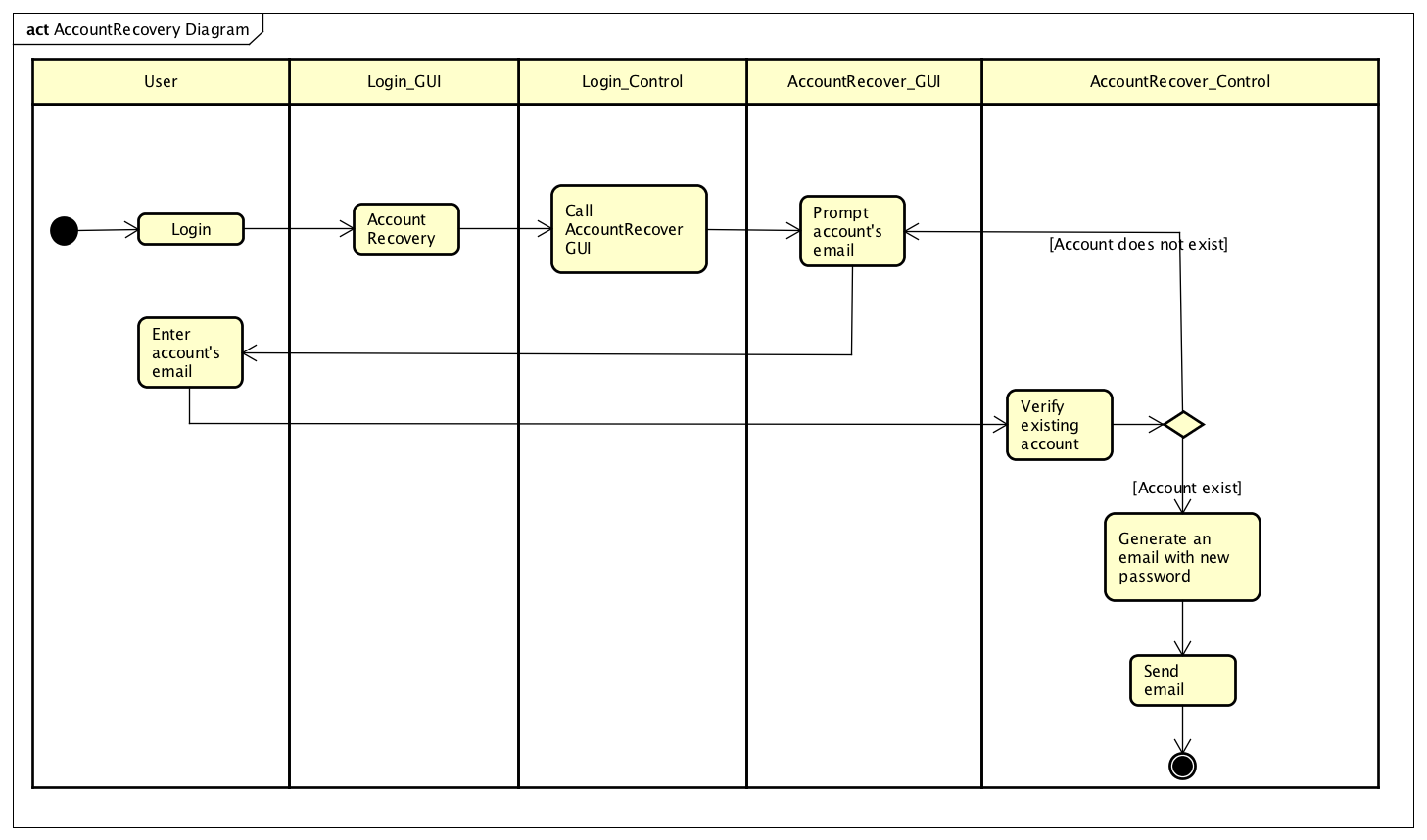


## Account Recovery

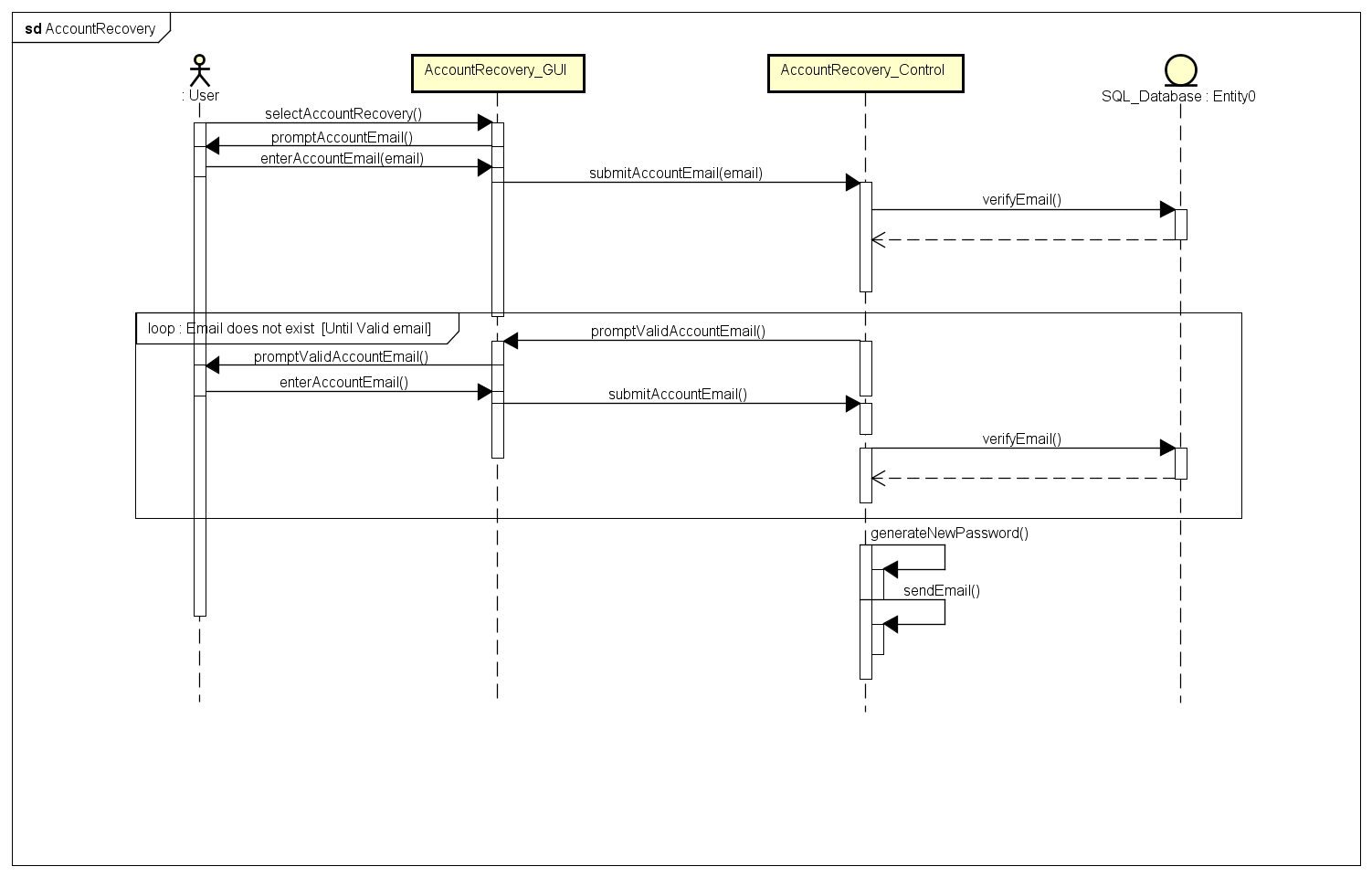
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 1.2 |
| Name | Account Recovery |
| Description | To allow user to recover their lost or forgotten account |
| Actor(s) | User |
| Precondition | The user has a registered account and forgotten password |
| Main Scenario | Step 1: User selects account recovery on Log in screen  Step 2: System prints out a form to prompt for User account email  Step 3: User fills up the email and system do validation  Step 3(alternate): If email is invalid, an error message will be displayed, and user is sent back to form for re-entry of right credentials  Step 4: System will generate new password  Step 5: Email with new generated password will be sent to User email address. |

### Activity Diagram



Sequence Diagram



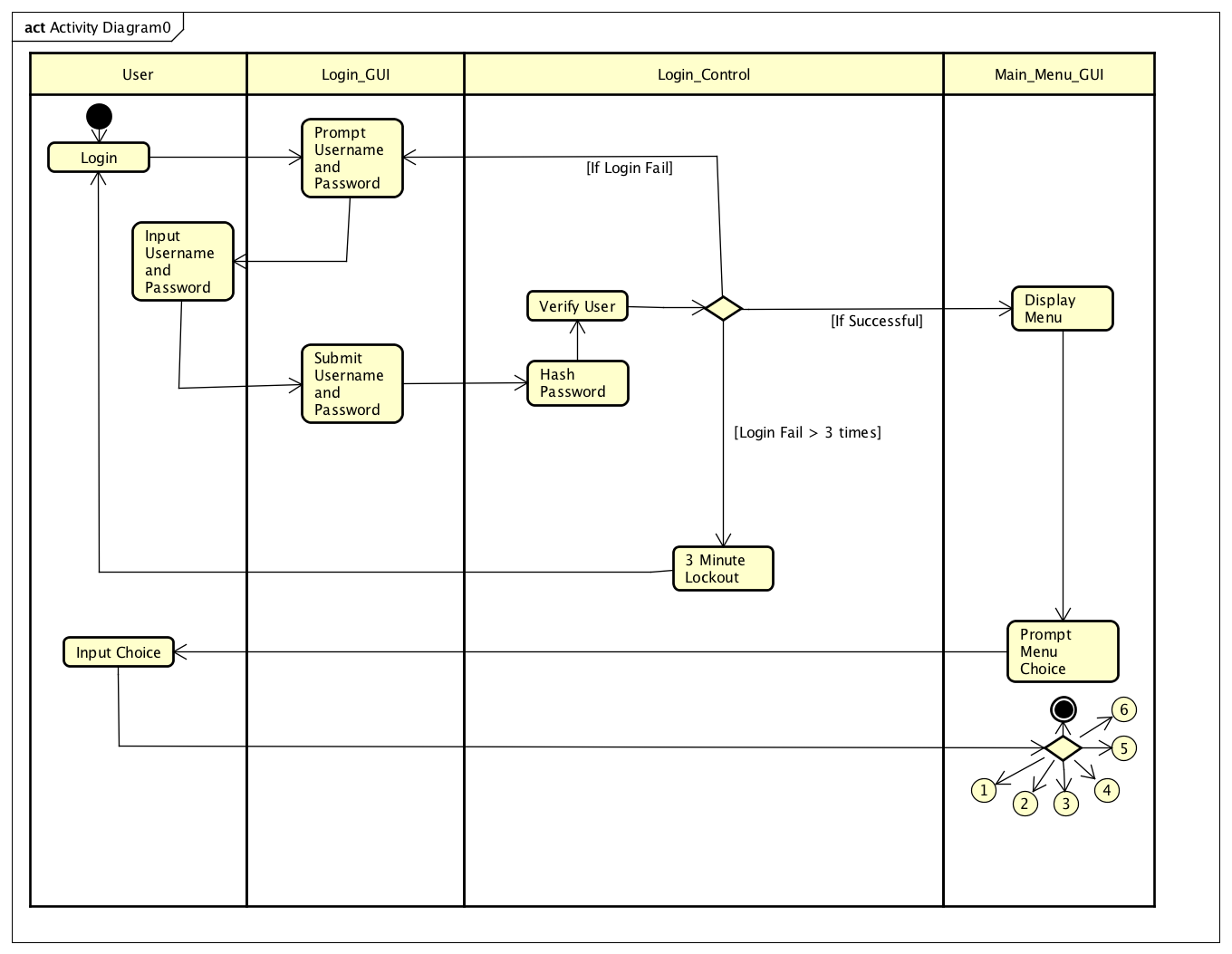
## Login

### Use Case

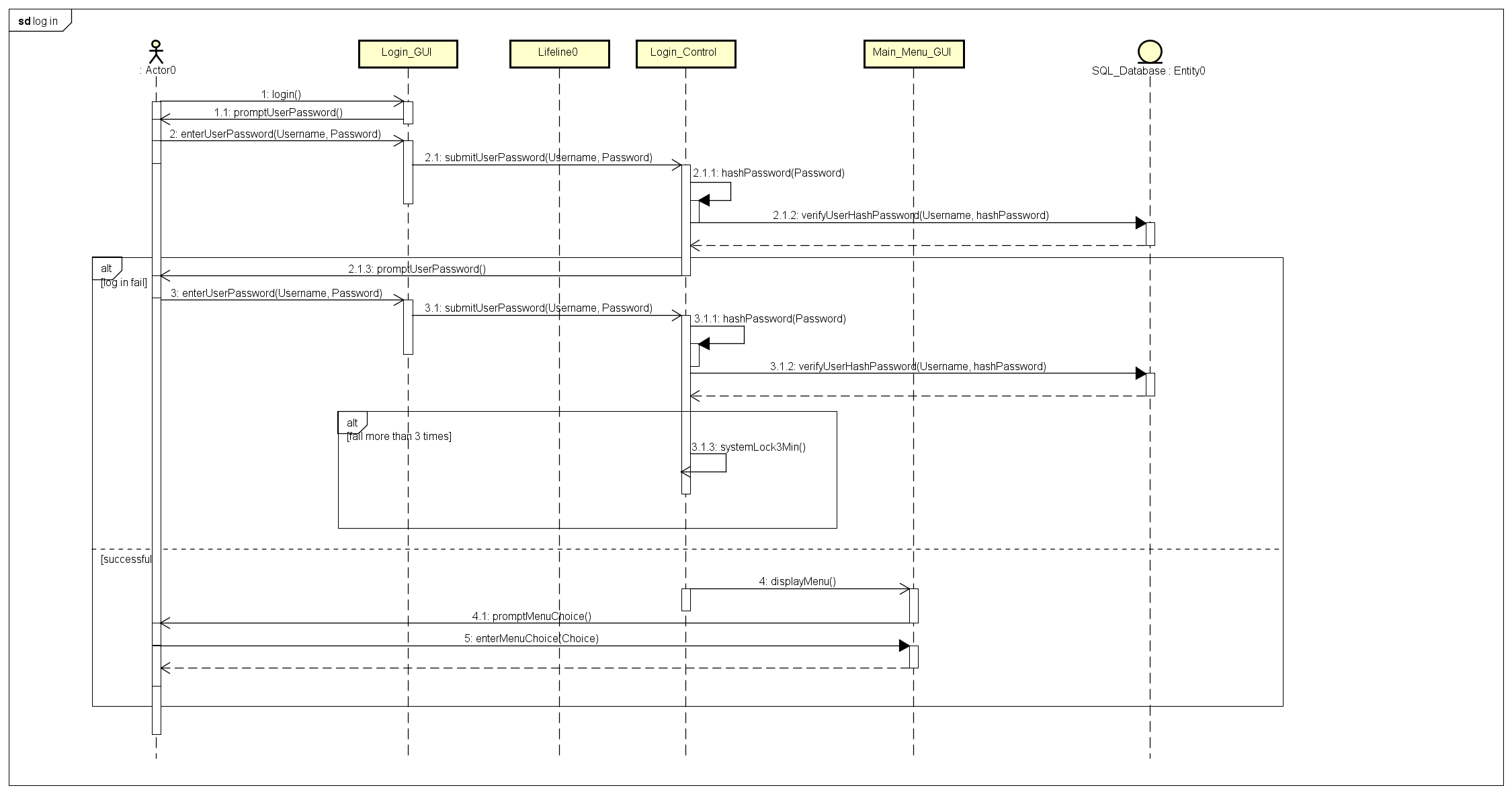
|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 1.3 |
| Name | Login |
| Description | To allow User to log in to the system |
| Actor(s) | User |
| Precondition | The user has a registered account |
| Main Scenario | Step 1: User enters their log in credentials in the log in screen  Step 2: System hash the User password and validates with SQL Database  Step 2(alternate): If log in credentials are wrong, an error message will display, and User will be prompt to enter credentials again  Step 2(alternate): If log in credentials is wrong for 3 times, system will be lock from User for 3 minutes **via U/C 1.3.1**  Step 3: System will display Main Menu  Step 4: User will enter choice from Main Menu |

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 1.3.1 |
| Name | System Lockout |
| Description | Lock User out after several failed log in attempts |
| Actor(s) | User |
| Precondition | The user has 3 failed log in attempts |
| Main Scenario | Step 1: System will lock the user out of the system for 3 minutes  Step 2: System will resume after 3 minutes. |

### Activity Diagram



### Sequence Diagram

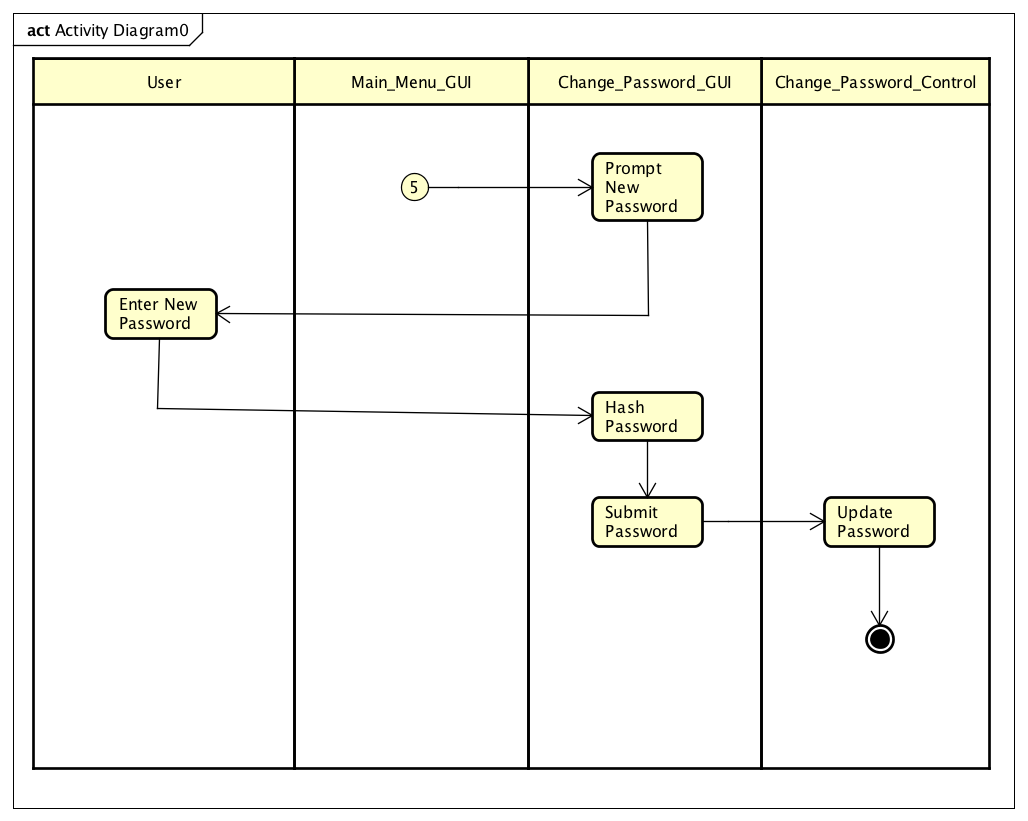


## Change Password

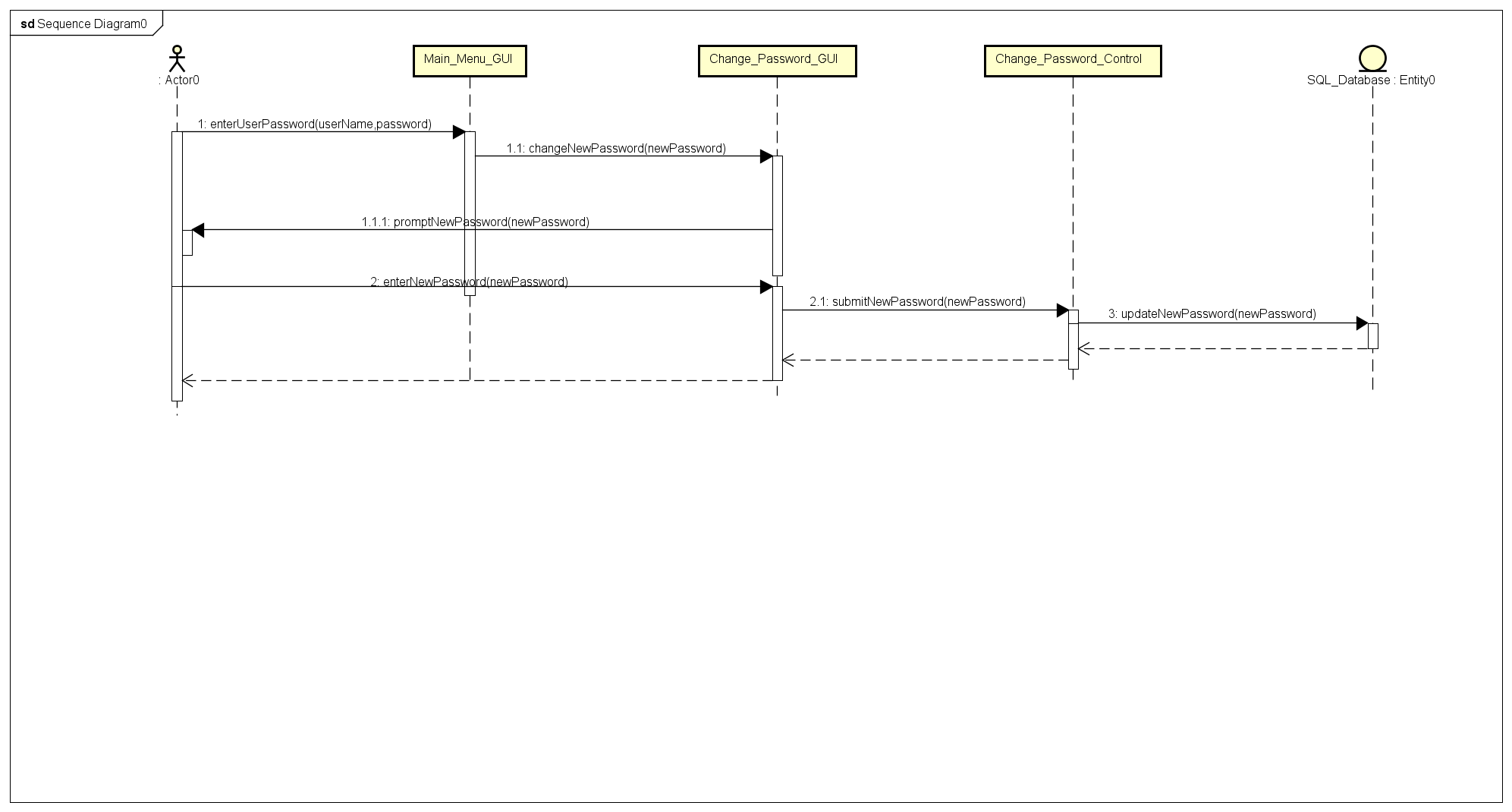
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 1.4 |
| Name | Change Password |
| Description | To allow User to change their password |
| Actor(s) | User |
| Precondition | The user has a registered account and wish to update the current password |
| Main Scenario | Step 1: User enters username and password at Log in screen  Step 2: User selects change password at the main menu  Step 3: System displays a form for user to fill up the new password they desire  Step 4: User enters the new desired password and system do validation  Step 5: System will update the SQL Database with the new password |

### Activity Diagram



### Sequence Diagram

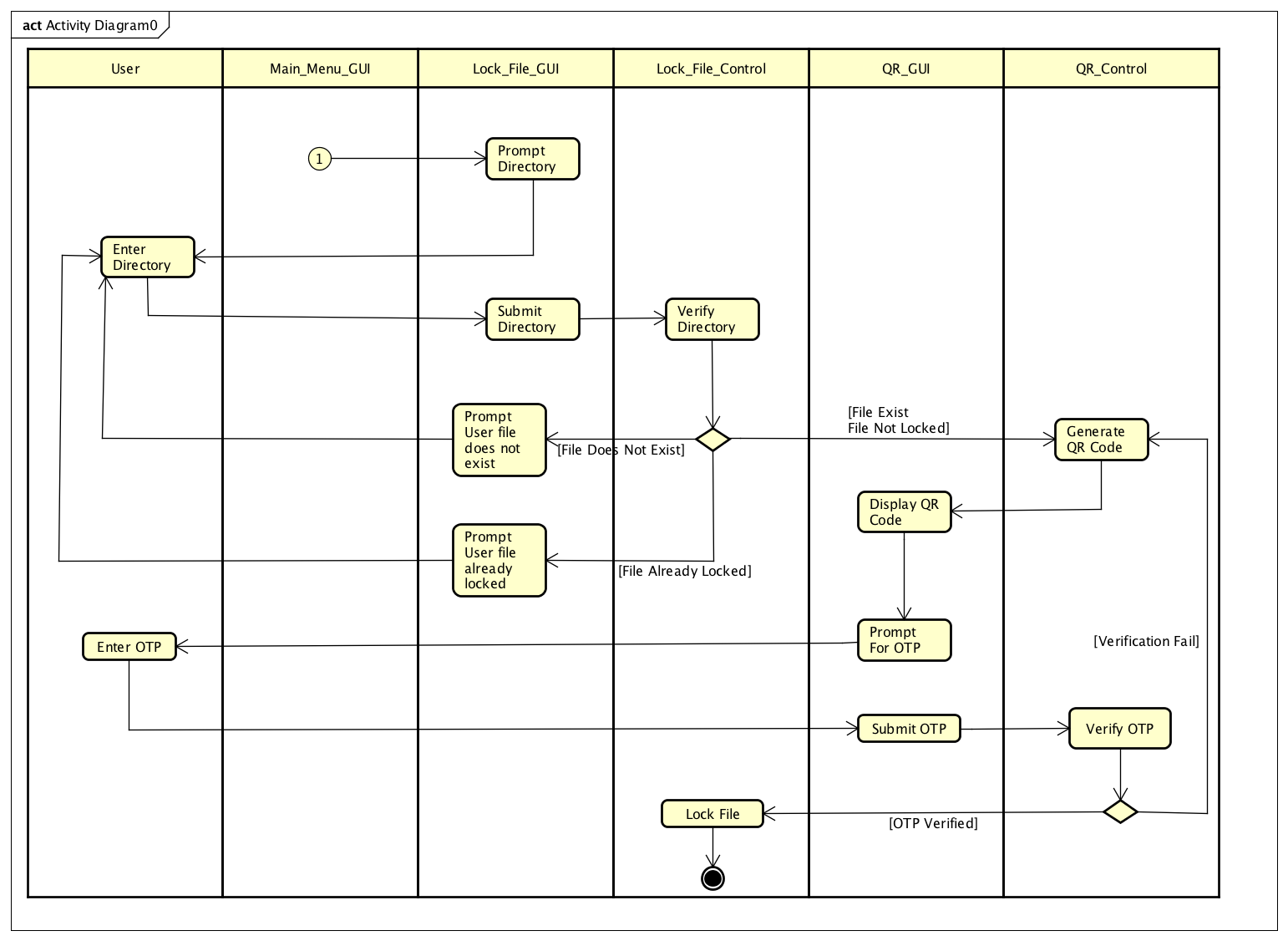


## Lock

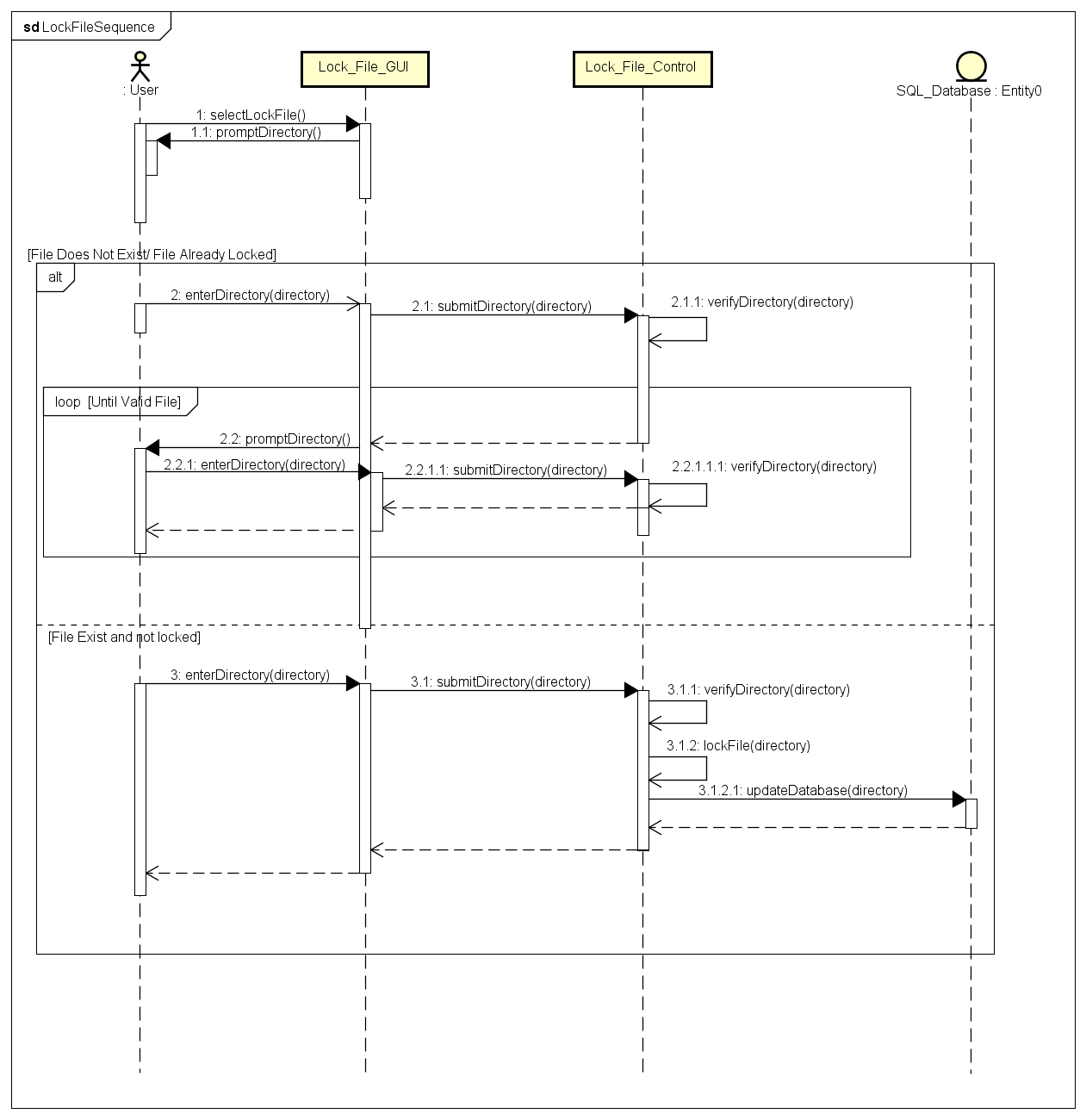
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 1.5 |
| Name | Lock |
| Description | To allow user to lock their folders or files using system cryptography feature |
| Actor(s) | User |
| Precondition | The user has a registered account and has a folder or file that they want to lock |
| Main Scenario | Step 1: User selects Lock file option in the Main Menu.  Step 2: System displays a form to fill up which directory the User has the folder or file  Step 3: User enters the directory and system validates  Step 3(alternate): If directory is invalid, an error message will display, and User will be redirected to form to fill up directory.  Step 3(alternate): If the file or folder in directory is already locked, an error message will display, and User will be redirected to form to fill up the correct directory  Step 4: The system locks the file or folder of the User directory with cryptography feature |

### Activity Diagram



### Sequence Diagram

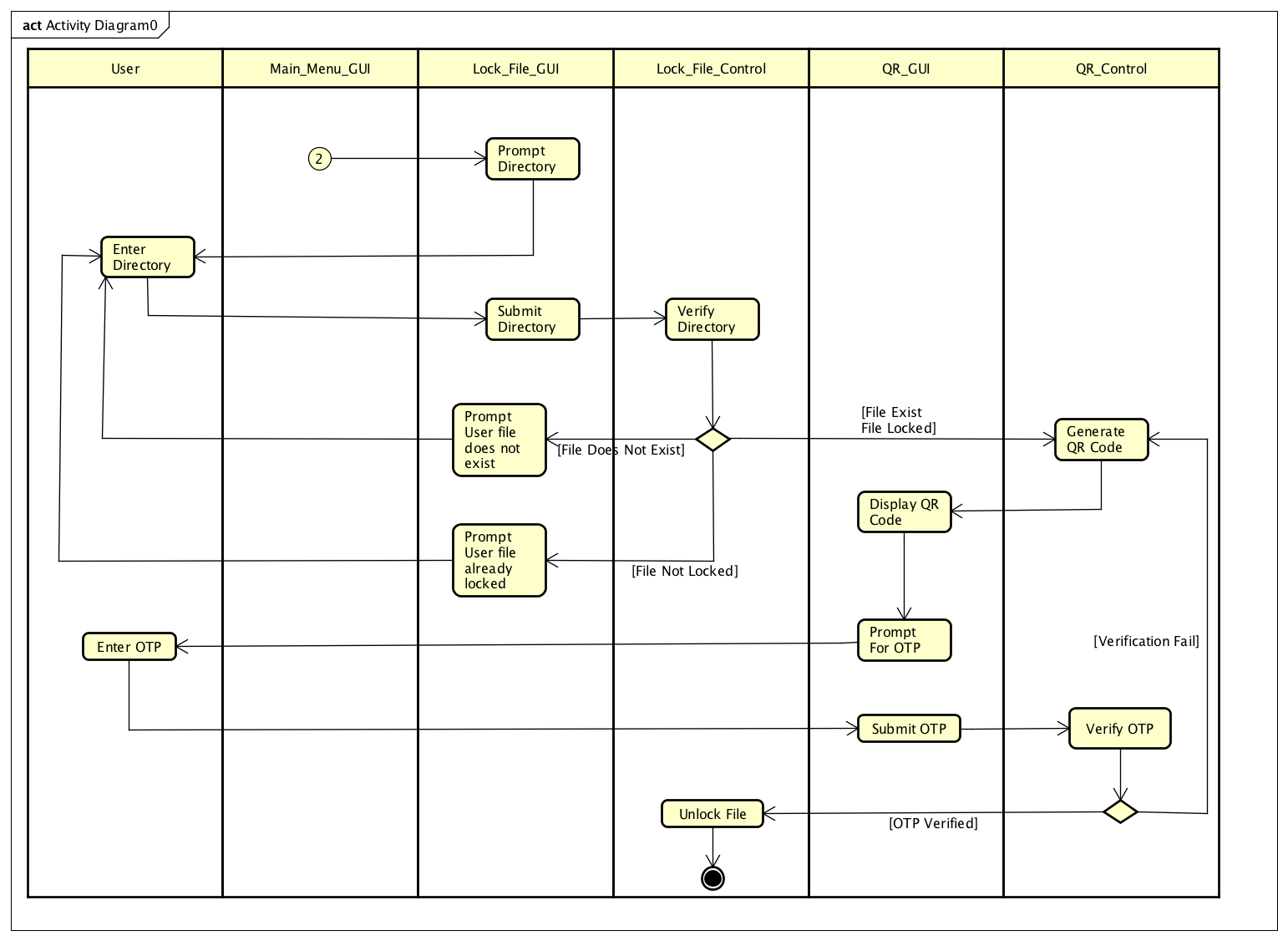


## Unlock

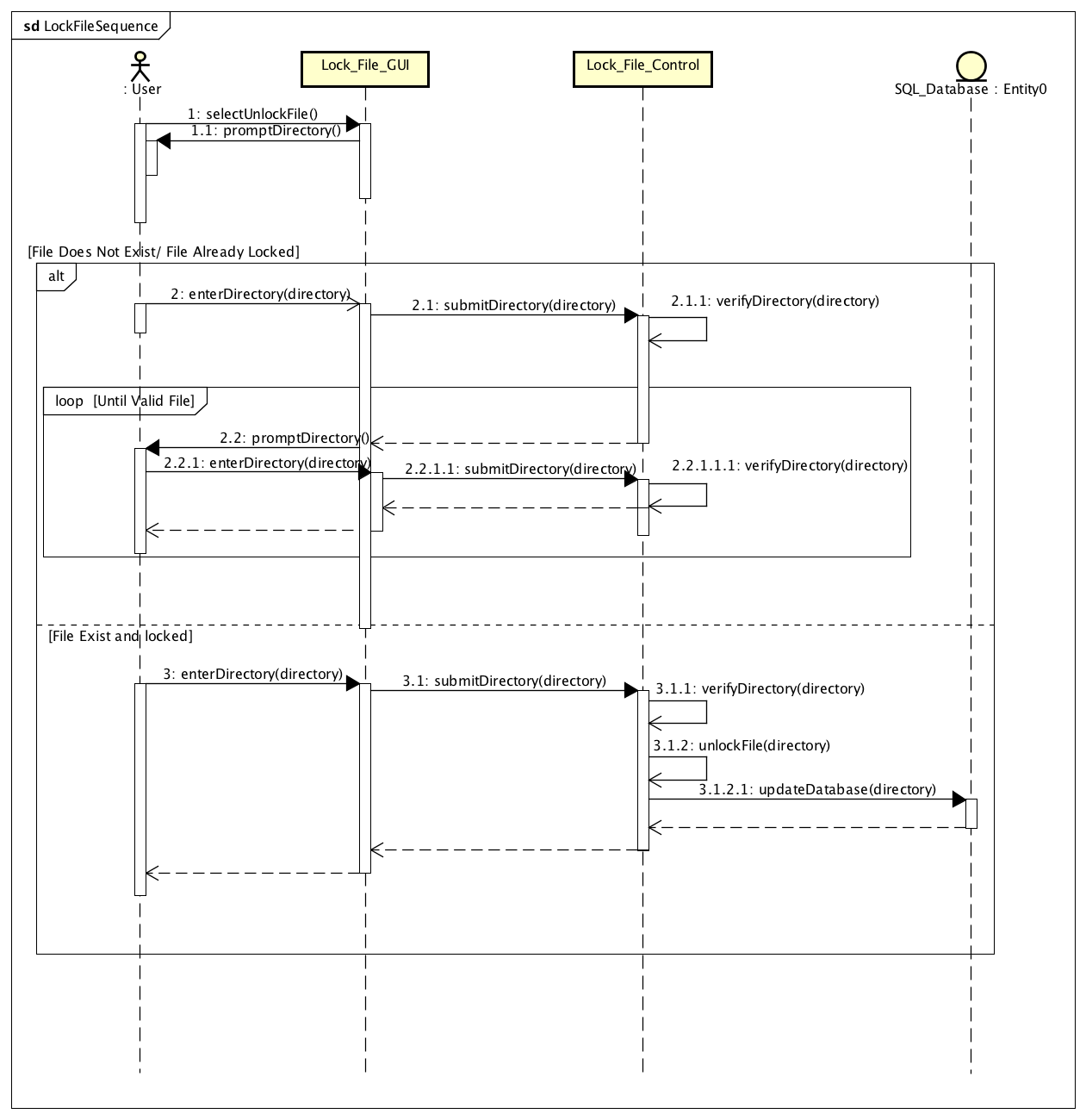
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 1.6 |
| Name | Unlock |
| Description | To allow user to unlock their folders or files using system cryptography feature |
| Actor(s) | User |
| Precondition | The user has a registered account and logged into the system |
| Main Scenario | Step 1: User selects Unlock file option in the Main Menu.  Step 2: System displays a form to fill up which directory the User has the folder or file  Step 3: User enters the directory and system validates  Step 3(alternate): If directory is invalid, an error message will display, and User will be redirected to form to fill up directory.  Step 3(alternate): If the file or folder in directory is not locked, an error message will display, and User will be redirected to form to fill up the correct directory  Step 4: The system unlocks the file or folder of the User directory with cryptography feature |

### Activity Diagram



### Sequence Diagram

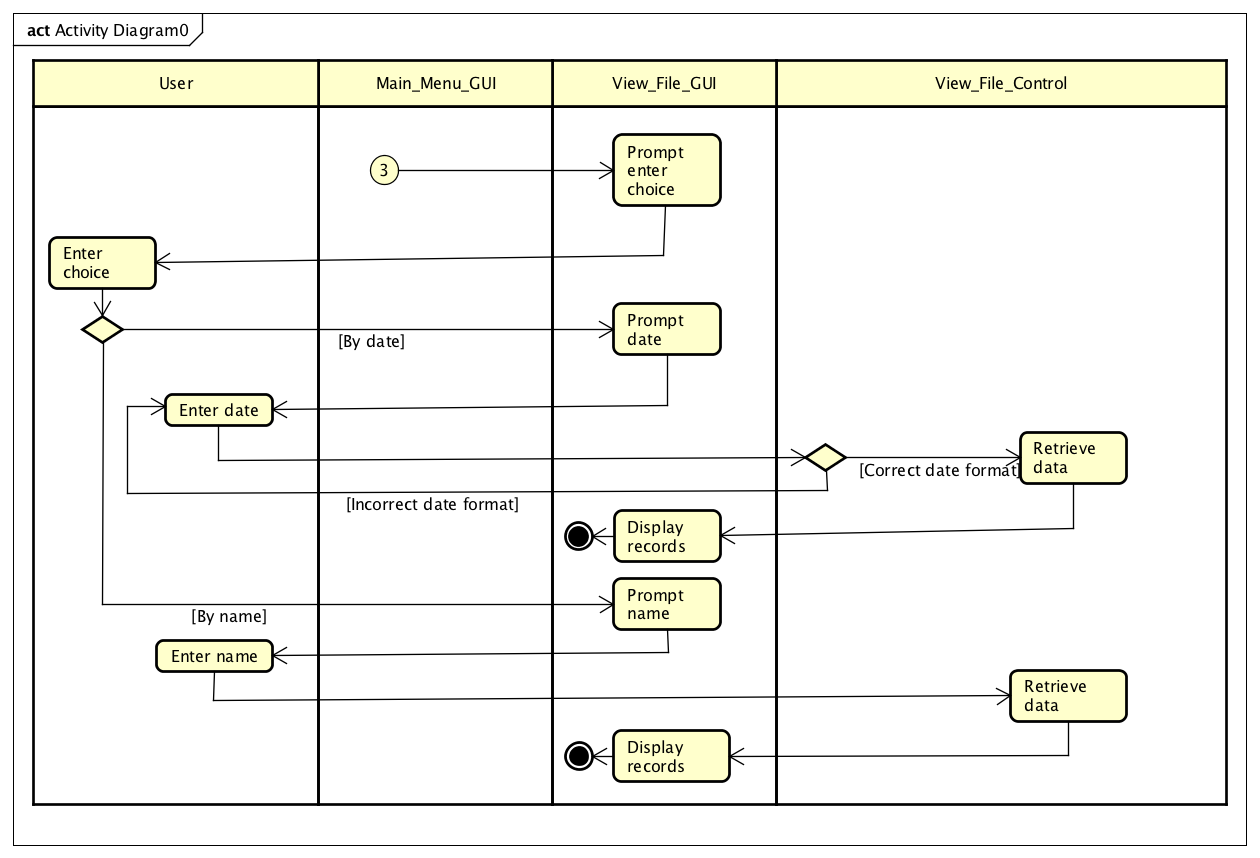


## View File

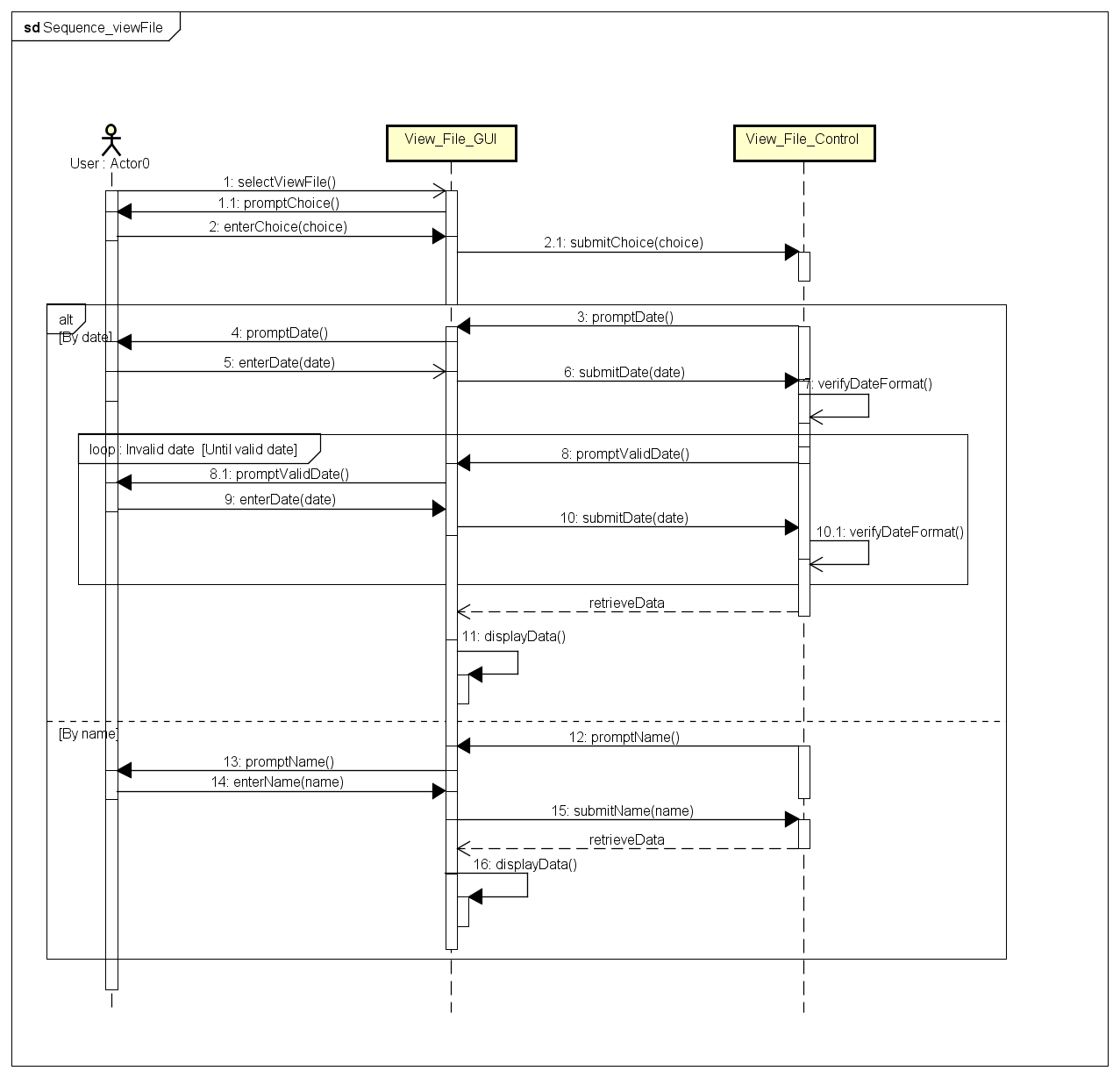
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 1.7 |
| Name | View File |
| Description | To allow user to view the locked files in the system |
| Actor(s) | User |
| Precondition | The user has a registered account and logged into the system |
| Main Scenario | Step 1: User selects View File option in the Main Menu  Step 2: User selects a choice in the view file menu  Step 3: System will display a prompt to User to choose from viewbydate or viewbyname  Step 4(viewbydate): User enters a date from where he locked a file and system validates **via U/C 1.7.1**  Step 4(viewbyname): User enters a date from where he locked a file and system validates **via U/C 1.7.2**  Step 4(alternate): if the date/name the User entered does not contain any locked file, the system will display an error message and prompt the User to enter a valid date  Step 5: When date/name is validated, system will retrieve the data and display the data to the User. |

### Activity Diagram



### Sequence Diagram

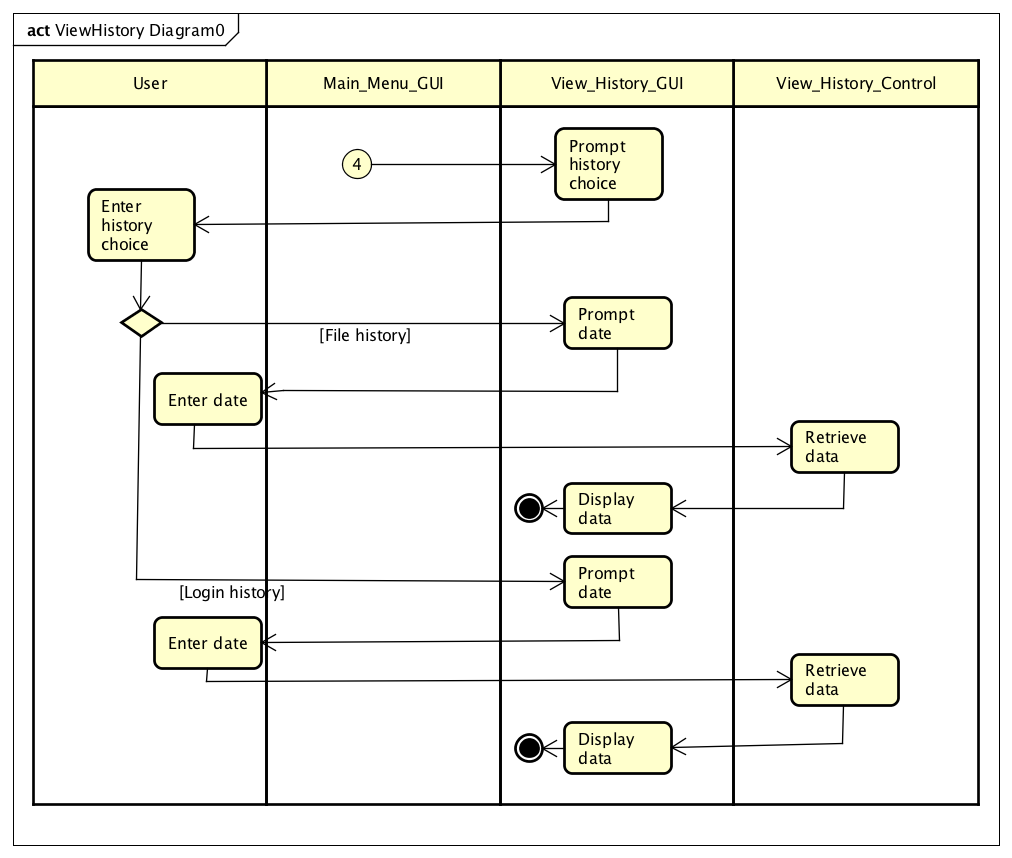


## View History

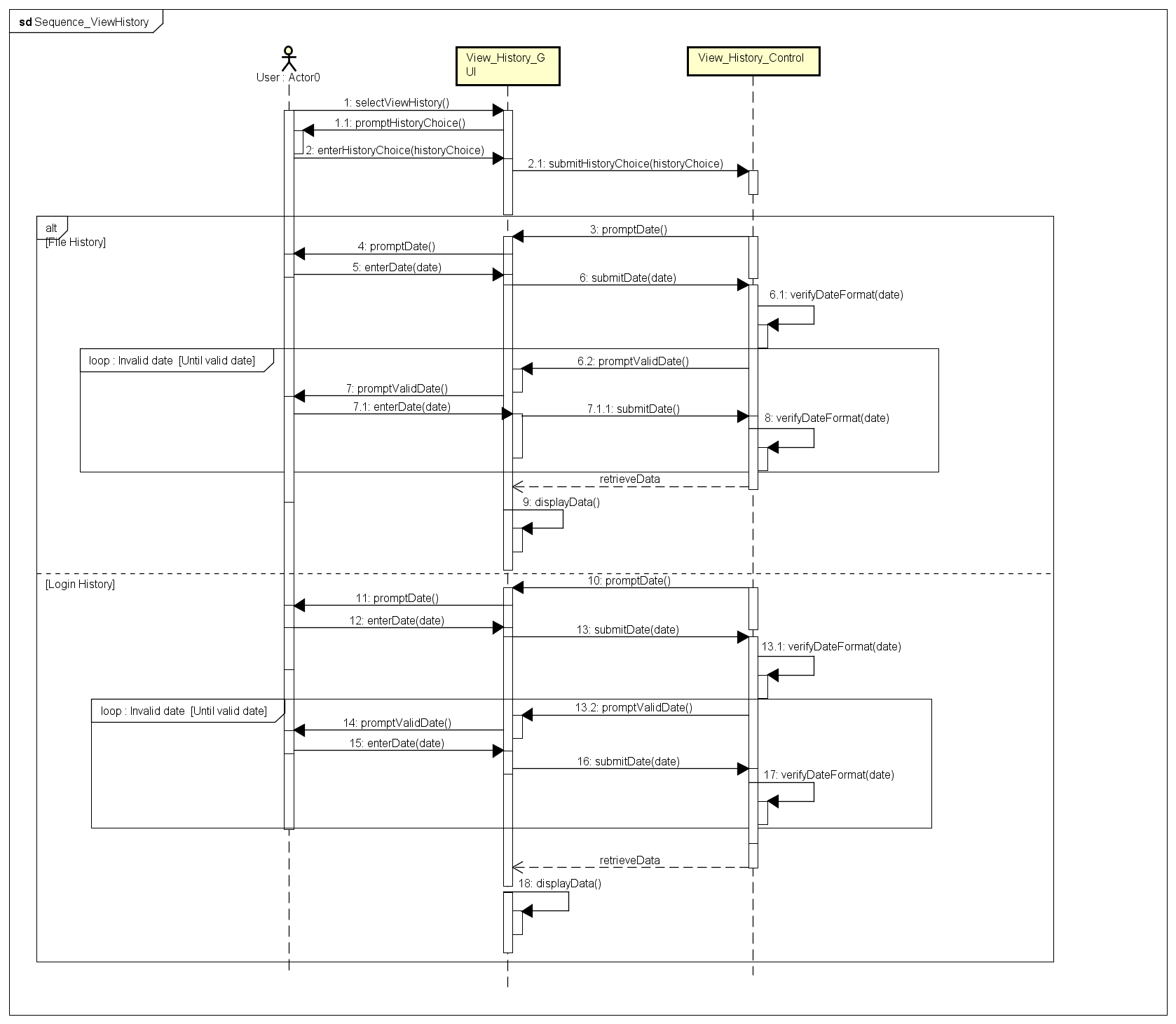
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 1.8 |
| Name | View History |
| Description | To allow user to view history of log in and file |
| Actor(s) | User |
| Precondition | The user has a registered account and logged into the system |
| Main Scenario | Step 1: User selects View History option in the Main Menu  Step 2: User selects a choice in the view history menu  Step 3: System will display a prompt to User to choose from viewfilehistory or viewloginhistory  Step 4(viewfilehistory): User enters a date from where he locked a file **via U/C 1.8.2**  Step 4(viewloginhistory): User enters a date from where he logged in **via U/C 1.8.1**  Step 5: System will retrieve the data and display the data to the User from the date User input. |

### Activity Diagram



### Sequence Diagram

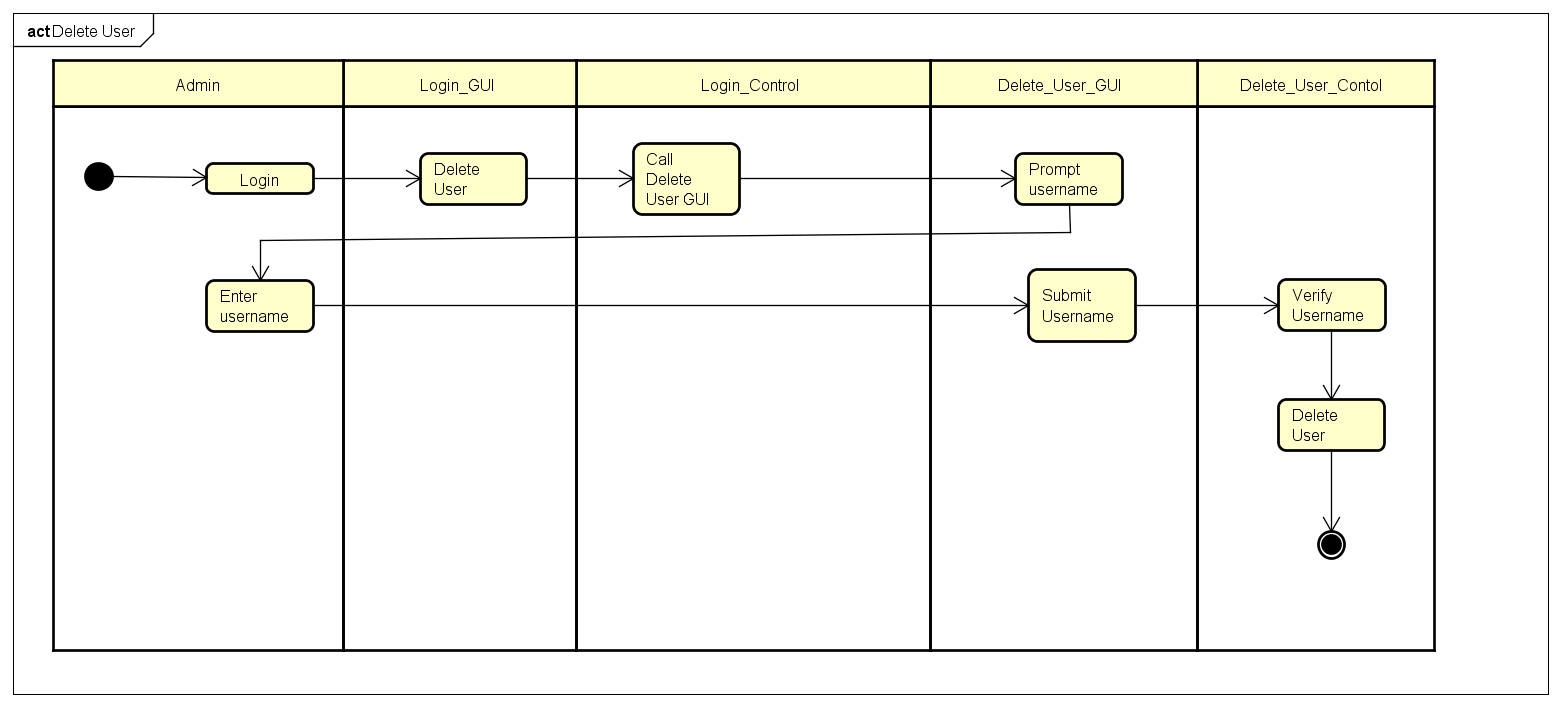


## 2.1 Delete User

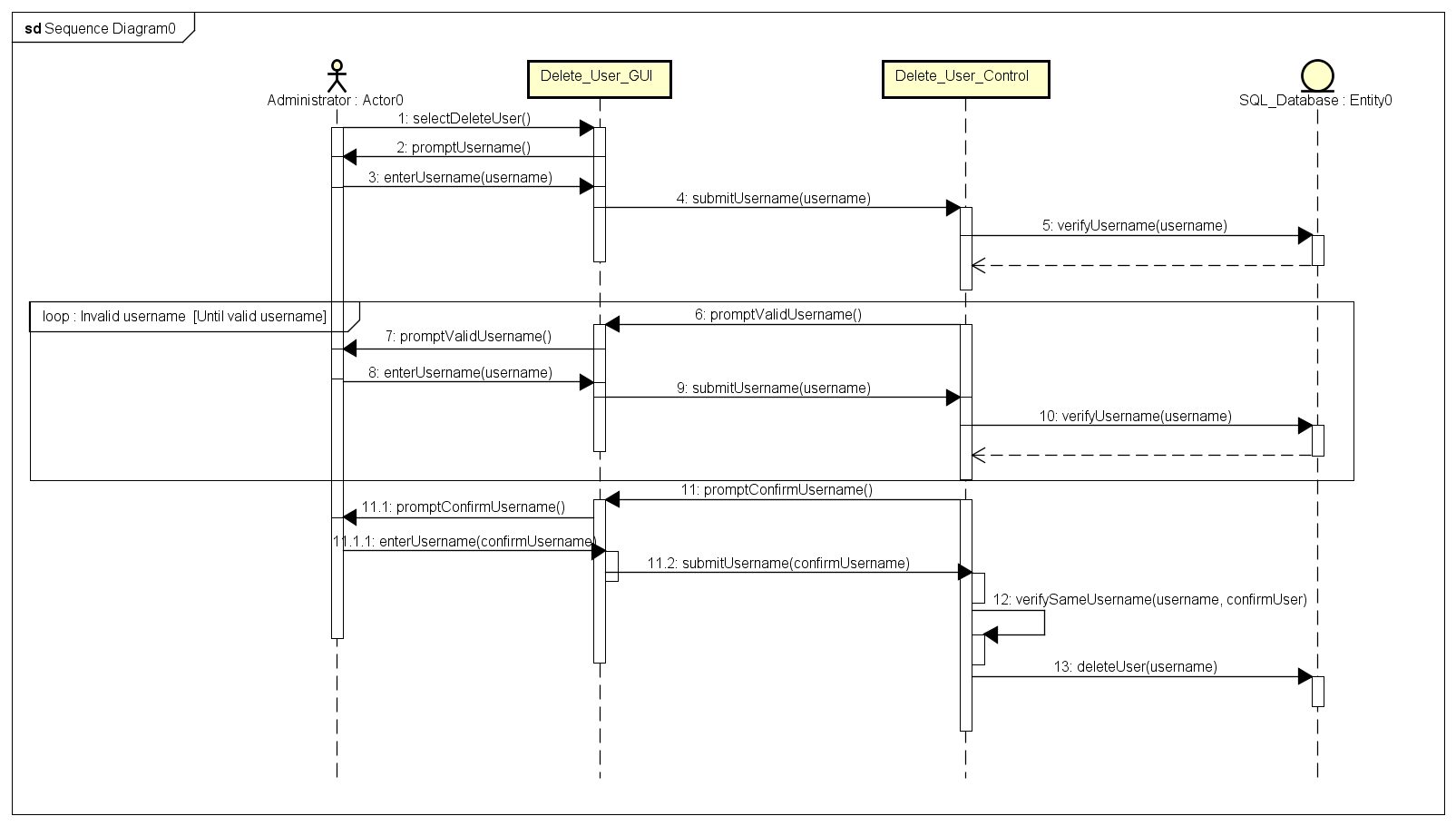
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 2.1 |
| Name | Delete User |
| Description | To allow the administrator to delete users from database |
| Actor(s) | Administrator |
| Precondition | The administrator wishes to delete off unused accounts |
| Main Scenario | Step 1: Administrator selects delete user on Main Menu  Step 2: System prompts to enter username of the account that is to be deleted  Step 3: Administrator fills up form and system validate  Step 3(alternate): If username is invalid, an error message will be displayed, and Administrator is sent back to form for re-entry of right credentials  Step 4: System will prompt to confirm username  Step 5: Administrator re-enter username of the account to be deleted  Step 6: System validates  Step 7: System deletes username from database |

### Activity Diagram



### Sequence Diagram

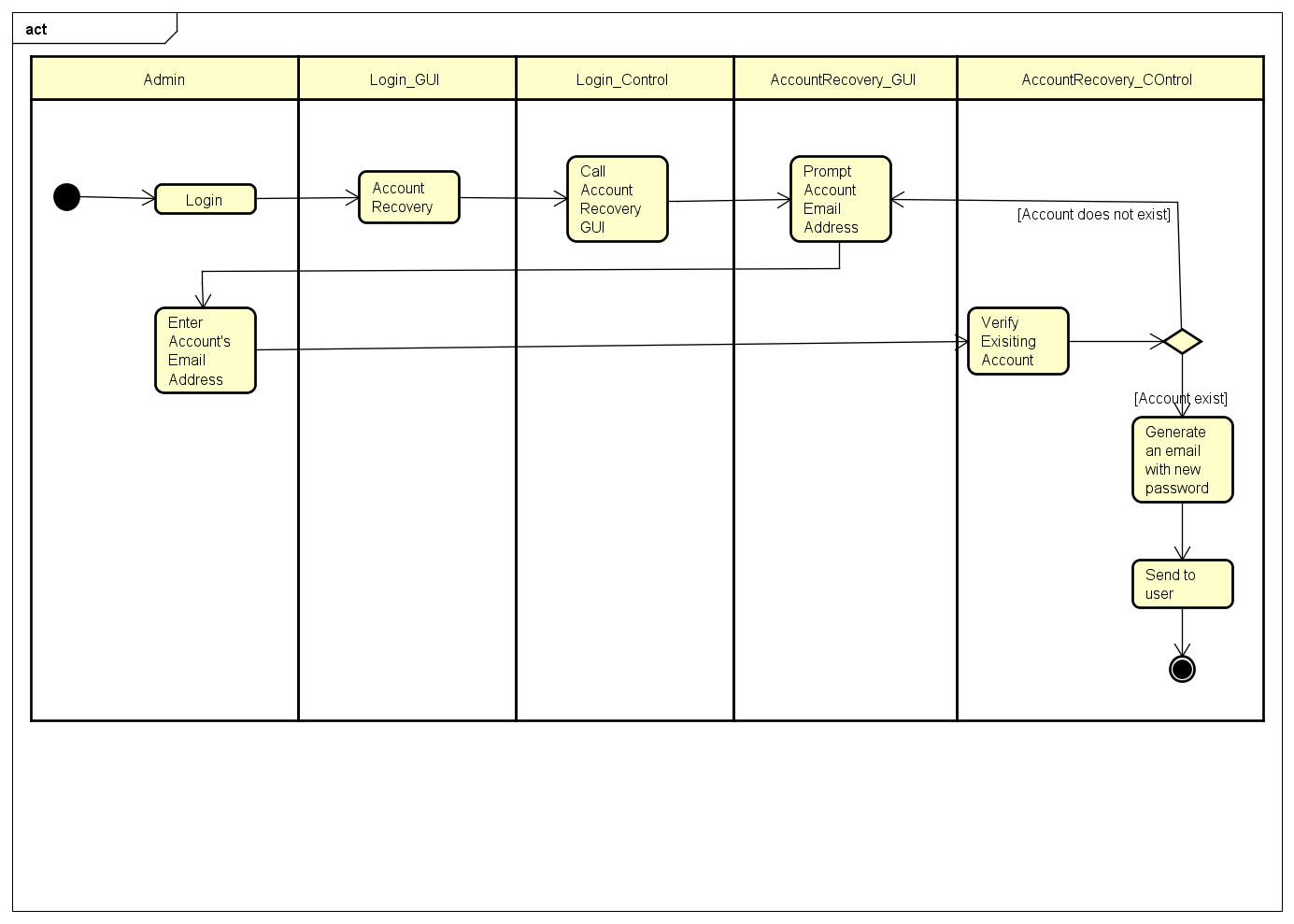


## 2.2 Account Recovery

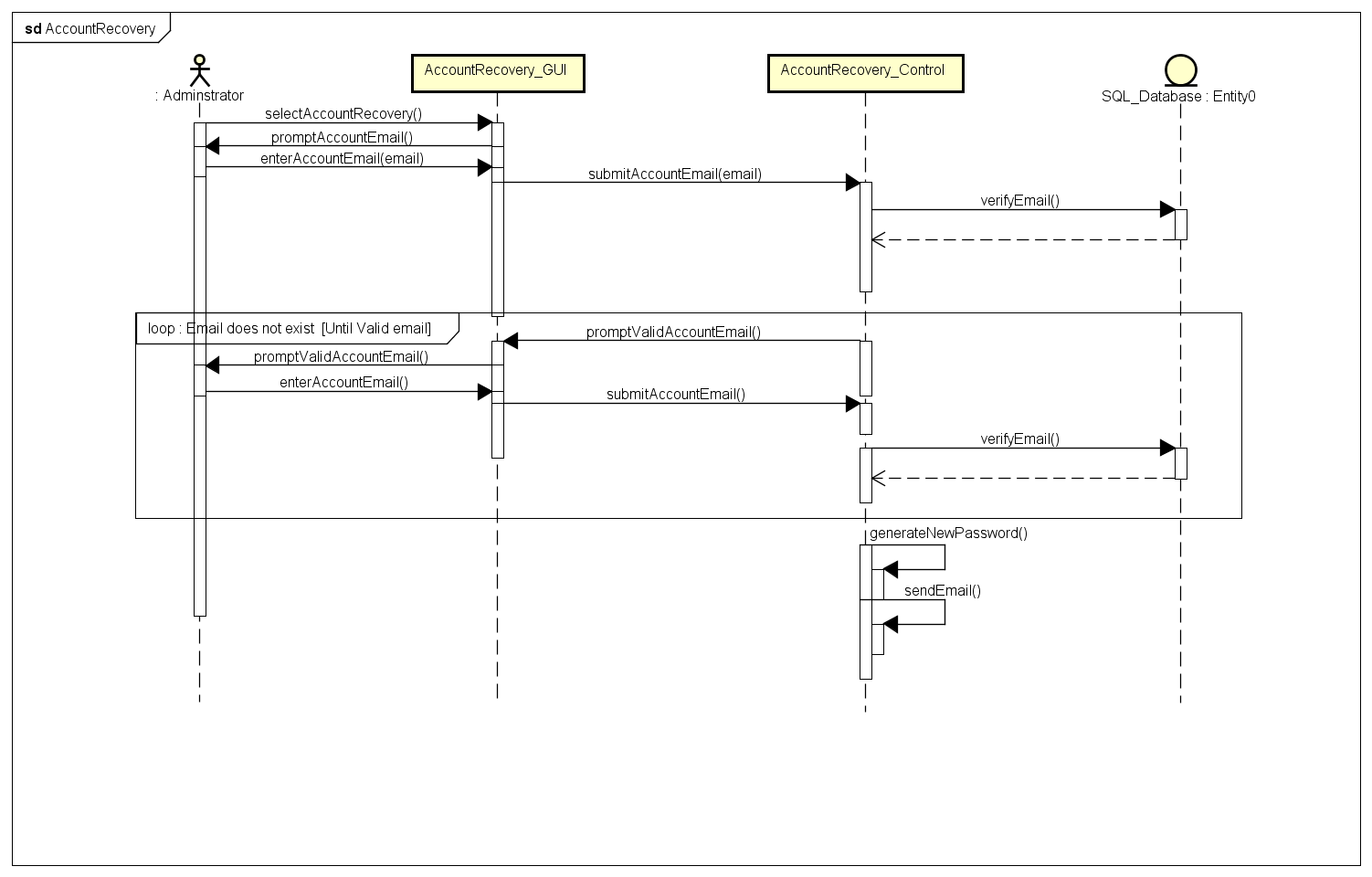
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 2.2 |
| Name | Account Recovery |
| Description | To allow Administrator to recover their lost or forgotten account |
| Actor(s) | User |
| Precondition | The Administrator has registered account and forgotten password |
| Main Scenario | Step 1: Administrator selects account recovery on Log in screen  Step 2: System prints out a form to prompt for Administrator account email  Step 3: Administrator fills up the email and system do validation  Step 3(alternate): If email is invalid, an error message will be displayed, and Administrator is sent back to form for re-entry of right credentials  Step 4: System will generate new password  Step 5: Email with new generated password will be sent to Administrator email address. |

### Activity Diagram



### Sequence Diagram

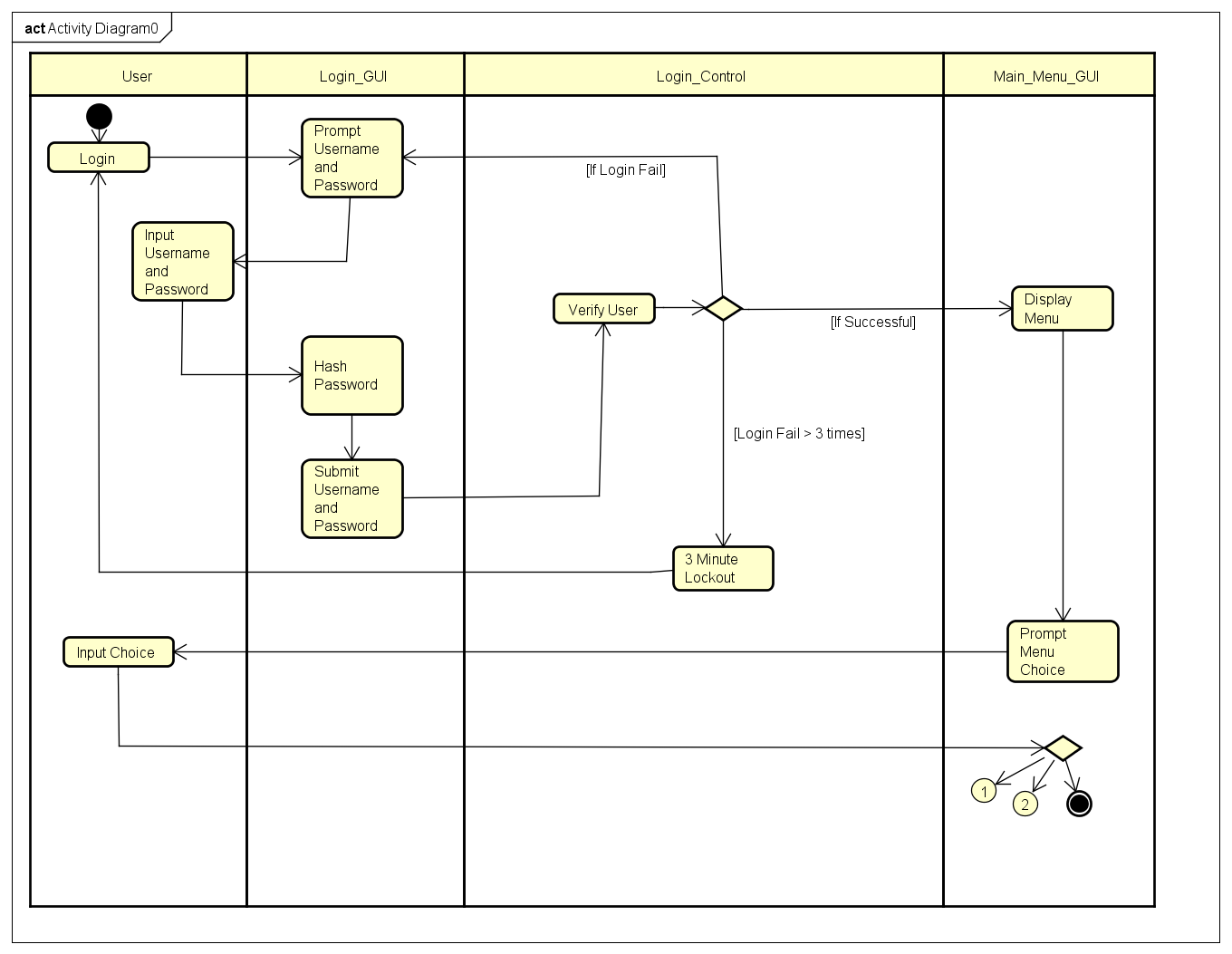


## 3.1 Login

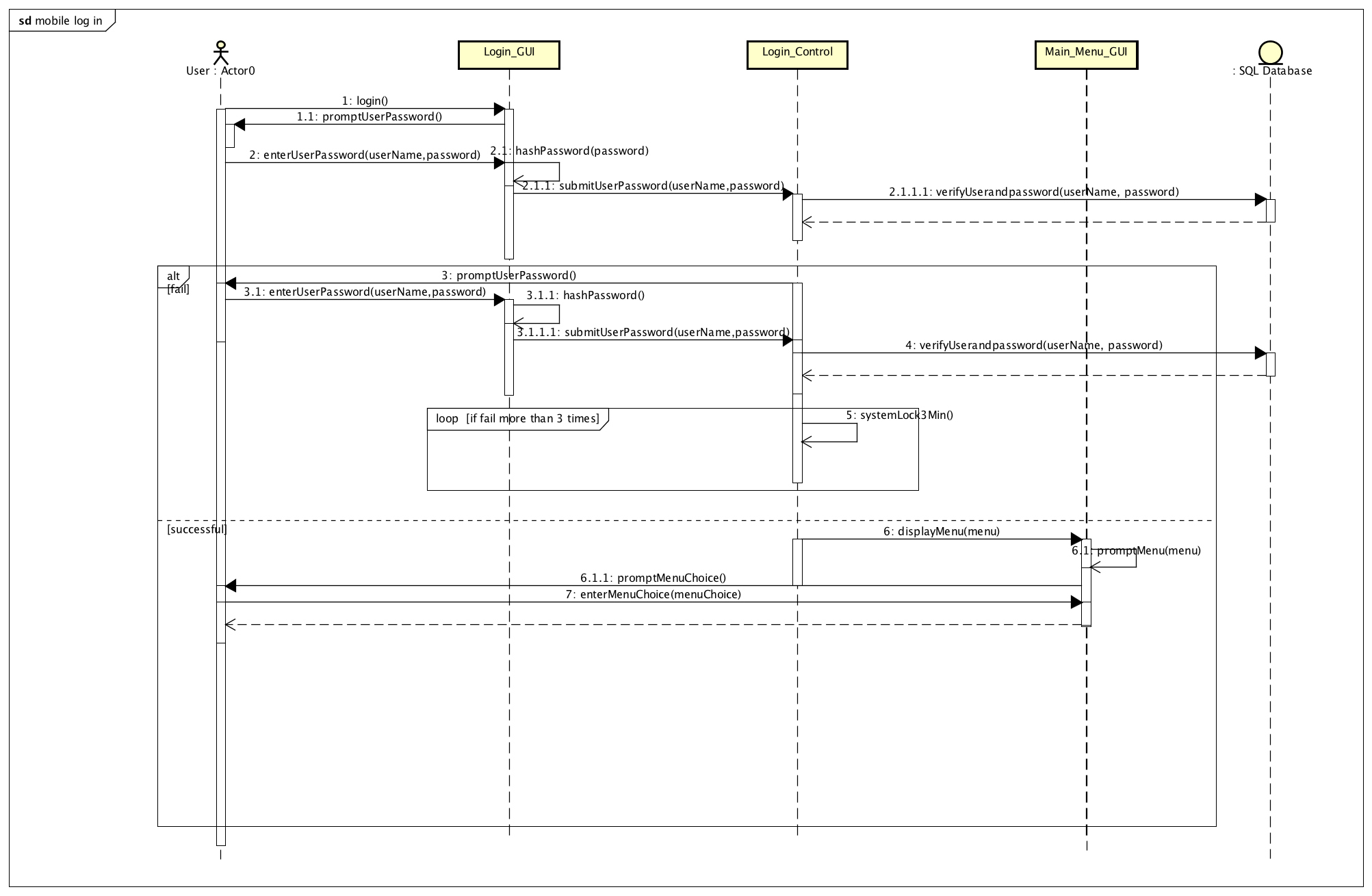
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 3.1 |
| Name | Login |
| Description | To allow User to log in to the system |
| Actor(s) | User |
| Precondition | User wants to use the system |
| Main Scenario | Step 1: User enters their log in credentials in the log in screen  Step 2: System hash the User password and validates with SQL Database  Step 2(alternate): If log in credentials is wrong, an error message will display, and User will be prompt to enter credentials again  Step 2(alternate): If log in credentials is wrong for 3 times, system will be lock from User for 3 minutes  Step 3: System will display Main Menu  Step 4: User will enter choice from Main Menu |

### Activity Diagram



### Sequence Diagram

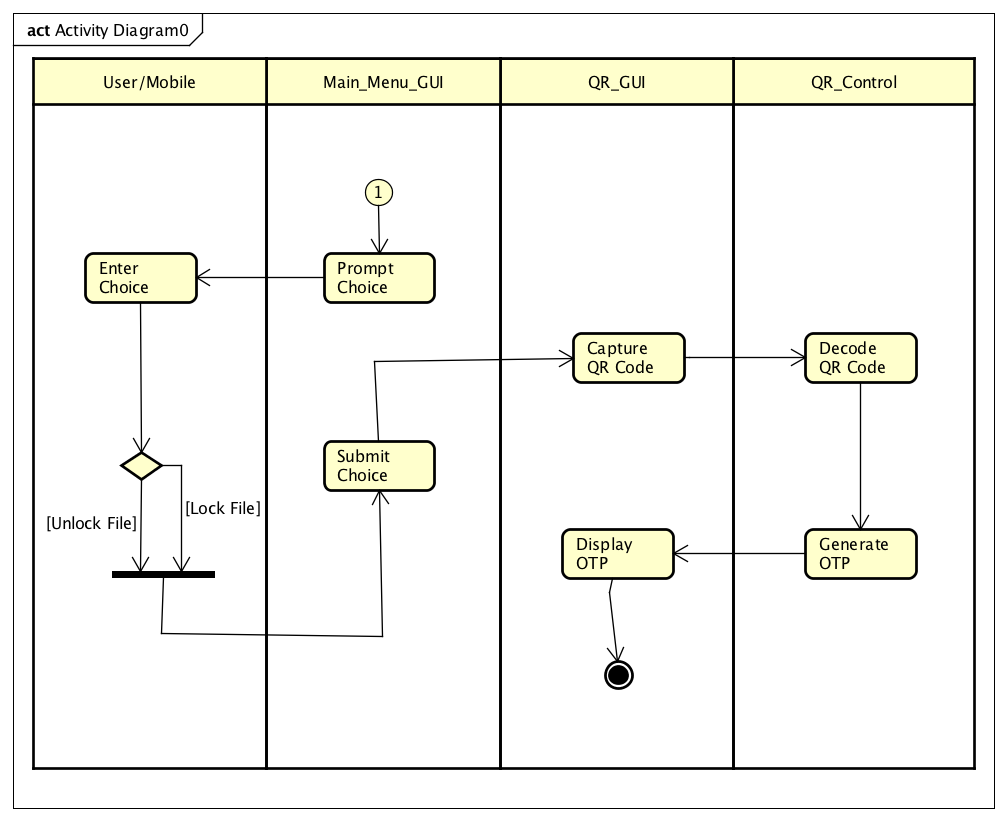


## 3.2 Lock/ Unlock

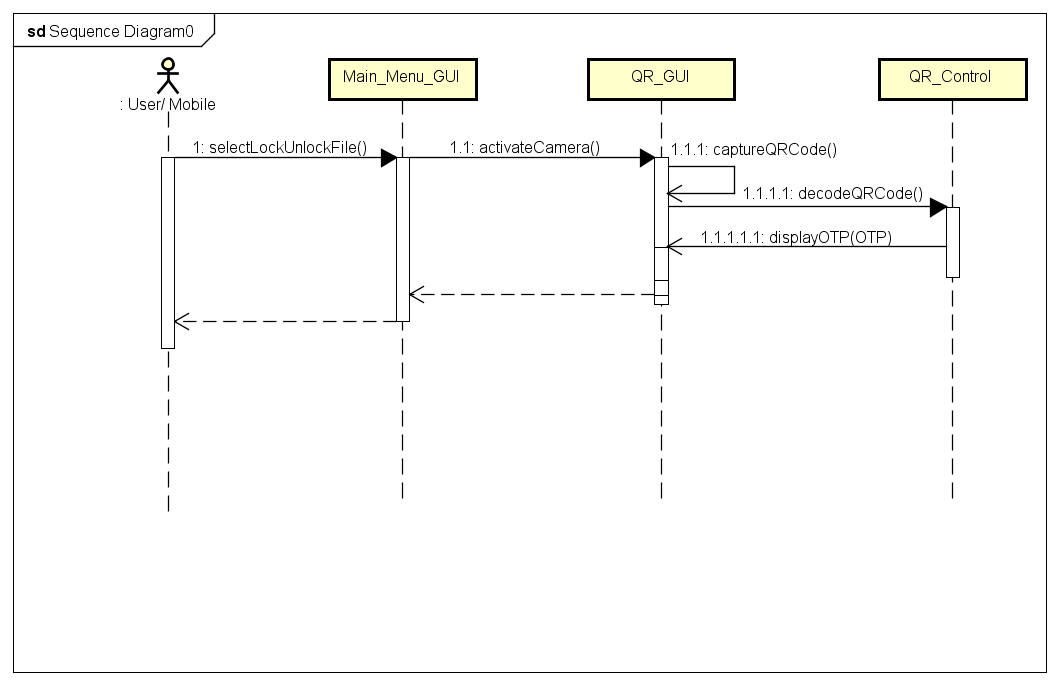
### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 3.2 |
| Name | Lock/Unlock |
| Description | To allow user to lock and unlock their folders or files using system cryptography feature |
| Actor(s) | User |
| Precondition | User has a folder or file that they want to lock or unlock |
| Main Scenario | Step 1: User selects LockUnlock file option in the Main Menu.  Step 2: System activates camera of the mobile phone  Step 3: User captures the QRCODE displayed on the webpage of the app  Step 4: System will decode the QRCODE  Step 5: System displays the hidden OTP in the QRCODE for the user to enter into the system to lock/unlock their files` |

### Activity Diagram



### Sequence Diagram



## 3.3 Change Password

### Use Case

|  |  |
| --- | --- |
| Use Case Textual Description | |
| UC-ID | 3.3 |
| Name | Change Password |
| Description | To allow User to change their password |
| Actor(s) | User |
| Precondition | User wants to change existing password to a new one |
| Main Scenario | Step 1: User enters username and password at Log in screen  Step 2: User selects change password at the main menu  Step 3: System displays a form for user to fill up the new password they desire  Step 4: User enters the new desired password and system do validation  Step 5: System will update the SQL Database with the new password |

### Activity Diagram

### ../Diagrams/Activity%20Diagram/Mobile_App(User)/PNG%20Image/Activity_Mobile_ChangePassword.png

### Sequence Diagram

