**Software Requirements**

**Specification**

**For**

**Memora**

**Version 3.0**

**Prepared by Sahil Kumar 2112232**

**Pershant Parkash 2112229**

**Shaheed Zulfikar Ali Bhutto Institute of Science and Technology**

**Jan 26, 2024**

# TABLE OF CONTENT

[TABLE OF CONTENT 2](#_Toc188878033)

[TABLE OF FIGURES 3](#_Toc188878034)

[TABLE OF TABLES 3](#_Toc188878035)

[Revision History 4](#_Toc188878036)

[1. Introduction 1](#_Toc188878037)

[1.1 Purpose 1](#_Toc188878038)

[1.2 Document Conventions 1](#_Toc188878039)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc188878040)

[1.4 Product Scope 1](#_Toc188878041)

[1.5 References 1](#_Toc188878042)

[2. Overall Description 2](#_Toc188878043)

[2.1 Product Perspective 2](#_Toc188878044)

[2.2 Product Functions 2](#_Toc188878045)

[2.2.1 Login, Logout, and User Registration: 2](#_Toc188878046)

[2.2.2 Profile Management: 2](#_Toc188878047)

[2.2.3 Friend Requests and Community Groups: 2](#_Toc188878048)

[2.2.4 Time Capsule Creation: 2](#_Toc188878049)

[2.2.5 Shared Capsules: 3](#_Toc188878050)

[2.2.6 Memory Capsules on Map: 3](#_Toc188878051)

[2.2.7 Memory Calendar 3](#_Toc188878052)

[2.2.8 Nested Capsules: 3](#_Toc188878053)

[2.2.9 Unlock Milestone Capsules: 3](#_Toc188878054)

[2.2.10 Emotional Connection: 3](#_Toc188878055)

[2.3 User Classes and Characteristics 3](#_Toc188878056)

[2.4 Operating Environment 3](#_Toc188878057)

[2.5 Design and Implementation Constraints 3](#_Toc188878058)

[2.6 User Documentation 4](#_Toc188878059)

[2.6 Assumptions and Dependencies 4](#_Toc188878060)

[3. External Interface Requirements 4](#_Toc188878061)

[3.1 User Interfaces 4](#_Toc188878062)

[3.2 Hardware Interfaces 11](#_Toc188878063)

[3.3 Software Interfaces 11](#_Toc188878064)

[3.4 Communications Interfaces 11](#_Toc188878065)

[4. System Features 11](#_Toc188878066)

[4.1 Login and User Registration 11](#_Toc188878067)

[4.1.1 Description and Priority 11](#_Toc188878068)

[4.1.2 Stimulus/Response Sequences 11](#_Toc188878069)

[4.1.3 Functional Requirements 12](#_Toc188878070)

[4.2 Time Capsule Creation 13](#_Toc188878071)

[4.2.2 Description and Priority 13](#_Toc188878072)

[4.2.3 Stimulus/Response Sequence 13](#_Toc188878073)

[4.2.4 Functional Requirements 14](#_Toc188878074)

[4.3 Profile and Friends Management 15](#_Toc188878075)

[4.3.2 Description and Priority 15](#_Toc188878076)

[4.3.3 Stimulus/Response Sequences 15](#_Toc188878077)

[4.3.4 Functional Requirements 16](#_Toc188878078)

[4.4 Memory Capsules on Map 16](#_Toc188878079)

[4.4.2 Description and Priority 16](#_Toc188878080)

[4.4.3 Stimulus/Response Sequences 16](#_Toc188878081)

[4.4.4 Functional Requirements 17](#_Toc188878082)

[4.5 Emotional Connection in Capsule 17](#_Toc188878083)

[4.5.2 Description and Priority 17](#_Toc188878084)

[4.5.3 Stimulus/Response Sequences 18](#_Toc188878085)

[4.5.4 Functional Requirements 19](#_Toc188878086)

[4.6 Nested Capsules 20](#_Toc188878087)

[4.6.2 Description and Priority 20](#_Toc188878088)

[4.6.3 Stimulus/Response Sequences 20](#_Toc188878089)

[4.6.4 Functional Requirements 21](#_Toc188878090)

[4.7 Media Upload/Creation Options in Capsule 21](#_Toc188878091)

[4.7.2 Description and Priority 21](#_Toc188878092)

[4.7.3 Stimulus/Response Sequences 21](#_Toc188878093)

[4.7.4 Functional Requirements 23](#_Toc188878094)

[4.8 Shared Capsules 23](#_Toc188878095)

[4.8.2 Description and Priority 23](#_Toc188878096)

[4.8.3 Stimulus/Response Sequences 23](#_Toc188878097)

[4.8.4 Functional Requirements 24](#_Toc188878098)

[5. Nonfunctional Requirements 25](#_Toc188878099)

[5.1 Performance Requirements 25](#_Toc188878100)

[5.2 Safety Requirements 25](#_Toc188878101)

[5.3 Security Requirements 25](#_Toc188878102)

[5.4 Software Quality Attributes 25](#_Toc188878103)

[5.5 Business Rules 25](#_Toc188878104)

[5.5.1 User Roles and Permissions: 25](#_Toc188878105)

[5.5.2 Capsule Creation: 25](#_Toc188878106)

[5.5.3 Milestone and Goal-Based Capsules: 26](#_Toc188878107)

[5.5.4 Notifications and Reminders: 26](#_Toc188878108)

[5.5.5 Profile and Community Management: 26](#_Toc188878109)

[5.5.6 Data Privacy and Security: 26](#_Toc188878110)

[6. Other Requirements 26](#_Toc188878111)

[Appendix A: Glossary 26](#_Toc188878112)

[Appendix B: Analysis Models 27](#_Toc188878113)

[Appendix C: To Be Determined List 28](#_Toc188878114)

# TABLE OF FIGURES

[Figure 1:Context Diagram 2](#_Toc188878115)

[***Figure 2:Login***  ***Figure 3:SignUp1*** 4](#_Toc188878116)

[Figure 4:SignUp2   
Figure 5 MemoraS 5](#_Toc188878117)

[Figure 6:Calendar   
Figure 7:CalendarDetails 6](#_Toc188878118)

[Figure 8:FriendList   
Figure 9:Start/Capture 7](#_Toc188878119)

[Figure 10:CapsuleCreation   
Figure 11:Capsules 8](#_Toc188878120)

[Figure 12:ShareCapsule   
Figure 13:Logout 9](#_Toc188878121)

[Figure 15 :ProfileEdit 10](file:///C:\Users\Administrator\Documents\Memora_SRS2_Updated...docx#_Toc188878122)

[Figure 14:ProfieUpdate 10](file:///C:\Users\Administrator\Documents\Memora_SRS2_Updated...docx#_Toc188878123)

# TABLE OF TABLES

[Table 1:: Registration Stimulus/Response Sequences 12](#_Toc188878128)

[Table 2:: Registration Functional Requirements 13](#_Toc188878129)

[Table 3:: Time Capsule Creation Stimulus/Response Sequences 14](#_Toc188878130)

[Table 4:: Time Capsule Creation Functional Requirements 15](#_Toc188878131)

[Table 5:: Profile and Friends Management Stimulus/Response Sequences 16](#_Toc188878132)

[Table 6:: Profile and Friends Management Functional Requirements 16](#_Toc188878133)

[Table 7:: Memory Capsules on Map Stimulus/Response Sequences 17](#_Toc188878134)

[Table 8:: Memory Capsules on Map Functional Requirements 17](#_Toc188878135)

[Table 9:: Emotional Connection in Capsule Stimulus/Response Sequences 19](#_Toc188878136)

[Table 10:: Emotional Connection in Capsule Functional Requirements 20](#_Toc188878137)

[Table 11:: Nested Capsules Stimulus/Response Sequences 21](#_Toc188878138)

[Table 12:: Nested Capsules Functional Requirements 21](#_Toc188878139)

[Table 13:: Media Upload Options in Capsule Stimulus/Response Sequences 23](#_Toc188878140)

[Table 14:: Media Upload Options in Capsule Functional Requirements 23](#_Toc188878141)

[Table 15:: Shared Capsules Stimulus/Response Sequences 24](#_Toc188878142)

[Table 16:: Shared Capsules Functional Requirements 24](#_Toc188878143)

# 

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

Memora is a digital time capsule application that allows users to preserve memories and information in secure "Time Capsules," which can be locked and set for future access. This document outlines the software requirements for the initial and future releases of the application. The SRS includes core functionalities such as user authentication, Time Capsule creation, media uploads, sharing capabilities, and interactive features. The focus is on offering a user-friendly platform for preserving and sharing memories over time, providing a meaningful and engaging experience.

## Document Conventions

**Font Style:** Text with headings is written in Times New Roman, Headings Bold.

**Font Size**: 12 pt for body text, 14 pt for Sub-headings, 18 pt for headings.

**Emphasis**: Key terms are italicized, and important features are bolded. **Prioritization**: Requirements are categorized as High, Medium, or Low priority. **Lists and Bullets**: Numbered lists for steps and bullet points for clarity.

## Intended Audience and Reading Suggestions

The Memora SRS is intended for developers, project managers, testers, marketing staff, and end users (friends, family and collaborators). Developers and testers will focus on technical specifications, while project managers review scope and milestones. End users can explore features like Time Capsule creation, profile management and capsule sharing.

## Product Scope

Memora allows users to create digital Time Capsules containing images, videos, and audio, which can be locked until a specified future date. The app enables users to share these capsules with friends and family, adding a social dimension to memory preservation. Memora aims to provide a seamless experience for users who want to create, share, and revisit their memories over time.

## References

N/A

# Overall Description

## Product Perspective

Memora is an independent mobile app focused on preserving and sharing memories securely. Users can create digital Time Capsules and set future unlock dates. Memora operates independently while allowing integration with external services, like map integration.

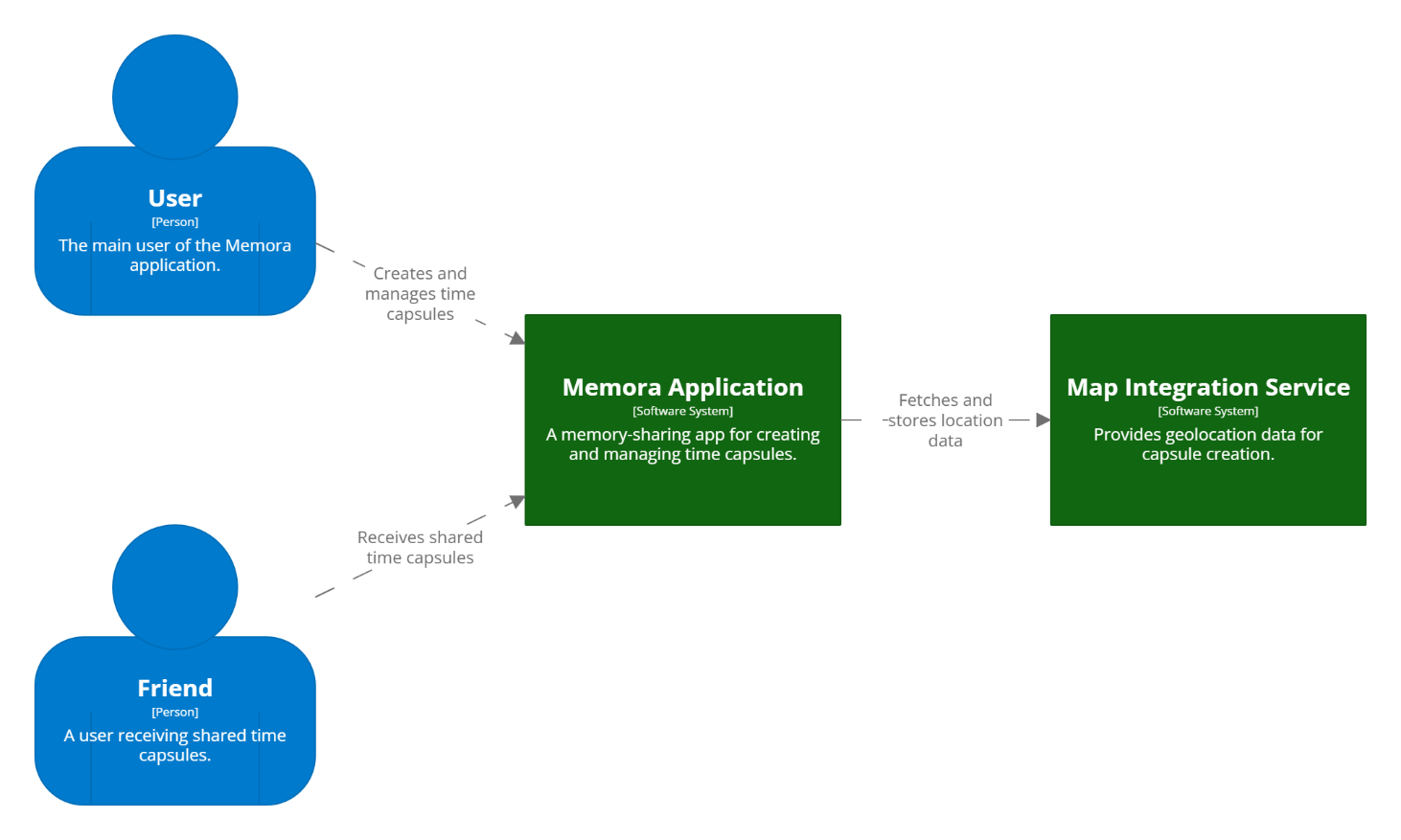


Figure 1:Context Diagram

## Product Functions

Memora includes the following core functions:

### Login, Logout, and User Registration:

* + - * Secure user authentication system allowing new users to register, log in, and access their accounts.

### Profile Management:

* + - * Users can create profiles, add friends or family members, and manage their connections through the app.

### Friend Requests and Community Groups:

* + - * Users can send and accept friend requests, as well as create and join community groups for sharing Time Capsules with multiple members.

### Time Capsule Creation:

* + - * Users can create digital Time Capsules containing images, videos, and audio, locked until a set future date.

### Shared Capsules:

* + - * Time Capsules can be shared with friends, family, or community groups for a collective opening experience, where everyone can access the capsule’s contents together.

### Memory Capsules on Map:

* + - * Users can view the locations where they have created capsules on an interactive map, making it easy to see memories tied to specific places.

### Memory Calendar

* Displays a calendar with highlighted dates showing when each Time Capsule is scheduled to unlock, allowing users to see upcoming memory events at a glance.

### Nested Capsules:

* + - * Allows users to create multiple smaller capsules within a single Time Capsule, enabling all the sub-capsules to open together on a shared unlock date for a layered memory experience.

### Unlock Milestone Capsules:

* + - * Provides the ability for users to set specific goals or milestones, where capsules only unlock once the defined goal is achieved, offering a sense of accomplishment and surprise when accessing the stored content.

### Emotional Connection:

* + - * Users can add text or audio previews to capsules that can be accessed before the official unlock date, creating anticipation and excitement for the memory or information to be revealed.

## User Classes and Characteristics

* **End Users**: Individuals using Memora can preserve and share memories, with friends and families.
* **Collaborators**: People who receive shared capsules, providing a joint memory experience.
* **Project Managers**: Oversee milestones and app features to ensure development meets user needs.
* **Technical Support**: Handles user issues and ensures smooth app performance.

## Operating Environment

* **Devices**: Smartphones with at least 2 GB RAM.
* **Operating Systems**: Android 8.0+ and iOS 12.0+.
* **Software Components**: Developed with React Native, Node.js backend, and Mongoose for database management.

## Design and Implementation Constraints

* Must comply with data privacy regulations.
* Limited by device hardware capabilities, such as camera and GPS.
* Uses secure communication protocols (HTTPS).

## 2.6 User Documentation

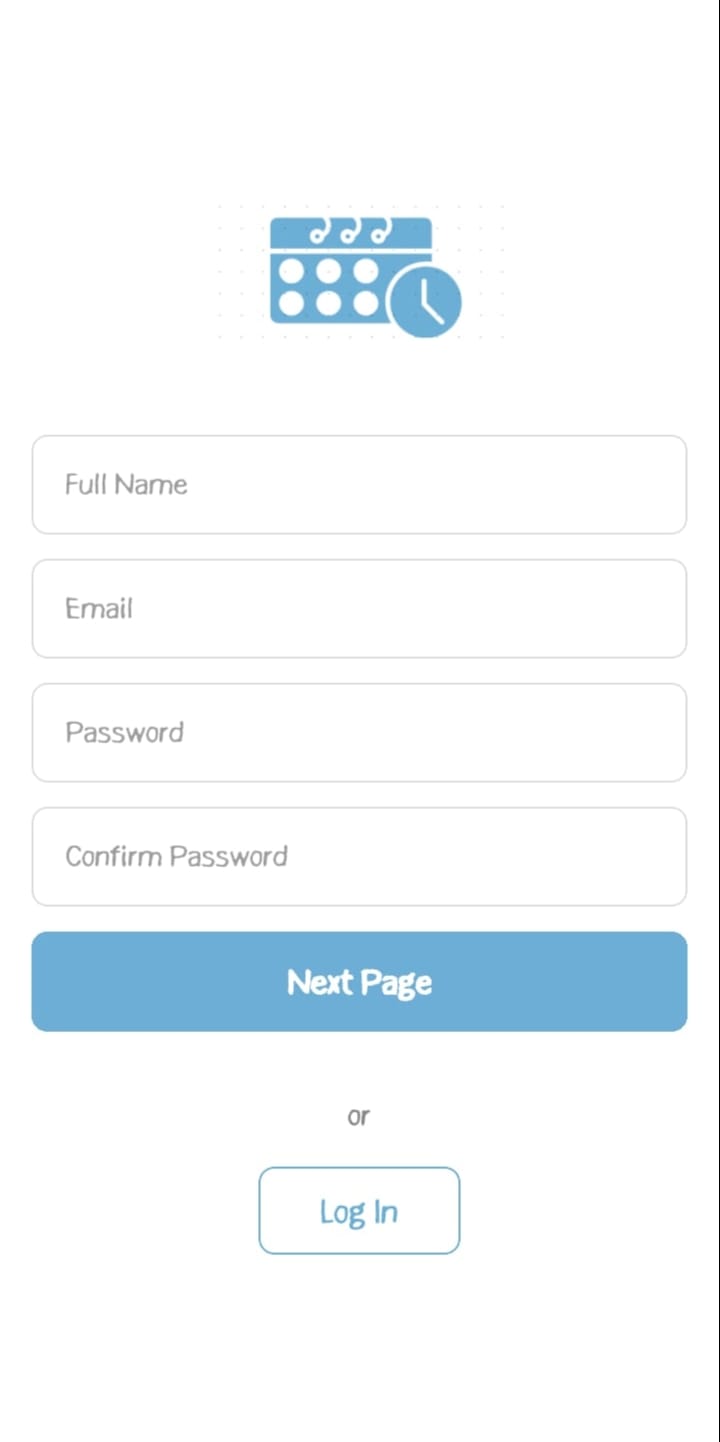
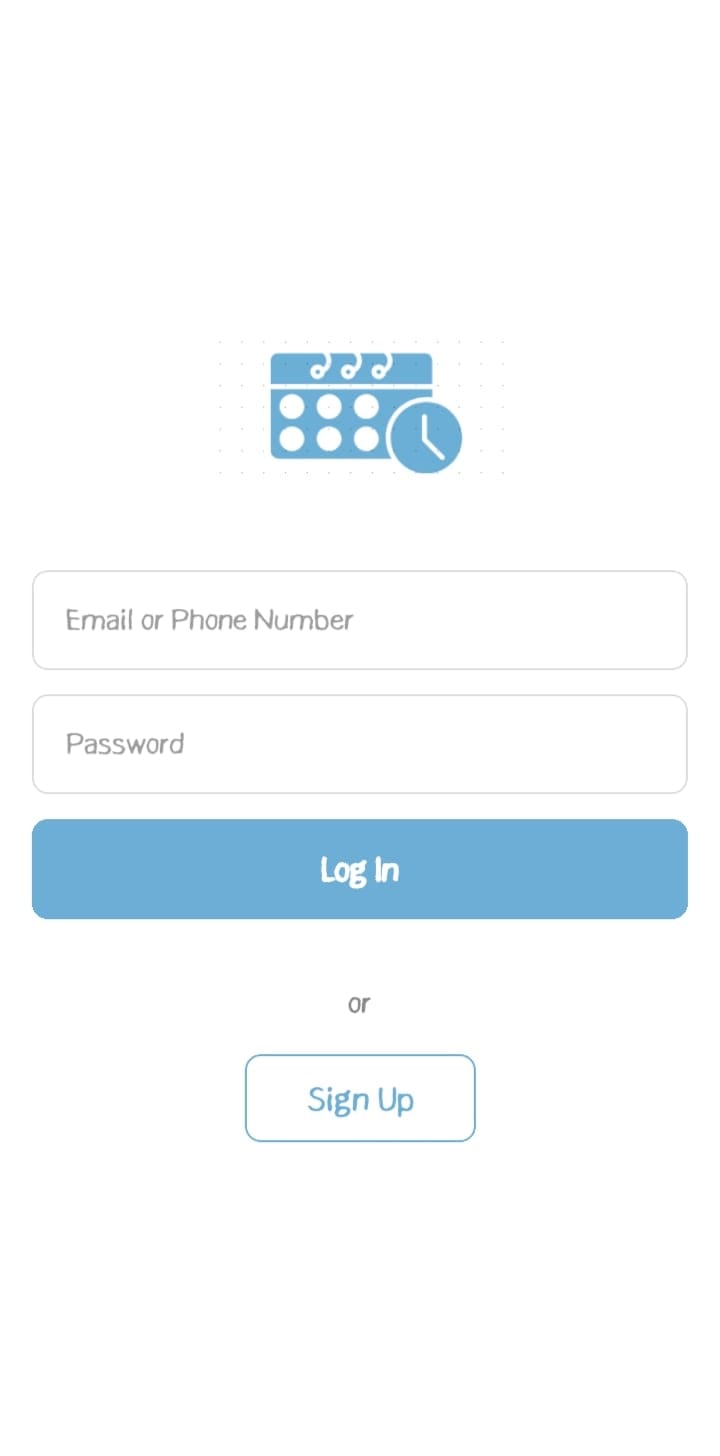
Includes a user manual, in-app help guides, and tutorials for Time Capsule creation, media uploads, and sharing features. Delivered as in-app tutorials and PDF manuals.

## Assumptions and Dependencies

* Depends on stable internet connections for seamless sharing and syncing.
* Relies on third-party APIs for location-based services and notifications.

# External Interface Requirements

## User Interfaces



***Figure 2:Login Figure 3:SignUp1***

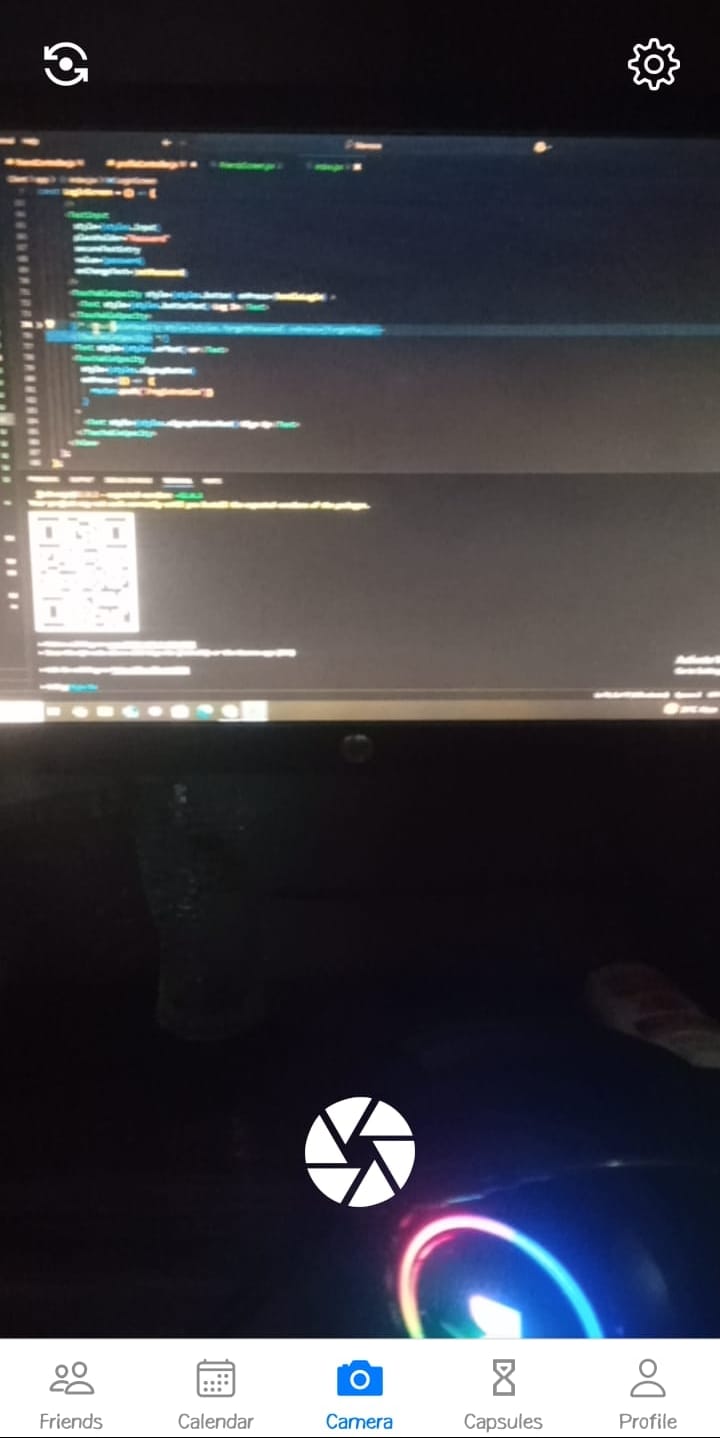
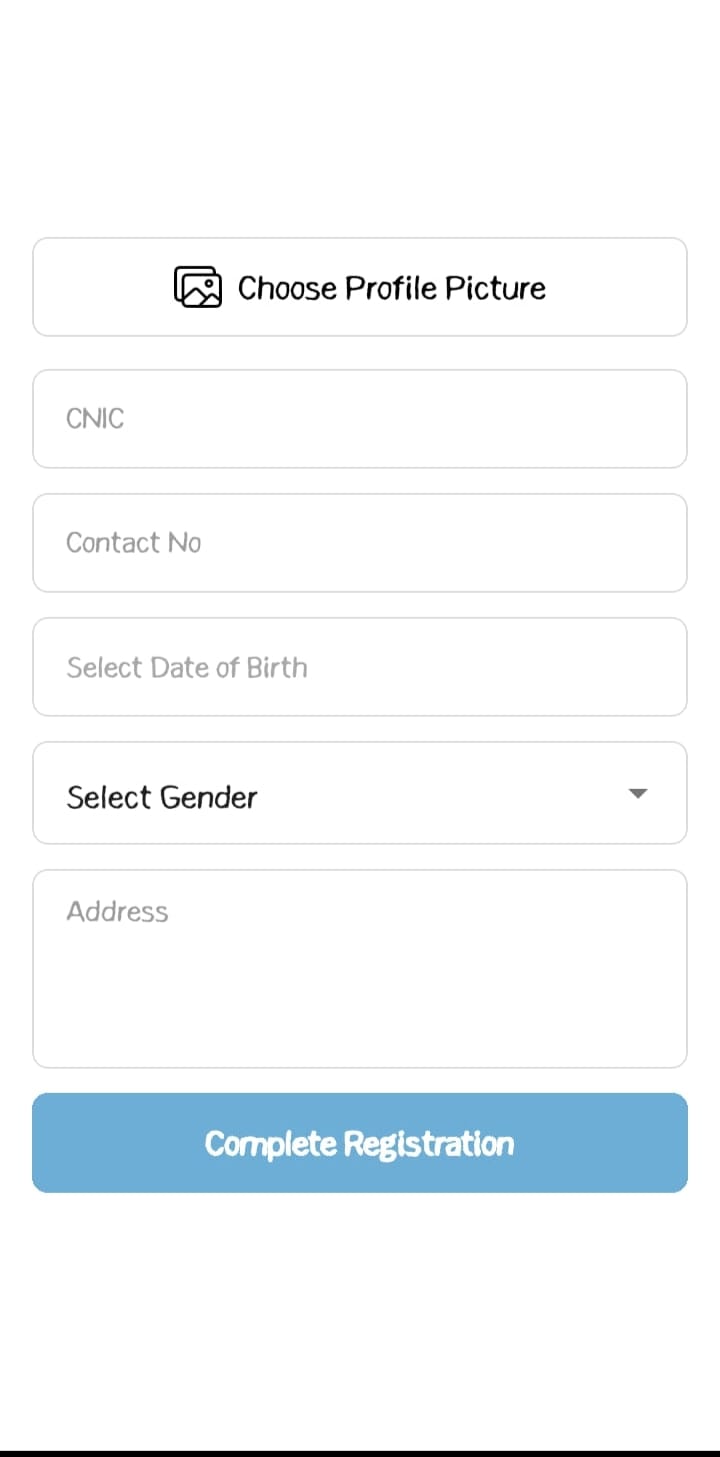


Figure 4:SignUp2 Figure 5 MemoraS

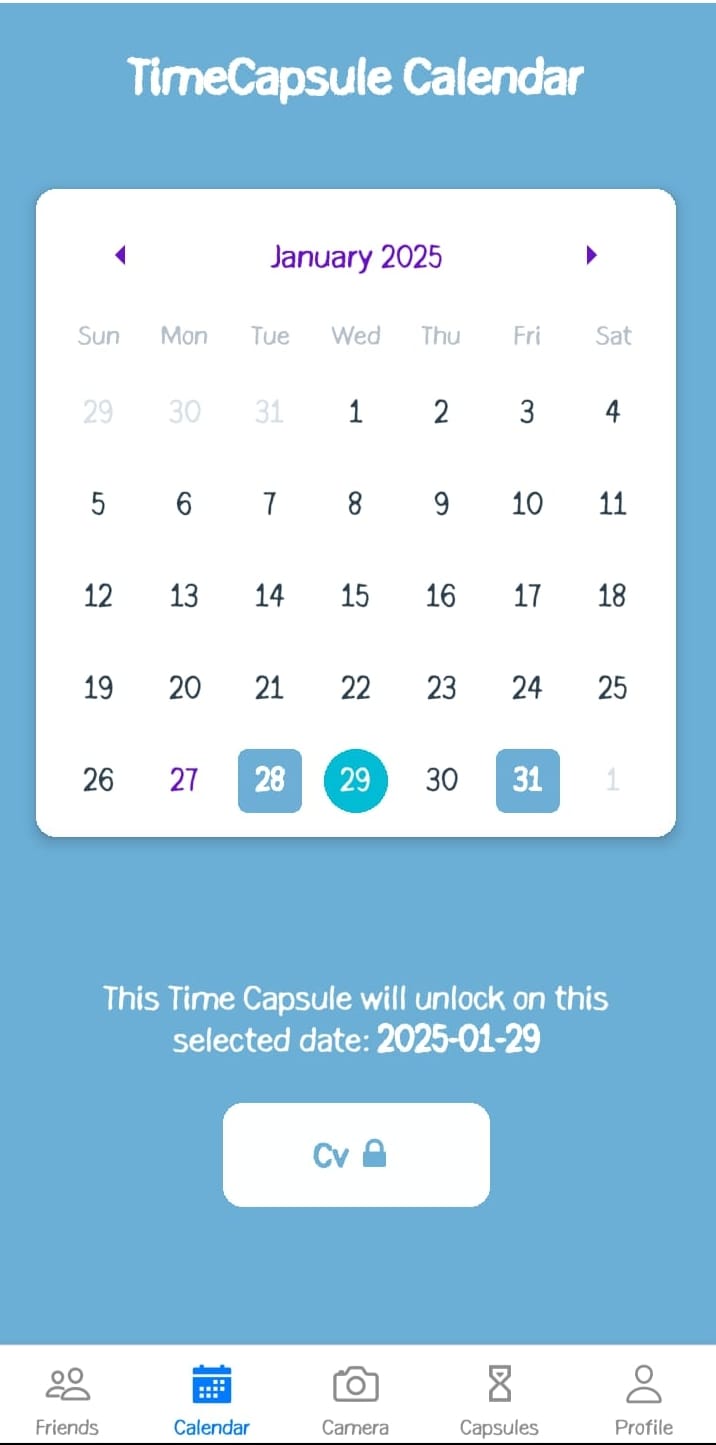
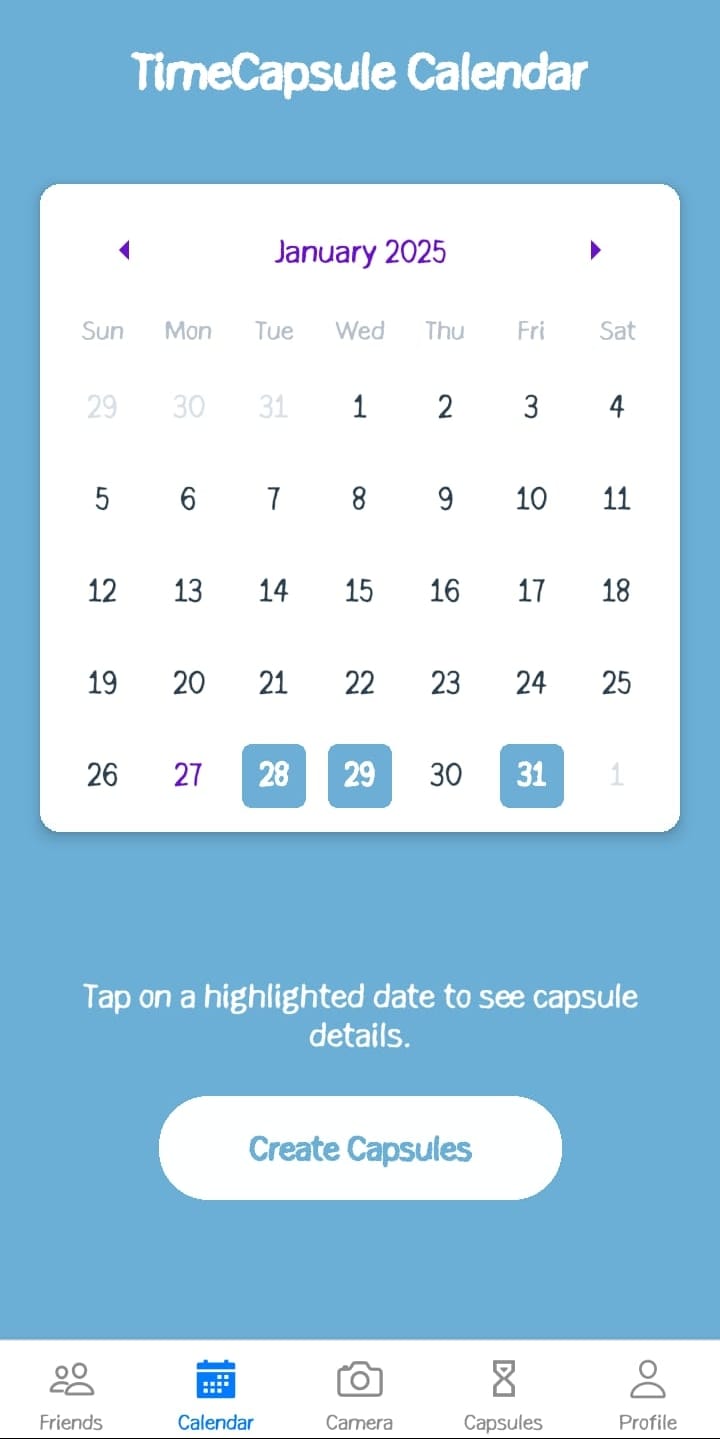


Figure 6:Calendar Figure 7:CalendarDetails

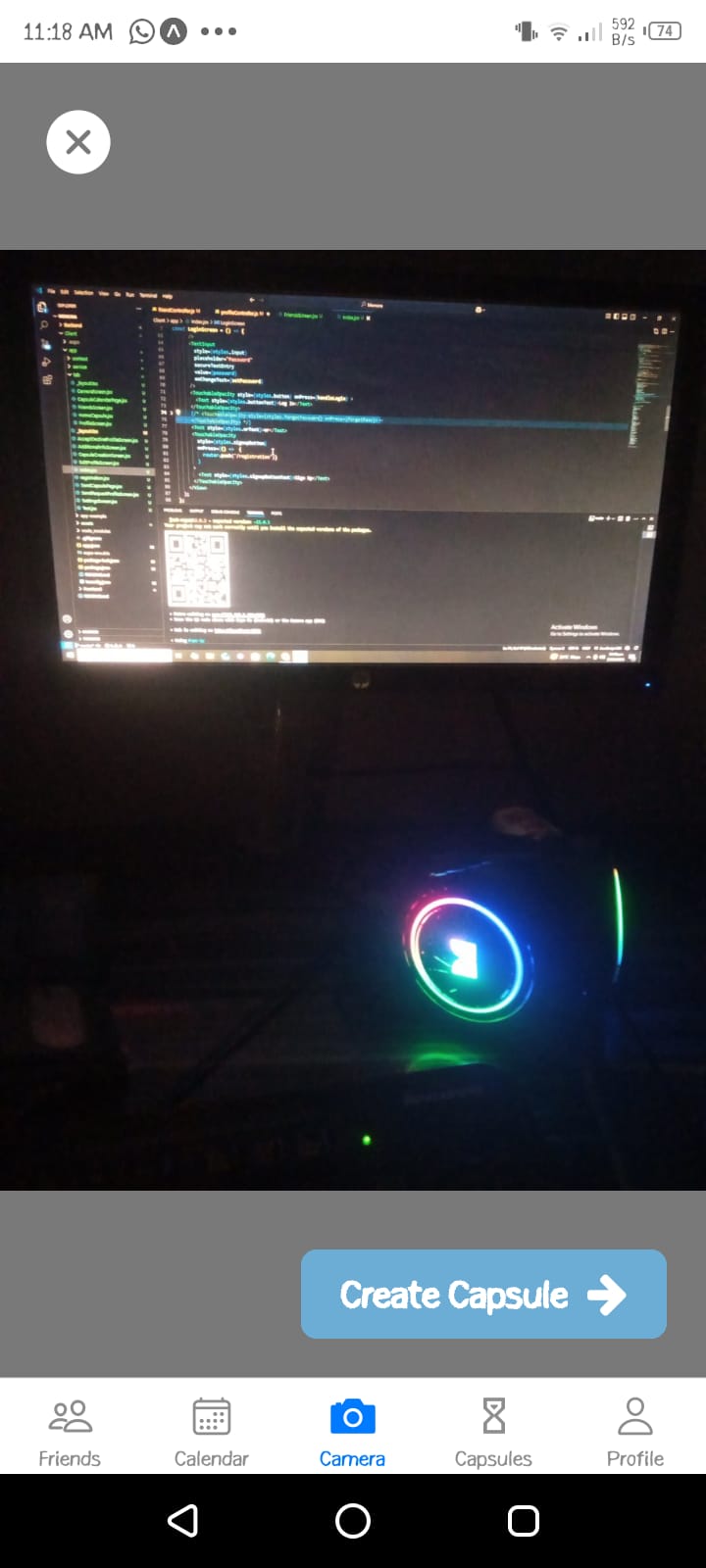
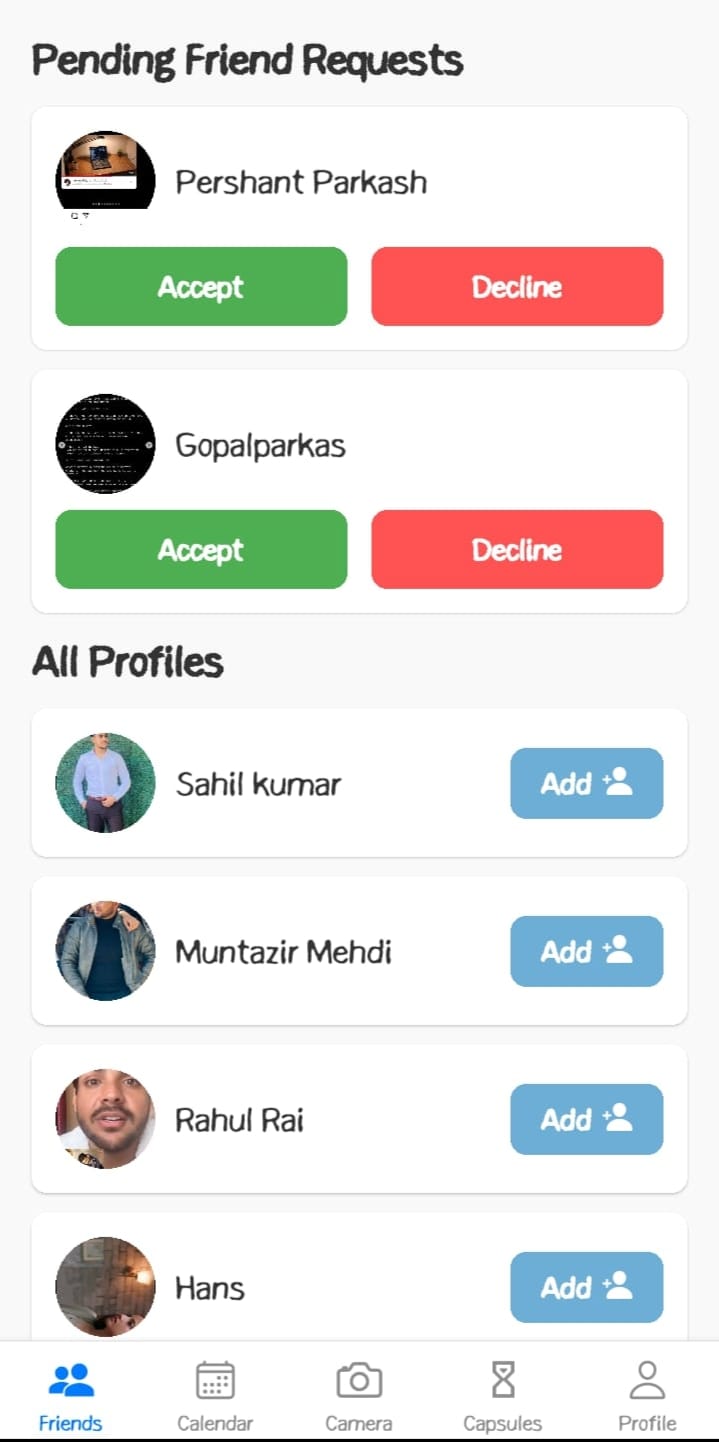


Figure 8 :FriendList Figure 9:Start/Capture

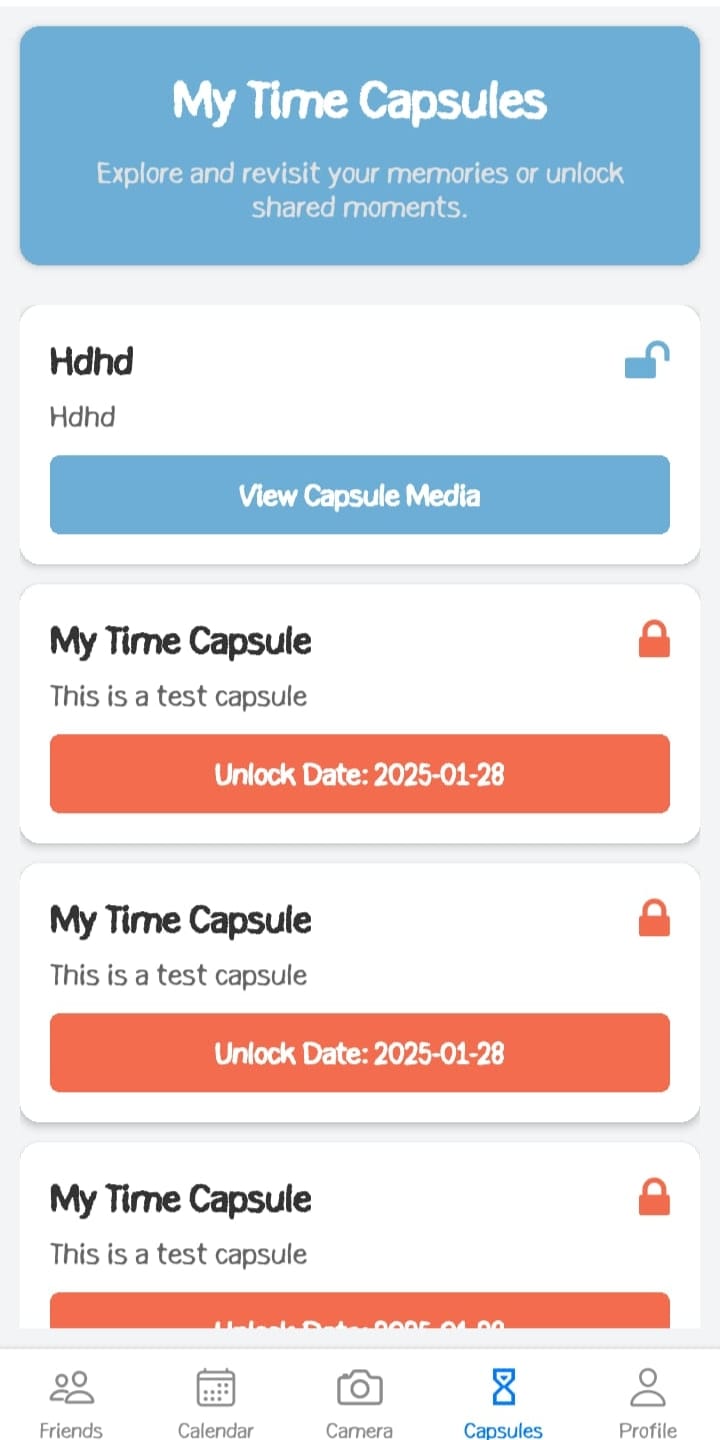
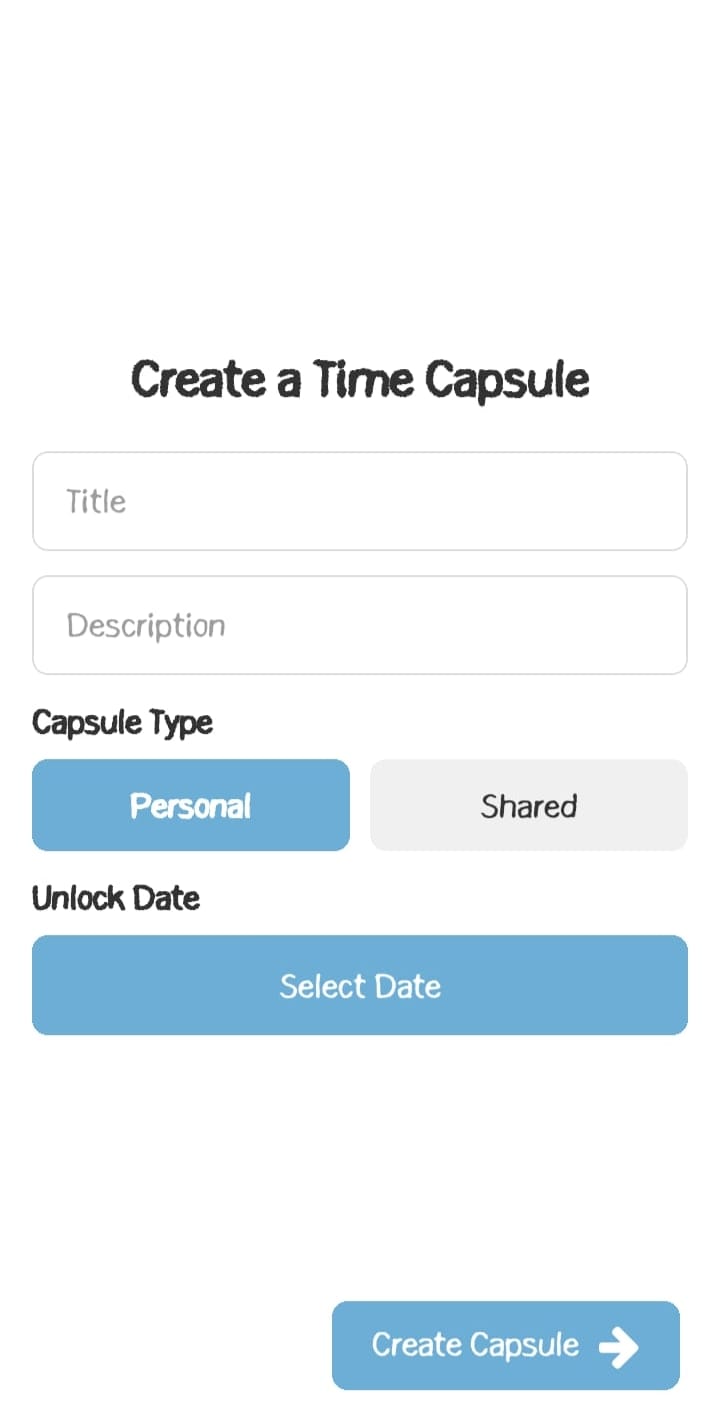
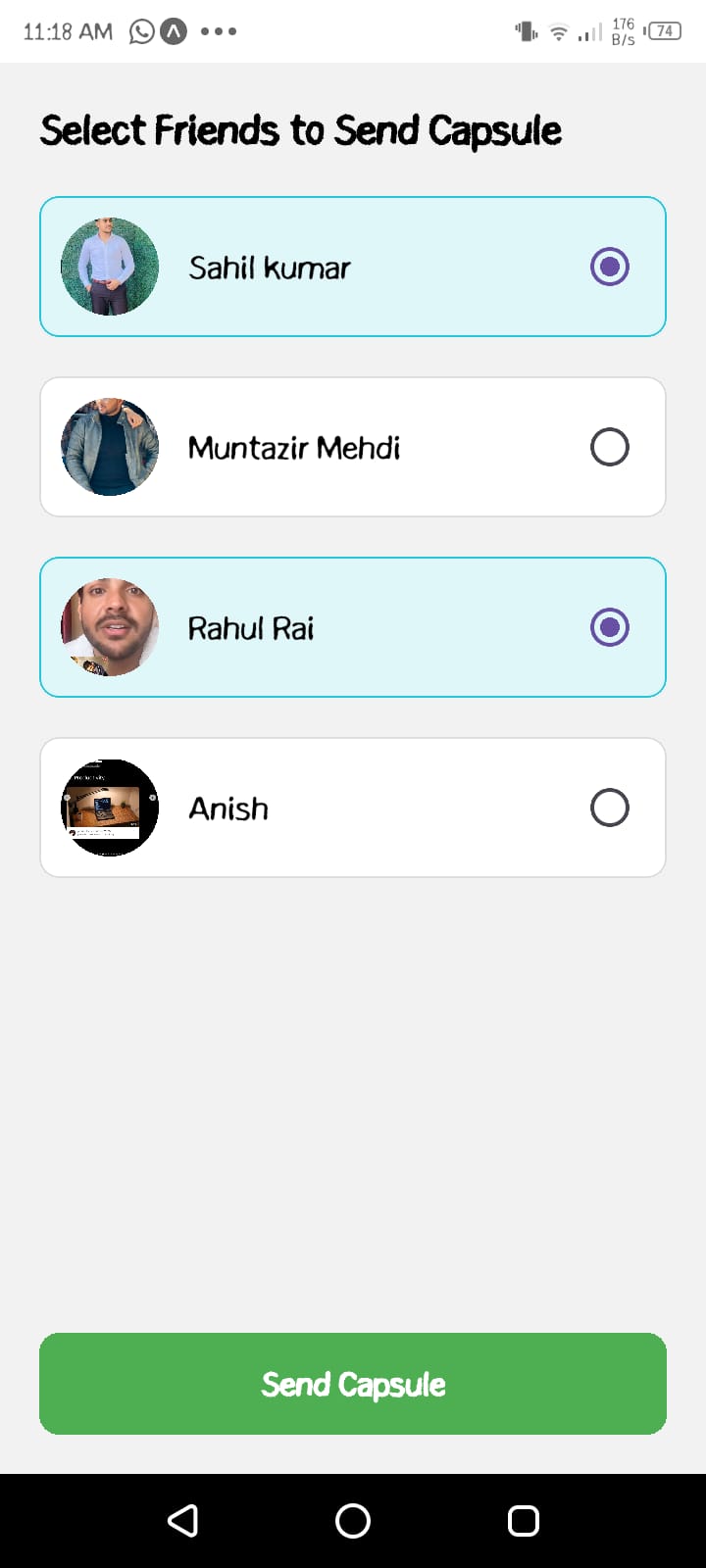


Figure 10:CapsuleCreation Figure 11:Capsules



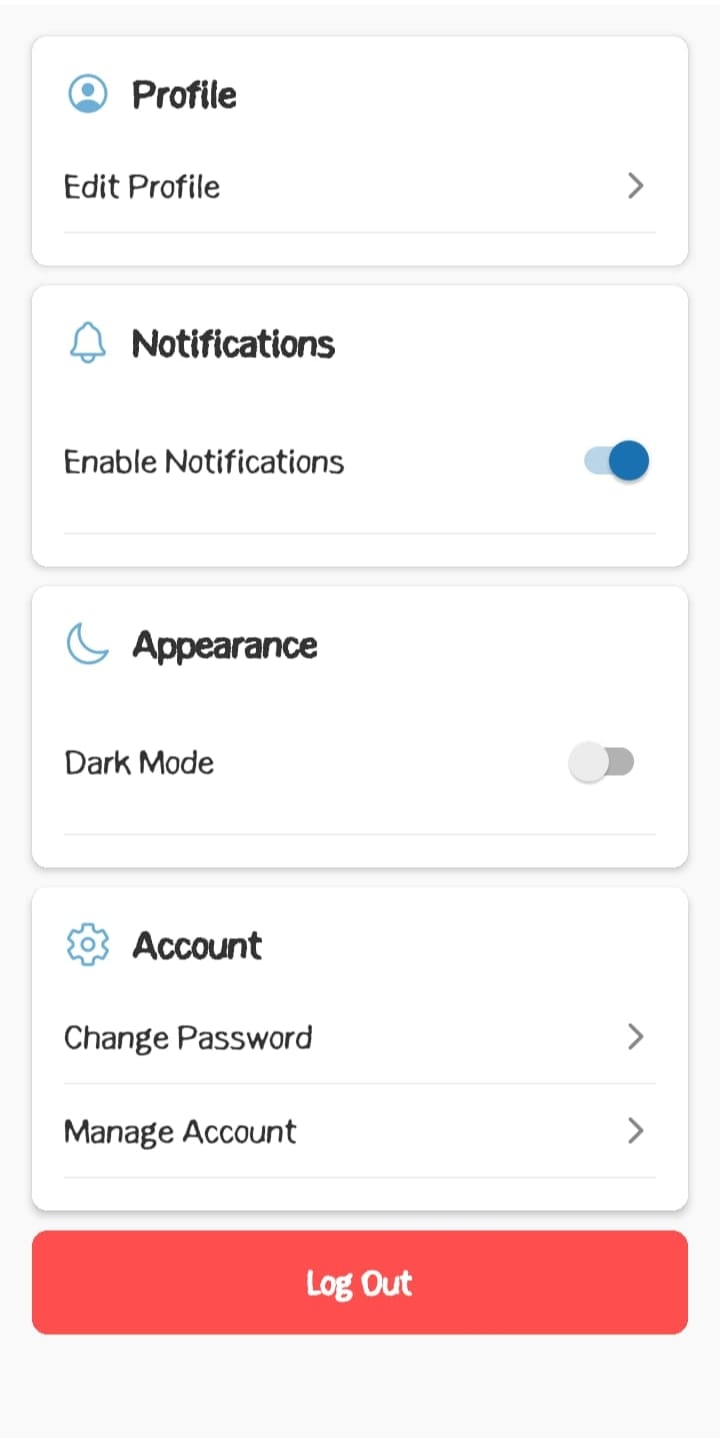


Figure 12:ShareCapsule Figure 13:Logout

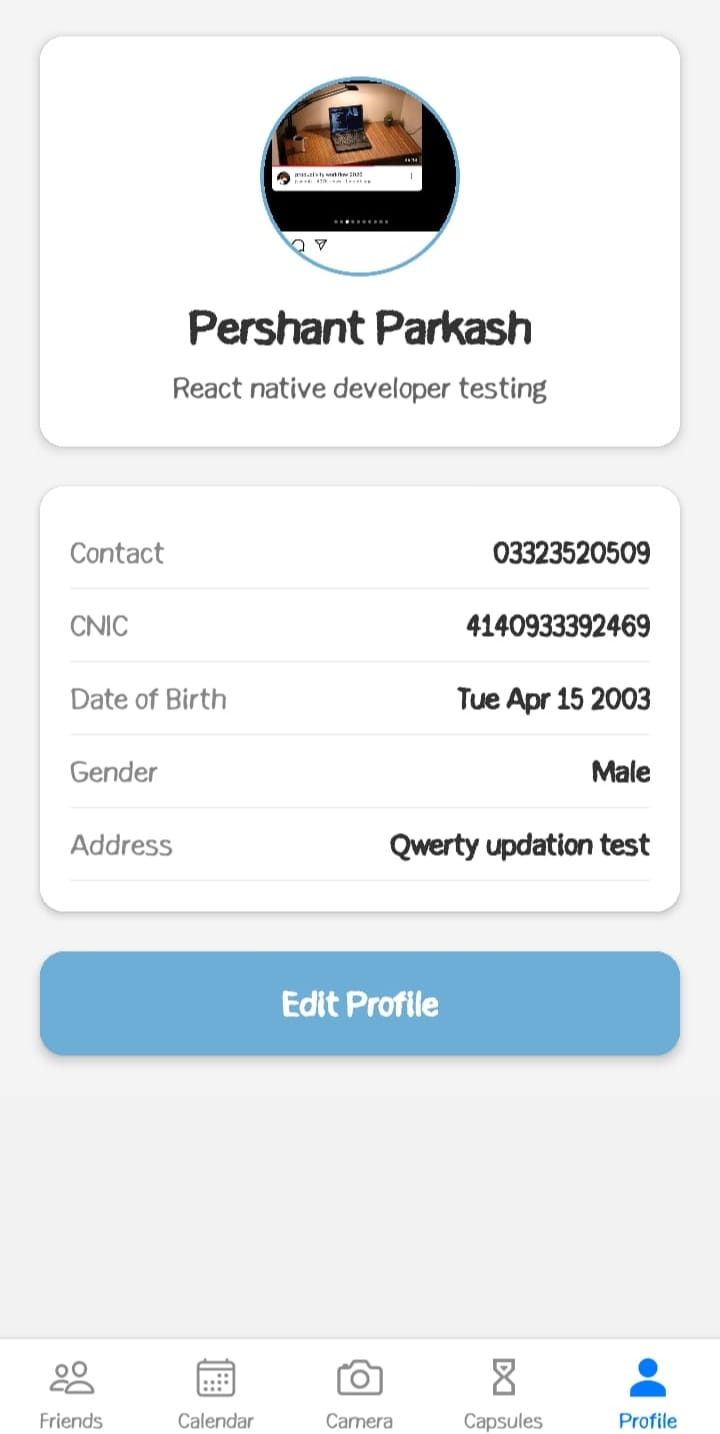
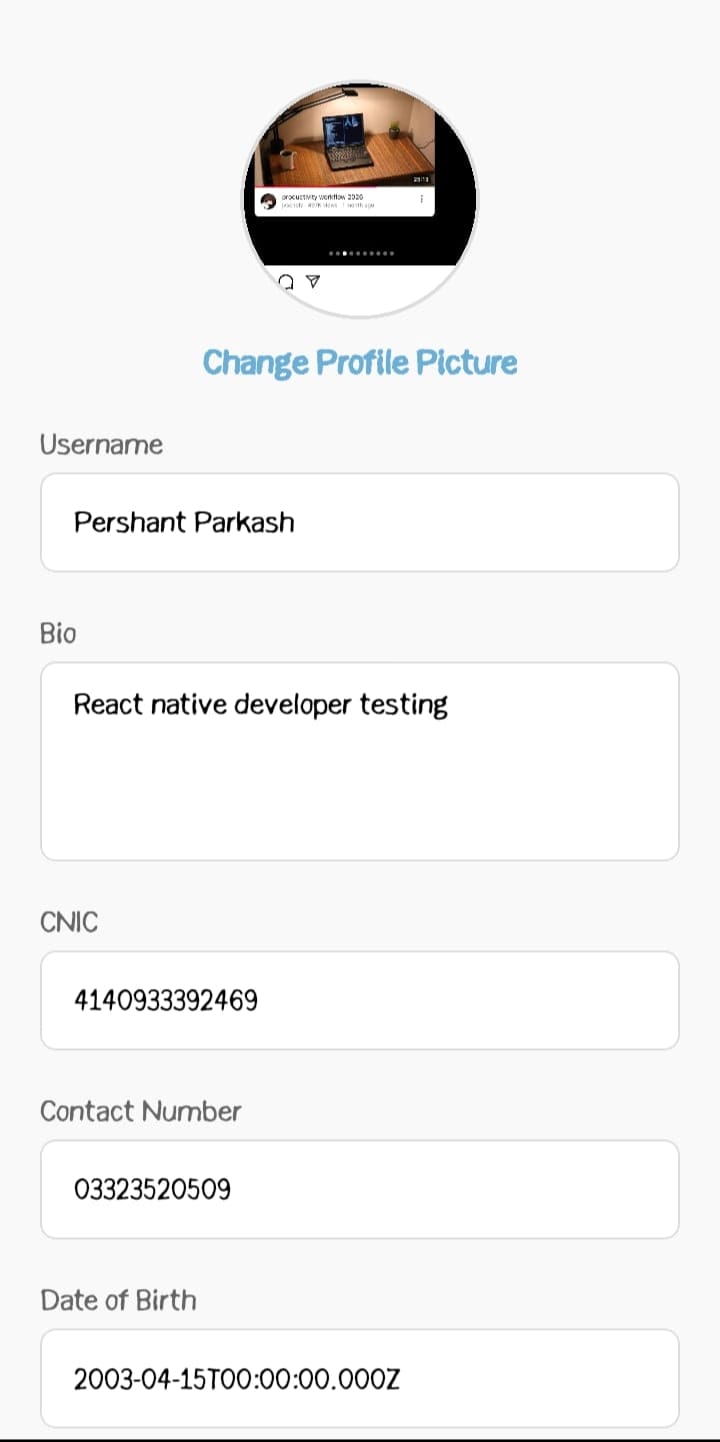


Figure 15 :ProfileEdit

Figure 14:ProfieUpdate

## Hardware Interfaces

Supports smartphones with camera and GPS functionality to enable media uploads and geo-tagging of capsules.

## Software Interfaces

The application uses RESTful APIs to facilitate communication between the front-end client and the back-end server. These APIs enable the system to interact with the MongoDB database, allowing for the creation, retrieval, updating, and deletion of data. MongoDB is used to store user profiles, time capsule metadata, and other relevant information. Its flexible schema makes it ideal for efficiently handling both structured and unstructured data, ensuring the application can scale and adapt to varying data requirements.

## Communications Interfaces

* **Protocols**: Uses HTTPS for secure data transmission.
* **Data Format**: JSON for all client-server exchanges.

# System Features

## Login and User Registration

### 4.1.1 Description and Priority

This feature allows users to register, log in, and log out securely, ensuring each user has a personalized and secure profile. It is a high-priority feature, as it provides the entry point for accessing Memora and managing time capsules

### Stimulus/Response Sequences

|  |
| --- |
| **Use Case Name:** Login and User Registration |
| **Description:** Users can sign up for an account, log in to access their existing profile, and log out when they are finished. |
| **Actors:** General users, family members, friends. |
| **Goal:** Create a new user account on the platform. |
| **Pre-Conditions:** user must not already have an account. User must have a valid email address. |
| **Basic Course of Events /Main Flow**   1. **User Action (Step 1)**:    * The user opens the Memora app and chooses to sign up or log in.   **System Response**:   * + The system displays a form for entering email, password, and other profile details. |

|  |
| --- |
| 1. **User Action (Step 2)**:    * This user submit the form.   **System Response**:   * + The system verifies the input and creates or accesses the user account.  1. **User Action (Step 3)**:    * The user logs out from their account.   **System Response**:  The system securely logs the user out and redirects to the welcome screen. |
| **Alternate Flow (Invalid Inputs)**   1. **User Action**:    * The user submits invalid data (e.g., incorrect email or weak password).   **System Response**:   * Highlights invalid fields and displays error messages:   + "Please enter a valid email."   + "Password must be at least 8 characters and include numbers." |
| **Exception Flow (System Errors) System Failure**:   * If a network or database error occurs during sign-up or login.   **System Response**   * Displays: "System temporarily unavailable. Please try again later." |
| **Post Conditions:** User is shown the Dashboard. |

Table 1:: Registration Stimulus/Response Sequences

### 4.1.3 Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Req. ID** | **Description** | **Priority** | **Status** |
| **Req. 1** | The system must validate email format and password strength during sign-up. | High | Pending |
| **Req. 2** | The system must allow users to log in with a registered email and password. | High | Pending |
| **Req. 3** | The system must provide error messages for invalid credentials or missing information. | High | Pending |

Table 2:: Registration Functional Requirements

## Time Capsule Creation

### Description and Priority

Users can create digital time capsules that contain media like images, videos, or audio, and set a future date for unlocking the capsule.

### Stimulus/Response Sequence

|  |
| --- |
| **Use Case Name:** Time Capsule Creation |
| **Description:** Users can create a time capsule with media and set a future unlock date. |
| **Actors:** User (Family Members, Friends) |
| **Goal:** To create a time capsule for future access**.** |
| **Pre-Conditions:** The user must be logged in. |
| **Basic Course of Events /Main Flow**  1. **User Action (Step 1):**  The user navigates to "Camera" and clicks "Capture the Moment."  **System Response:**  Displays a form for entering, uploading media, and setting an unlock date. |

|  |
| --- |
| 1. **User Action (Step 2)**:   The user fills out the form and clicks "Create Capsule."  **System Response**:  Validates the input:   * + Ensures the unlock date is in the future.   + Verifies media file formats (e.g., JPEG, MP4).   + Checks that mandatory fields are filled.   3. **User Action (Step 3)**:  The user submits the form with valid data.  **System Response**:  Saves the time capsule and displays: "Capsule created successfully." |
| **Alternate Flow (Invalid Inputs)**   * Na |
| **Exception Flow (System Errors) System Failure:**  A server or database error occurs while saving the capsule.  **System Response:**  Displays: "Error. Please try again later." |
| **Post Conditions:**   * Successful: The time capsule created successfully with its unlock date. * Unsuccessful: The user is prompted to correct errors and try again. |

Table 3:: Time Capsule Creation Stimulus/Response Sequences

### 4.2.4 Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Req. ID** | **Description** | **Priority** | **Status** |
| Req 1 | The system shall display a Screen for Creating memory, and setting an | High | Pending |

|  |  |  |  |
| --- | --- | --- | --- |
|  | unlock date. |  |  |
| Req 2 | The system shall validate the input data, ensuring the unlock date is  in the future. | High | Pending |
| Req 3 | The system shall save the time capsule in the database upon  successful submission. | High | Pending |
| Req 4 | The system shall display error messages for invalid inputs or  unsupported media formats. | High | Pending |

Table 4:: Time Capsule Creation Functional Requirements

## Profile and Friends Management

### Description and Priority

Users manages their profiles and add friends, enabling shared capsule experiences. Friends can be invited to view or collaborate on a Time Capsule.

### Stimulus/Response Sequences

|  |
| --- |
| **Use Case Name:** Profile and Friends Management |
| **Description:** Users can manage their profile and add friends or family members for capsule sharing |
| **Actors:** User (Family Members, Elders, Friends) |
| **Goal:** To manage user profiles and connect with friends or family for memory sharing**.** |
| **Pre-Conditions:** The user must be logged in. |
| **Basic Course of Events /Main Flow**   1. **User Action (Step 1)**:   The user navigates to the profile page and clicks "Edit Profile."  **System Response**:  Displays a form with profile fields (e.g., Username, profile picture, bio , cnic , address).   1. **User Action (Step 2)**:   The user edits details and clicks "Save Changes."  **System Response**:  Validates input and updates the profile in the system.   1. **User Action (Step 3)**:   The user navigates to the "Friends" section and there will be bunch of pending requests and also other profile/users to connect with.  **System Response**:  User must add and Accept/reject the Requests.   1. **User Action (Step 4)**:   The user sends the friend request.  **System Response**:  Sends the request and notifies the recipient. |

|  |
| --- |
|  |
| **Alternate Flow (Invalid Inputs)**   * NA |
| **Exception Flow (System Errors) System Failure**:  A system error occurs while updating the profile or sending a friend request.  **System Response**:  Displays: "Unable to update profile or send friend request. Please try again later." |
| **Post Conditions:**   * Successful: Profile updates are saved, and friends are added. * Unsuccessful: Errors are displayed, and the user remains on the current screen. |

Table 5:: Profile and Friends Management Stimulus/Response Sequences

### Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Req. ID** | **Description** | **Priority** | **Status** |
| Req 1 | The system shall display a form for profile updates, including fields  for Username, bio, cnic, address | High | Pending |
| Req 3 | The system shall validate profile and friend request inputs. | High | Pending |
| Req 4 | The system shall notify the user of errors or successful updates. | High | Pending |

Table 6:: Profile and Friends Management Functional Requirements

## Memory Capsules on Map

### Description and Priority

Displays capsules on a map, showing where memories were created or shared. Users can explore capsules based on location.

### Stimulus/Response Sequences

|  |
| --- |
| **Use Case Name:** Memory Capsules on Map |
| **Description:** Users can view the locations of their capsules on an interactive map. |
| **Actors:** User (Family Members, Friends). |
| **Goal:** Visualize memory capsules geographically**.** |
| **Pre-Conditions:** Users must have created time capsules with location data. |
| **Basic Course of Events /Main Flow**   1. **User Action (Step 1):**   The user navigates to the "Map" section.  **System Response**:  Loads an interactive map displaying capsule locations as pins.   1. **User Action (Step 2)**:   The user clicks on a pin to view capsule details.  **System Response**:  Displays information about the capsule, such as title and unlock date.  . |
| **Alternate Flow (Invalid Inputs)**  **User Action**:  The user attempts to view a capsule without Geolocation data.  **System Response**:   * Displays: "Capsule location unavailable." |
| **Exception Flow (System Errors) System Failure**:  Map service fails to load or capsule data retrieval is unsuccessful.  **System Response**:  Displays: "Unable to load map. Please try again later." |
| **Post Conditions:**   * Successful: Capsules are displayed on the map with their respective locations. * Unsuccessful: The user is notified and remains on the map view. |

Table 7:: Memory Capsules on Map Stimulus/Response Sequences

### Functional Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req. ID** | | **Description** | **Priority** | **Status** |
| Req 1 | The system shall display capsule locations on an interactive  map. | | High | Pending |
| Req 2 | The system shall allow users to click on map pins to view  capsule details. | | High | Pending |
| Req 3 | The system shall display an error if the map service fails. | | Medium | Pending |

Table 8:: Memory Capsules on Map Functional Requirements

## Emotional Connection in Capsule

### Description and Priority

Allows users to add text or audio teasers that can be accessed before the capsule officially opens, adding an emotional anticipation.

### Stimulus/Response Sequences

|  |
| --- |
| **Use Case Name:** Emotional Connection in Capsule |
| **Description:** Users can add emotional content such as text or audio to their time capsules. This content can be accessed before the capsule unlocks, fostering anticipation and emotional engagement. |
| **Actors:** User (Family Members, Friends) |
| **Goal:** To create pre-unlock emotional triggers for time capsules. |
| **Pre-Conditions:** The user must be logged in and A time capsule must already exist.. (this capsule will be create on option salection in a capsule creation form) |
| **Basic Course of Events /Main Flow**   1. **User Action (Step 1)**:   The user selects a time capsule and clicks on "Add Emotional Connection."  **System Response**:  Displays a form for adding text or recording/uploading audio as emotional content.   1. **User Action (Step 2)**:   The user enters the text or uploads/records the audio and clicks "Save."  **System Response**:  Validates the input:   * + Ensures the text is within character limits. |

|  |
| --- |
| * Checks audio file formats (e.g., MP3, WAV) and size.   3. **User Action (Step 3)**:  The user submits the form.  **System Response**:  Saves the emotional content and associates it with the selected time capsule. Displays: "Emotional content added successfully." |
| **Alternate Flow (Invalid Inputs)**  **User Action**:  The user uploads unsupported file formats or exceeds audio size limits.  **System Response**:   * Highlights invalid fields and displays error messages like:   + "File size exceeds the allowed limit."   + "Unsupported file format. Please upload a valid audio file." |
| **Exception Flow (System Errors)**  **System Failure**:  An error occurs while saving the emotional content to the database.  **System Response**:  Displays: "System error. Unable to save emotional content. Please try again later." |
| **Post Conditions:**   * **Successful**: Emotional content is saved and linked to the capsule for pre-unlock access. * **Unsuccessful**: The user is notified of the issue and remains on the form for corrections.   . |

Table 9:: Emotional Connection in Capsule Stimulus/Response Sequences

### Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Req. ID** | **Description** | **Priority** | **Status** |
| Req 1 | The system shall allow users to add emotional content (text or audio)  to time capsules. | High | Pending |
| Req 2 | The system shall validate audio files for format and size limits before | High | Pending |

|  |  |  |  |
| --- | --- | --- | --- |
|  | submission. |  |  |
| Req 3 | The system shall securely save emotional content associated with the  time capsule. | High | Pending |
| Req 4 | The system shall display error messages for invalid inputs or system  errors. | High | Pending |

Table 10:: Emotional Connection in Capsule Functional Requirements

## Nested Capsules

### Description and Priority

Users can create multiple capsules within one, all accessible on a single unlock date.

### Stimulus/Response Sequences

|  |
| --- |
| **Use Case Name:** Nested Capsules |
| **Description:** Users can create multiple smaller capsules within a main capsule, unlocking on the same date. |
| **Actors:** User (Family Members, Elders, Friends) |
| **Goal:** To group related memories into a nested capsule structure. |
| **Pre-Conditions:** The user must be logged in and have a main capsule created. |
| **Basic Course of Events /Main Flow**   1. **User Action (Step 1)**:   The user selects a main capsule and chooses "Add Nested Capsule."  **System Response**:  Displays a form for adding a nested capsule with media, and unlock date.   1. **User Action (Step 2)**:   The user enters details and clicks "Save."  **System Response**:  Validates the input:   * + Verifies media formats and mandatory fields.  1. **User Action (Step 3)**:   The user submits the nested capsule form.  **System Response**: |

|  |
| --- |
| Saves the nested capsule under the main capsule. |
| **Alternate Flow (Invalid Inputs)**  **User Action**:  The user sets a mismatched unlock date or uploads unsupported media.  **System Response**:  Displays: "Unlock date must match the main capsule." |
| **Exception Flow (System Errors)**  **System Failure**:  An error occurs while saving the nested capsule.  **System Response**:  Displays: "Unable to save nested capsule. Please try again later." |
| **Post Conditions:**   * Successful: The nested capsule is created and linked to the main capsule. * Unsuccessful: Errors are displayed, and the user remains on the form. |

Table 11:: Nested Capsules Stimulus/Response Sequences

### Functional Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req. ID** | | **Description** | **Priority** | **Status** |
| Req 1 | The system shall allow users to add nested capsules to a main  capsule. | | High | Pending |
| Req 2 | The system shall save nested capsules under their respective  main capsules. | | High | Pending |
| Req 3 | The system shall notify users of invalid inputs or system errors. | | High | Pending |

Table 12:: Nested Capsules Functional Requirements

## Media Upload/Creation Options in Capsule

### Description and Priority

Enables users to upload various types of media, such as images, videos, and audio, to enrich their time capsules and preserve diverse memories

### Stimulus/Response Sequences

|  |
| --- |
| **Use Case Name:** Media Upload/Creation Options in Capsule |
| **Description:** Users can upload various types of media, including text, images, videos, and audio, to enrich their time capsules |
| **Actors:** User (Family Members, Elders, Friends) |
| **Goal:** Add diverse media content to a time capsule. |
| **Pre-Conditions**: The user must be logged in and a new or existing time capsule must be selected for adding media. |
| **Basic Course of Events /Main Flow**   1. **User Action (Step 1)**:   The user selects a capsule and clicks "Add Media."  **System Response**:  Displays options to upload text, images, videos, or audio files.   1. **User Action (Step 2)**:   The user selects media files and clicks "Upload."  **System Response**:  Validates the media:   * + Ensures file formats are supported (e.g., JPG, MP4, MP3).   + Checks file size limits.  1. **User Action (Step 3)**:   The user submits the media.  **System Response**:  Saves the media to the capsule and displays a success message: "Media uploaded successfully." |
| **Alternate Flow (Invalid Inputs)**  **User Action**:  The user uploads unsupported file types or exceeds file size limits.  **System Response**:   * Highlights invalid fields and displays error messages: * "File size exceeds the allowed limit." * "Unsupported file format. Please upload a valid file." |

|  |
| --- |
| **Exception Flow (System Errors) System Failure**:  An error occurs while uploading media to the database.  **System Response**:  Displays: "System error. Unable to upload media. Please try again later." |
| **Post Conditions:**   * **Successful**: Media is added to the capsule. * **Unsuccessful**: Errors are displayed, and the user is prompted to retry |

Table 13:: Media Upload Options in Capsule Stimulus/Response Sequences

### Functional Requirements

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Req. ID** | **Description** | | **Priority** | | **Status** |
| Req 1 | The system shall allow users to create images, videos, and audio to capsules. | | High | Pending | |
| Req 2 | | The system shall validate the file format and size limits of uploaded media. | High | Pending | |
| Req 3 | The system shall securely save media content to the selected capsule. | | High | Pending | |
| Req 4 | The system shall display error messages for unsupported formats or system errors. | | High | Pending | |

Table 14:: Media Upload Options in Capsule Functional Requirements

## Shared Capsules

### Description and Priority

Allows users to share their time capsules with selected friends or family members, granting them access after the unlock date for collaborative or shared experiences.

### Stimulus/Response Sequences

|  |
| --- |
| **Use Case Name:** Shared Capsules |
| **Description:** Users can share their capsules with friends or family, allowing them to view the content after the unlock date |
| **Actors:** User (Family Members, Elders, Friends) |
| **Goal:** Share capsules with selected users for collaborative or shared access. |
| **Pre-Conditions**: The user must have friends or family added in their profile. |

|  |
| --- |
| **Basic Course of Events /Main Flow**   1. **User Action (Step 1)**:   The selected a capsule will be "Shared"  **System Response**:  Displays a list of friends and family for selection.   1. **User Action (Step 2)**:   The user selects Friends and clicks "Send."  **System Response**:  Notifies the selected users about the shared capsule and confirms the action. |
| **Alternate Flow (Invalid Inputs)**  **User Action**:  The user selects an invalid recipient (e.g., non-existent user or restricted contact).  **System Response**:  Displays: "Recipient not found. Please select a valid contact." |
| **Exception Flow (System Errors) System Failure**:  An error occurs while sharing the capsule.  **System Response**:  Displays: "System error. Unable to share capsule. Please try again later." |
| **Post Conditions:**   * Successful: The capsule is shared with the selected recipients. * Unsuccessful: Errors are displayed, and the user is prompted to retry. |

Table 15:: Shared Capsules Stimulus/Response Sequences

### Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Req. ID** | **Description** | **Priority** | **Status** |
| Req 1 | The system shall allow users to share capsules with specific friends or family. | High | Pending |
| Req 2 | The system shall notify recipients about the shared capsule. | High | Pending |
| Req 3 | The system shall display an error message for invalid recipients. | Medium | Pending |

Table 16:: Shared Capsules Functional Requirements

# Nonfunctional Requirements

## Performance Requirements

* The app should load the dashboard/Camera within seconds.
* Capsules should open exactly on the unlock date being reached.

## Safety Requirements

* + - * **Strong Authentication**: Users are required to authenticate through a secure process to ensure that only authorized individuals can access their accounts and capsules.
      * **Session Timeouts**: The application automatically logs out inactive users after a specified period to prevent unauthorized access.

## Security Requirements

The Memora app secures user data with encryption, strong authentication, and session timeouts. Capsules are protected, and communications use HTTPS to ensure secure data transmission.

## Software Quality Attributes

* **Availability**: 99.9% uptime.
* **Scalability**: Supports increasing users without performance issues.
* **Security**: encryption for all user data. (not all data is encrypted)

## Business Rules

## User Roles and Permissions:

* + - * Only registered users can create Time Capsules.
      * Friends and family members can access shared capsules only after the designated unlock date.
      * Users can only send friend requests to other registered users.

## Capsule Creation:

* + - * Users cannot edit the contents of a Time Capsule once it is locked with an unlock date. Once the capsule is locked, no further edits are allowed.
      * Sub-capsules within a nested capsule follow the same rules as their parent capsule in terms of unlock date and visibility.

## Milestone and Goal-Based Capsules:

* + - * Milestone capsules will remain locked until the user completes the required goal. The app will not allow manual override to unlock these capsules early.
      * Users must set at least one goal before creating a milestone capsule.

## Notifications and Reminders:

* + - * Users will receive notifications only for capsules they have created or for shared capsules they are a part of.

## Profile and Community Management:

* + - * Users can only join a community group if they receive an invite from a group admin or request to join is approved.

## Data Privacy and Security:

* + - * All user data, including Time Capsules and profile information, is encrypted to protect against unauthorized access.

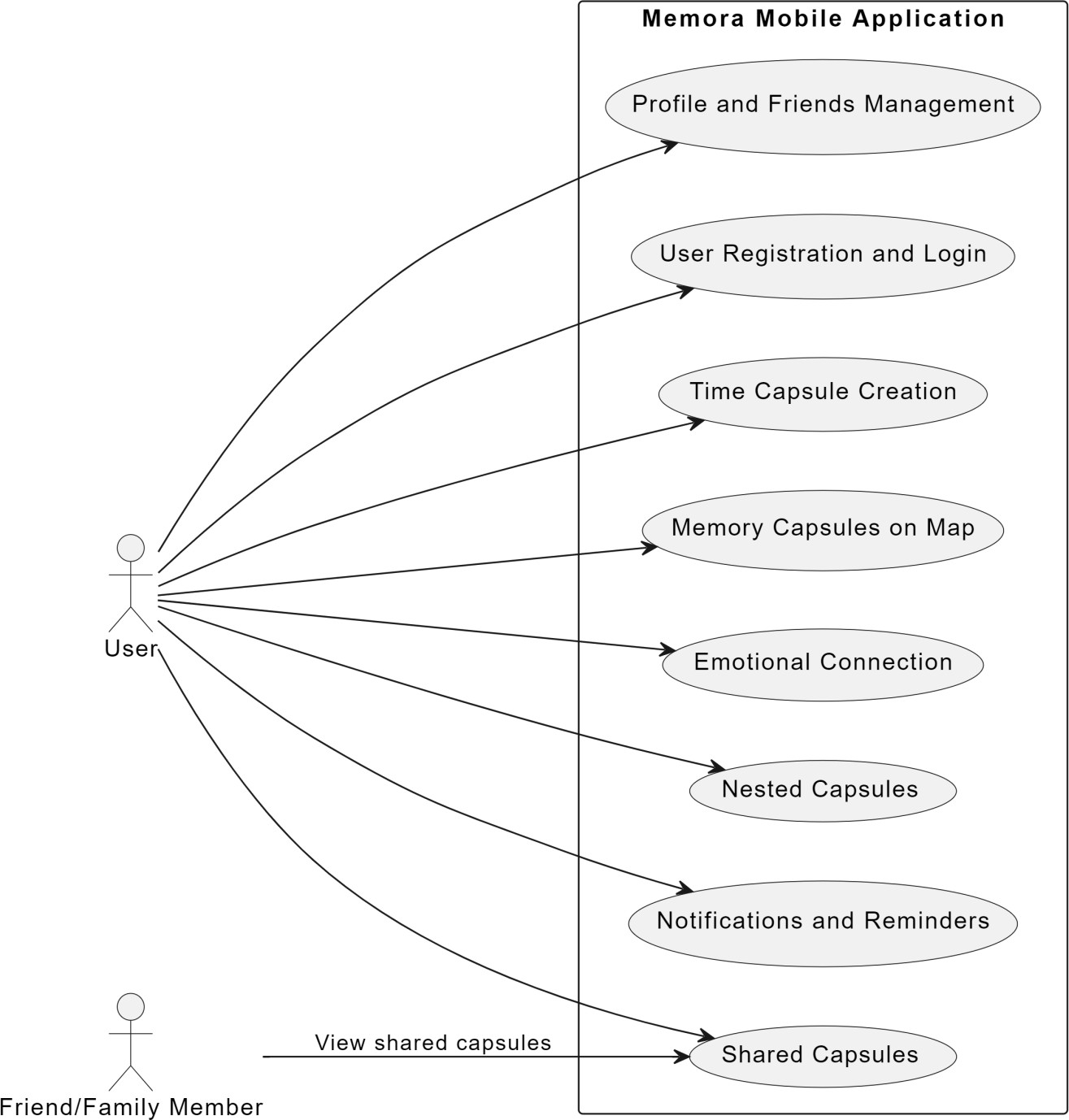
# Other Requirements

The system must support integration with third-party APIs for notifications and data syncing. Additionally, it should comply with local and international data privacy laws to ensure user security.

# Appendix A: Glossary

* **Time Capsule**: A digital container where users can store text, images, videos, and audio, which is locked and only accessible after a set date or milestone is reached.
* **Shared Capsule**: A Time Capsule that can be shared with specific users like friends, family, or community groups for a collective opening experience.
* **Memory Calendar**: A feature that displays upcoming dates when Time Capsules are set to unlock.
* **Nested Capsule**: A capsule that contains multiple sub-capsules, all of which are accessible on the same unlock date.
* **Unlock Milestone Capsule**: A capsule that unlocks when the user completes a specific goal or milestone.
* **Profile Management**: The process of creating and managing a user profile, including adding friends or family.
* **Community Group**: A group where multiple users can join and share Time Capsules with each other.
* **UI**: User Interface, the part of the application that users interact with directly.
* **API**: Application Programming Interface, used for communication between different software components.

# Appendix B: Analysis Models

****

Figur:: Use case Diagram

# Appendix C: To Be Determined List

* **TBD-01**: Selection of third-party API providers for sending notifications and integrating map services.
* **TBD-02**: Development of user testing plans, including testing dates and methods for evaluating FYP-II features.
* **TBD-03**: Final design and functionality details for the profile management interface.
* **TBD-04**: Criteria and metrics for users to achieve specific goals tied to unlocking milestone capsules.