CEBU INSTITUTE OF TECHNOLOGY
UNIVERSITY

IT342-G1
SYSTEMS INTEGRATION AND ARCHITECTURE 1

FUNCTIONAL REQUIREMENTS SPECIFICATION (FRS)

Project Title: User Registration and Authentication SRS

Prepared By: Frances Aaliyah Maturan

Date of Submission: February 06, 2026

Version: 1

Table of Contents

1. Introduction....3
   1.1. Purpose....3
   1.2. Scope....3
   1.3. Definitions, Acronyms, and Abbreviations....3
2. Overall Description....3
   2.1. System Perspective....3
   2.2. User Classes and Characteristics....3
   2.3. Operating Environment....4
   2.4. Assumptions and Dependencies....4
3. System Features and Functional Requirements....4
   3.1. Feature 1:....4
   3.2. Feature 2:....5
   3.4. Feature 3:....5
   3.5. Feature 4:....5
4. Non-Functional Requirements....5
5. System Models (Diagrams)....6
   5.1. ERD....6
   5.2. Use Case Diagram....6
   5.3. Activity Diagram....7
   5.4. Class Diagram....8
   5.5. Sequence Diagram....9
6. Appendices....10

1. Introduction

1.1. Purpose

The purpose of this document is to describe the user authentication system of the application. This system allows users to create an account, log in, view their profile or dashboard, and log out securely. The intended audience of this document includes students, instructors, and developers who want to understand how the authentication system works.

1.2. Scope

The system focuses on user account management, including registration, login, logout, and access to protected pages such as the user profile or dashboard. The system does not cover advanced features like payment systems or admin controls yet. Its main goal is to control user access and keep user information secure.

1.3. Definitions, Acronyms, and Abbreviations

User – A person who creates an account and uses the system
Authentication – The process of verifying a user's identity
Registration – The process of creating a new user account
Login – The process of entering credentials to access the system
Logout – The process of ending a user session
Dashboard/Profile – A page where the user can see their personal information
Credentials – Information such as email and password used to log in

2. Overall Description

2.1. System Perspective

The user authentication system is a core part of the application. It works together with other parts of the system by controlling who can access certain pages. Only registered and logged-in users are allowed to view protected pages like the profile or dashboard.

2.2. User Classes and Characteristics

Guest User

- Has no account or is not logged in
- Can only access public pages such as login and registration

Authenticated User

- Has a valid account
- Can log in, view their profile/dashboard, and log out

2.3. Operating Environment

Hardware:

- A personal computer or laptop for development
- Android mobile device or emulator for testing the mobile application
- Server hardware provided by a cloud hosting service

Software:

- **Backend:** Spring Boot (Java) for handling user authentication, registration, and session management
- **Frontend:** ReactJS for building the web user interface
- **Mobile Application:** Android application developed using Kotlin
- **Database:** MySQL, PostgreSQL, or MongoDB for storing user account information
- **Operating System:** Windows, macOS, or Linux

Development Tools and Services:

- **Version Control:** Git for source code management and GitHub for repository hosting
- **Deployment:** Cloud hosting platforms such as Render, Vercel, Railway, or Google Cloud for deploying the system
- **IDE/Editors:** IntelliJ IDEA, Visual Studio Code, or Android Studio

2.4. Assumptions and Dependencies

- Users have access to the internet
- Users provide valid and correct information during registration
- The system depends on a database to store user accounts
- The system assumes that passwords are stored securely

3. System Features and Functional Requirements

Describe each major feature of the system and its functional requirements.

3.1. Feature 1: User Registration

Description: This feature allows new users to create an account by providing the required information.

Functional Requirements:

-The system shall allow users to register using an email and password

-The system shall check if the email already exists

-The system shall store the user's information in the database

3.2. Feature 2: User Login

Description: This feature allows registered users to log in to the system using their credentials.

Functional Requirements:

-The system shall allow users to log in using a valid email and password
-The system shall verify the user's credentials
-The system shall redirect the user to their dashboard after successful login

3.3. Feature 2: View Profile/Dashboard

Description: This feature allows logged-in users to view their personal profile or dashboard.

Functional Requirements:

-The system shall allow only logged-in users to access the profile/dashboard
-The system shall display user-related information
-The system shall prevent access to this page when the user is logged out

3.4. Feature 2: User Logout

Description: This feature allows users to safely log out of the system.

Functional Requirements:

-The system shall allow users to log out at any time
-The system shall end the user session after logout
-The system shall redirect the user to the login page

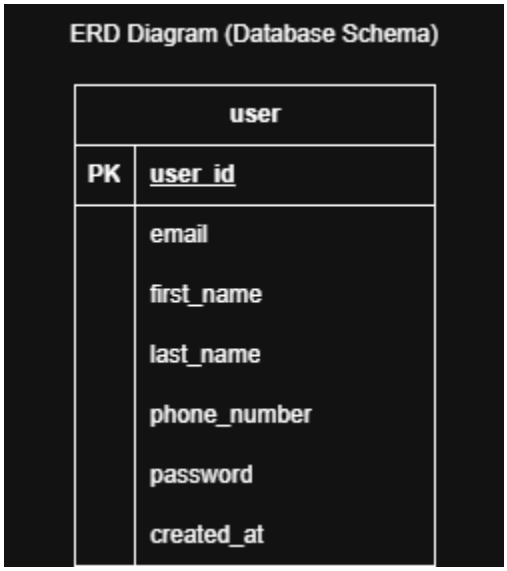
4. Non-Functional Requirements

Security: User passwords must be protected and not visible to other users
Usability: The login and registration process should be easy to understand
Performance: Login and logout actions should respond quickly
Reliability: The system should consistently allow or deny access based on login status

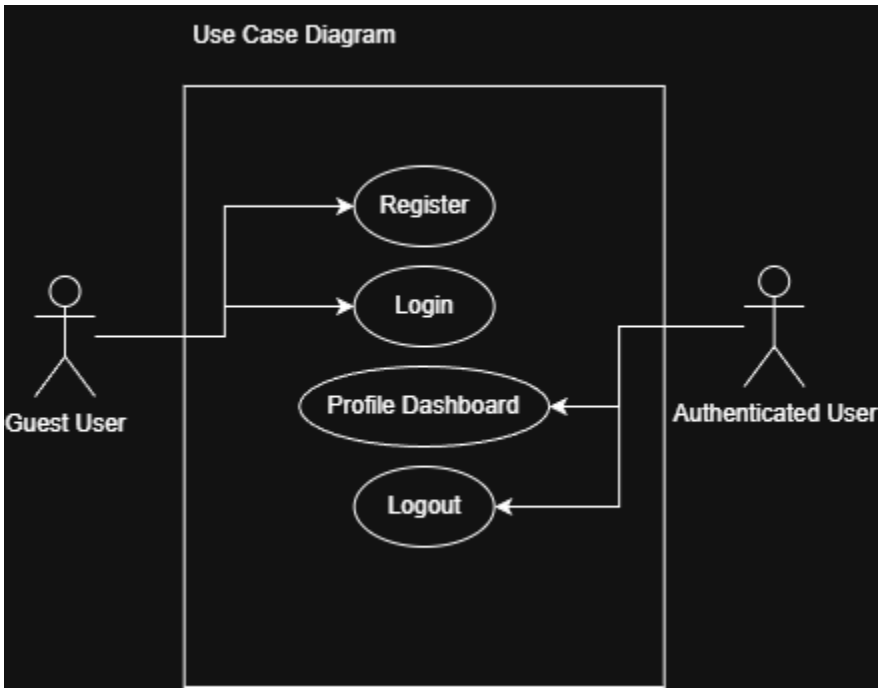
5. System Models (Diagrams)

Insert the necessary diagrams for the system:

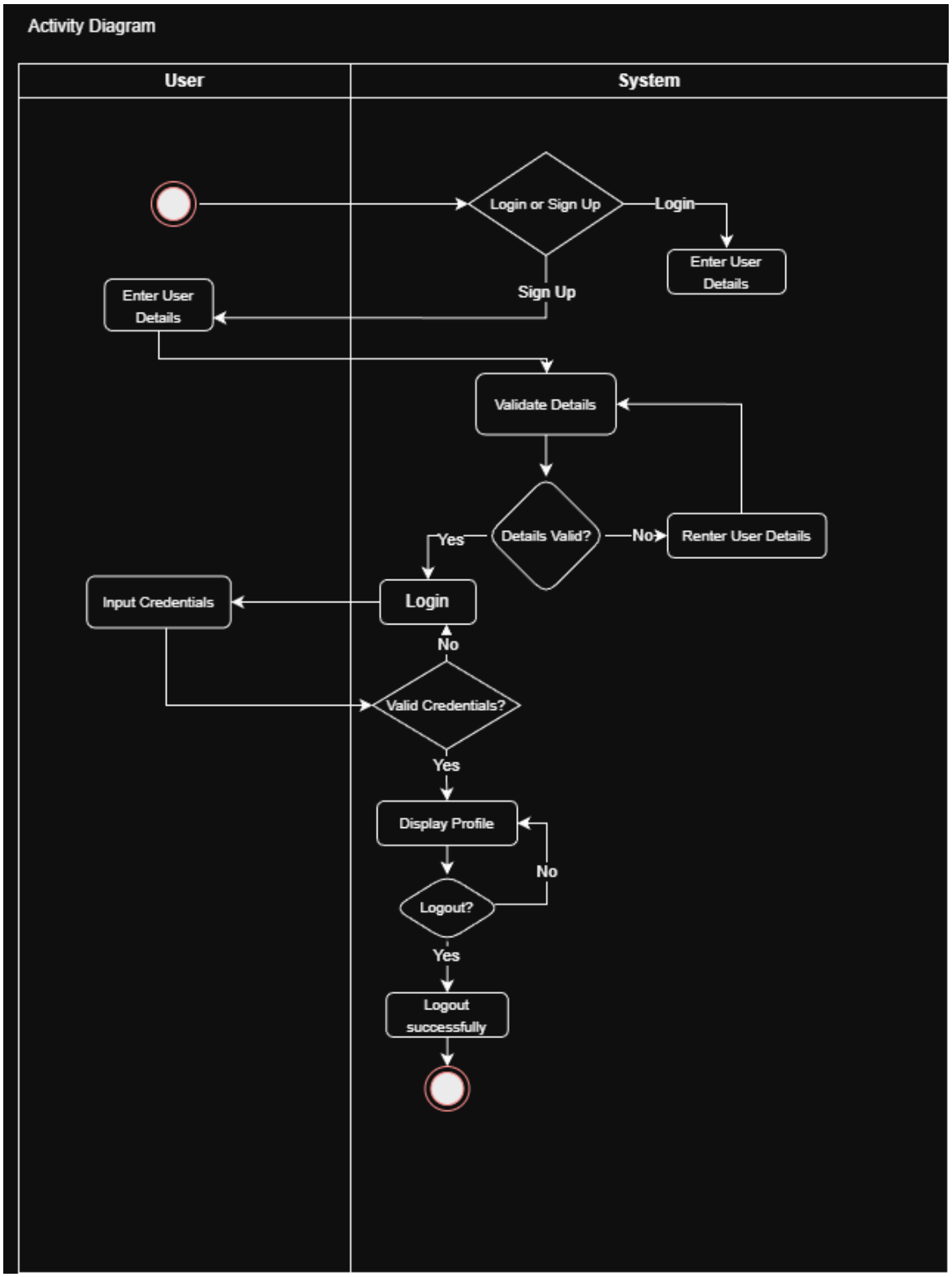
5.1. ERD



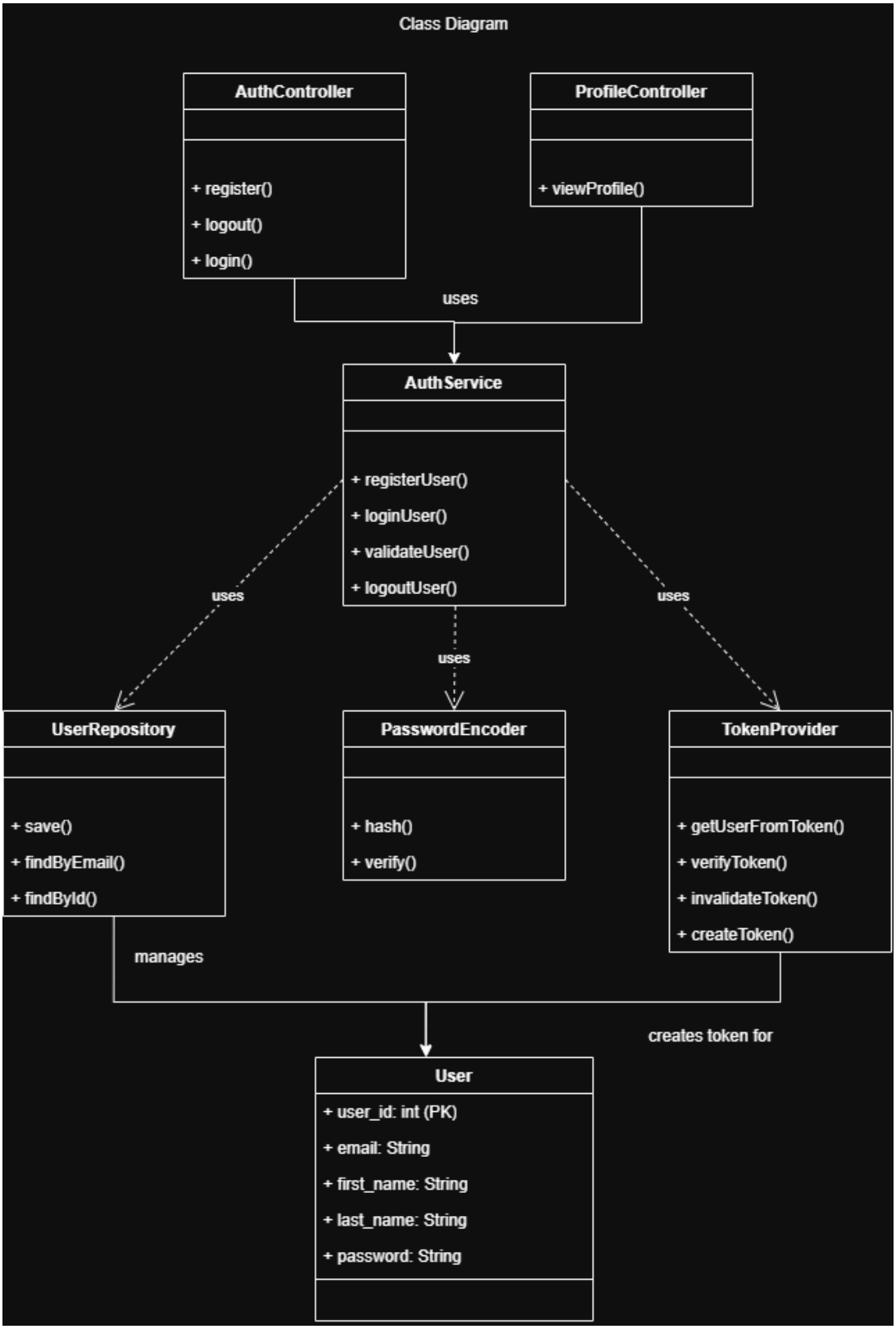
5.2. Use Case Diagram



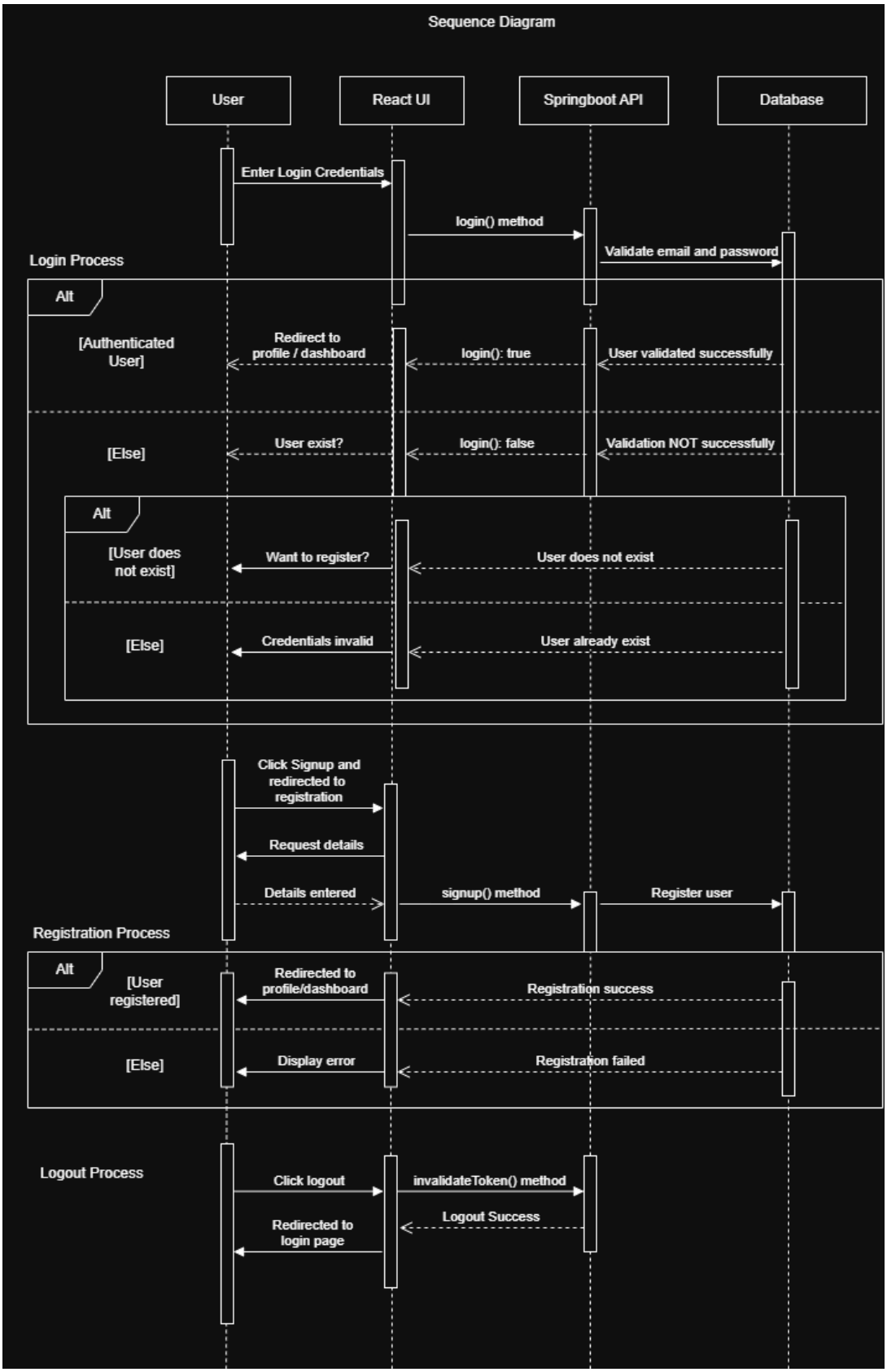
5.3. Activity Diagram



5.4. Class Diagram

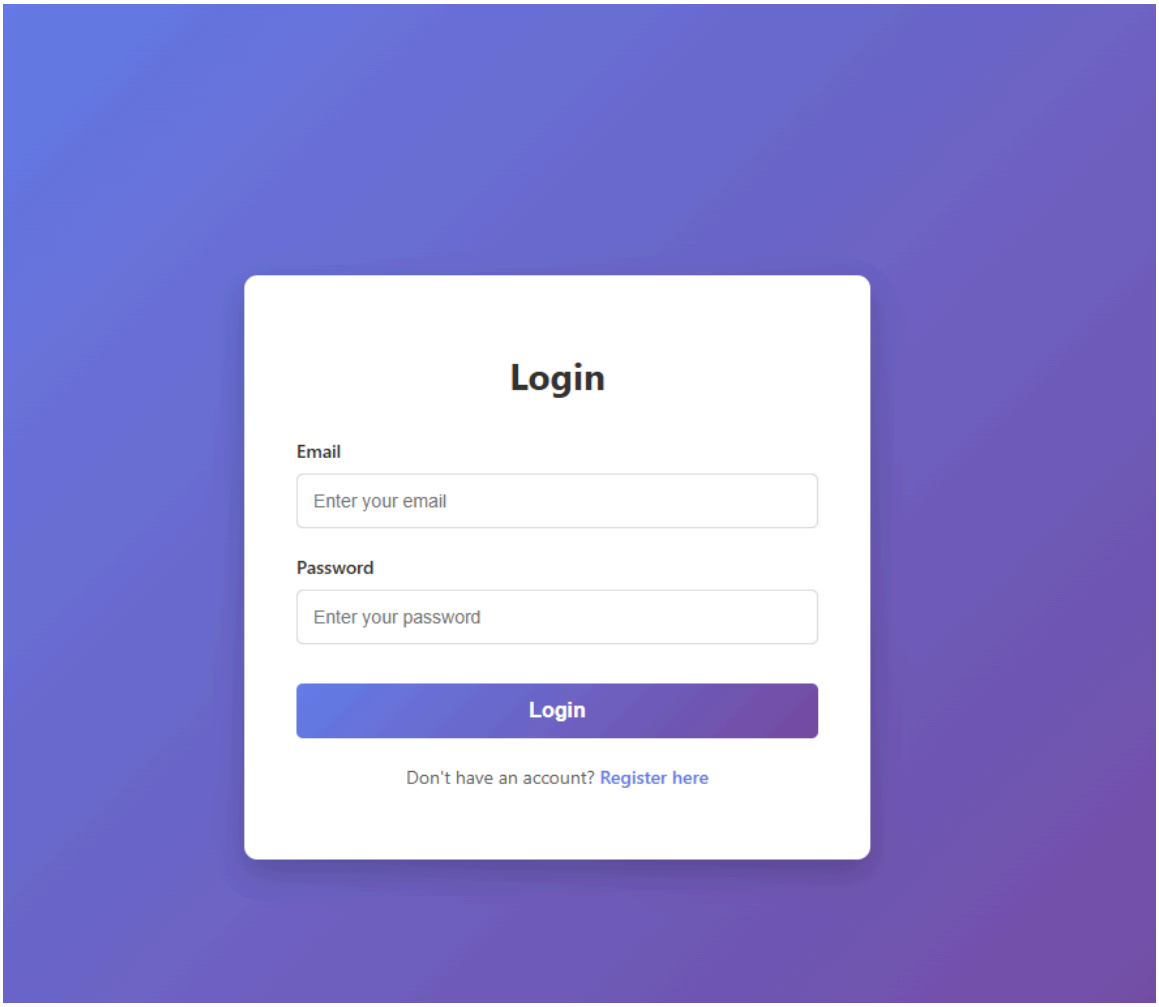


5.5. Sequence Diagram

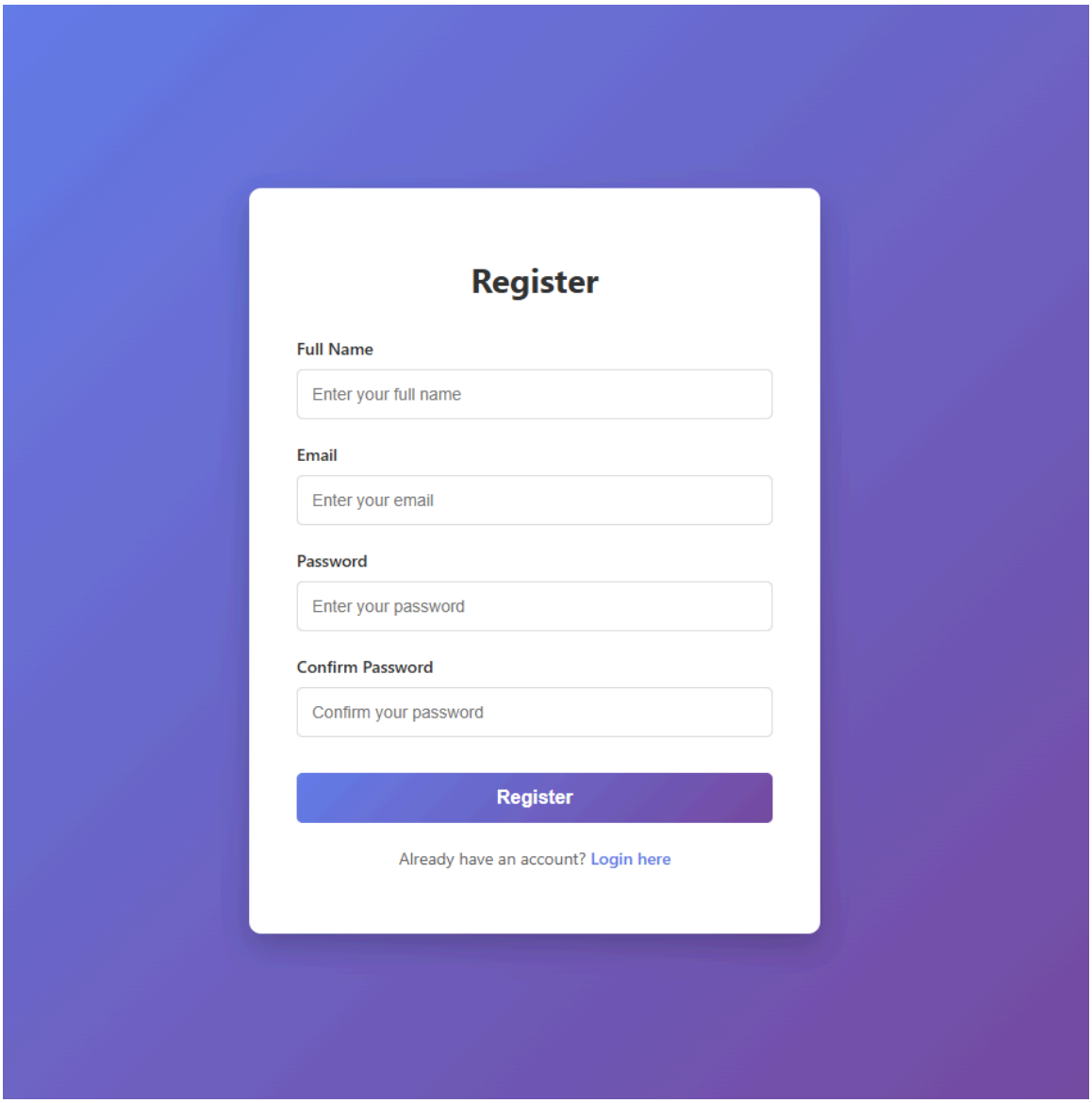


Screenshots from Web UI

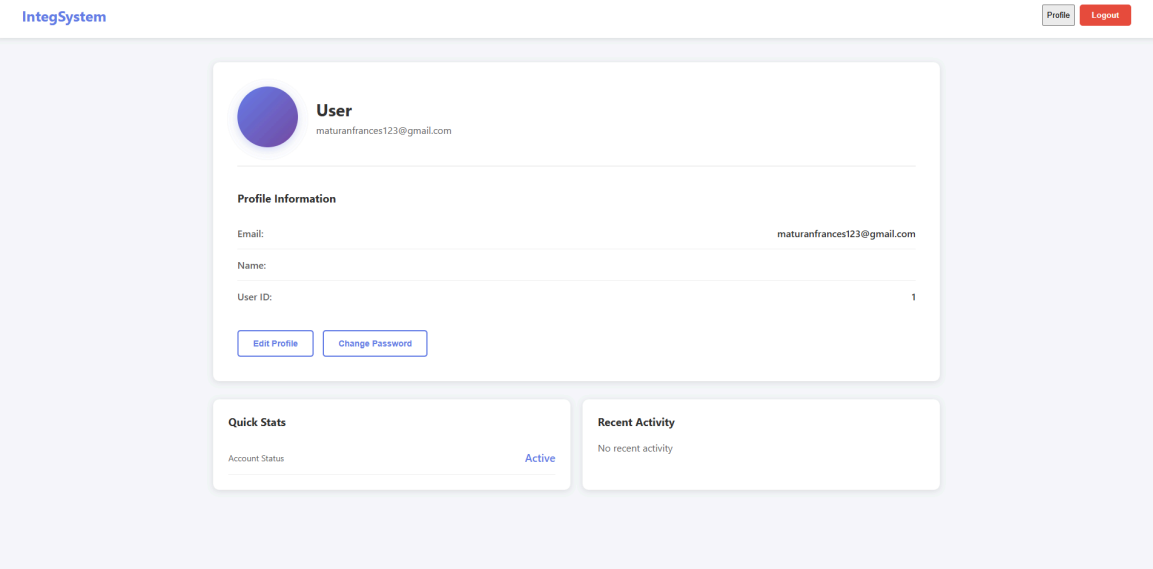
Login Screenshot



Register Screenshot



Profile Screenshot



Screenshots from mobile UI

Login Screenshot

Register Screenshot

Profile Screenshot

6. Appendices