CEBU INSTITUTE OF TECHNOLOGY
UNIVERSITY

IT342-Section SYSTEMS INTEGRATION AND ARCHITECTURE 1

FUNCTIONAL REQUIREMENTS SPECIFICATION (FRS)

Project Title: Mini App – User Registration & Authentication

Prepared By: Artazo Airon Kit A.

Date of Submission: 08/02/2026

Version:

Table of Contents

1. Introduction

1.1. Purpose

The purpose of this system is to design a mini application that handles user registration, login, logout, and profile viewing with proper authentication. This document describes the functional and non-functional requirements of the system and serves as a guide for understanding how the system operates. The intended audience of this document includes students, instructors, and developers involved in the project.

1.2. Scope

The system allows users to create an account, log in using their credentials, view their profile/dashboard after authentication, and log out securely. The system also ensures that protected pages cannot be accessed when the user is not logged in. This project focuses only on the authentication and user management part of a web application.

1.3. Definitions, Acronyms, and Abbreviations

Authentication – Verifying the identity of a user.

Authorization – Granting access to protected resources.

ERD – Entity Relationship Diagram.

UI – User Interface.

API – Application Programming Interface.

JWT/Token – A token used to maintain user session.

React UI – Frontend interface.

Spring Boot API – Backend server.

2. Overall Description

2.1. System Perspective

The system follows a client-server architecture. The React UI serves as the frontend where users interact with the system. The Spring Boot API handles authentication logic and communicates with the database to store and retrieve user information.

2.2. User Classes and Characteristics

- **Guest User**
  - Can register an account
  - Can log in
- **Authenticated User**

- Can view profile/dashboard
- Can log out
- Cannot access pages after logout

2.3. Operating Environment

Frontend: ReactJS (HTML, CSS, JavaScript)

Backend: Spring Boot (Java)

Database: MySQL / any relational database

Tools: draw.io / diagrams.net for diagrams

Web Browser: Chrome, Edge, Firefox

2.4. Assumptions and Dependencies

Users have internet access and a browser.

The database server is running properly.

Backend API is available and connected to the database.

Users provide valid email and password during registration.

3. System Features and Functional Requirements

This feature allows a guest user to create a new account in the system.

3.1. Feature 1:

Description:
The system shall allow users to enter name, email, and password.

The system shall validate if the email is not already registered.

The system shall store the user data securely in the database.

The system shall encrypt the password before saving.

The system shall confirm successful registration.

3.2. Feature 2:

Description:
The system shall allow users to input email and password.

The system shall verify the credentials from the database.

The system shall generate a token/session after successful login.

The system shall redirect the user to the profile/dashboard.

The system shall display an error message for invalid credentials.

4. Non-Functional Requirements

Security: Passwords must be encrypted. Protected routes must require authentication.

Performance: System should respond to login/registration within 2 seconds.

Usability: Interface should be simple and user-friendly.

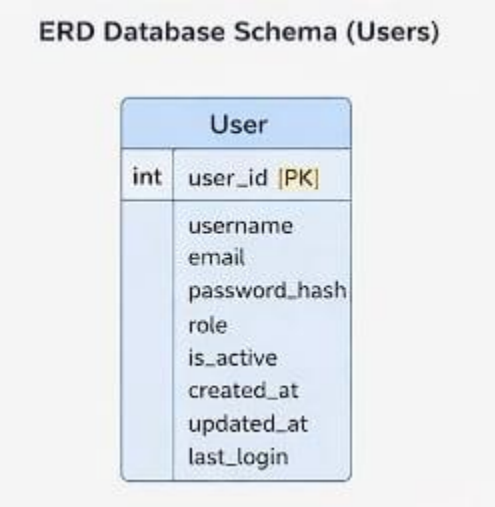
Reliability: System should handle multiple user requests without crashing.

Scalability: System should allow future addition of more features.

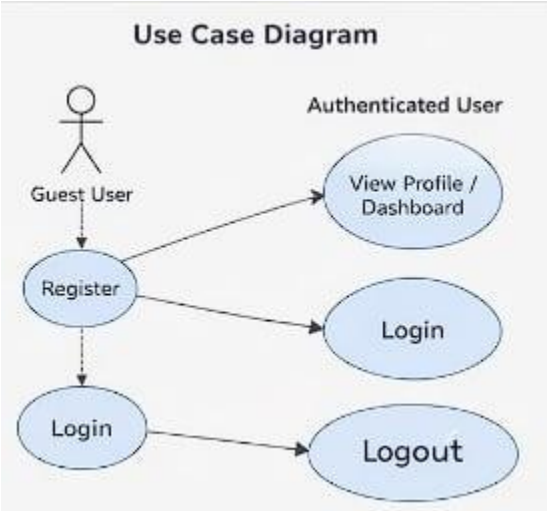
5. System Models (Diagrams)

Insert the necessary diagrams for the system:

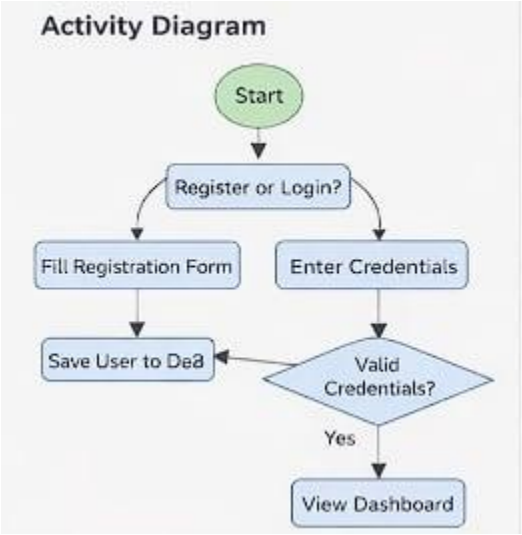
5.1. ERD



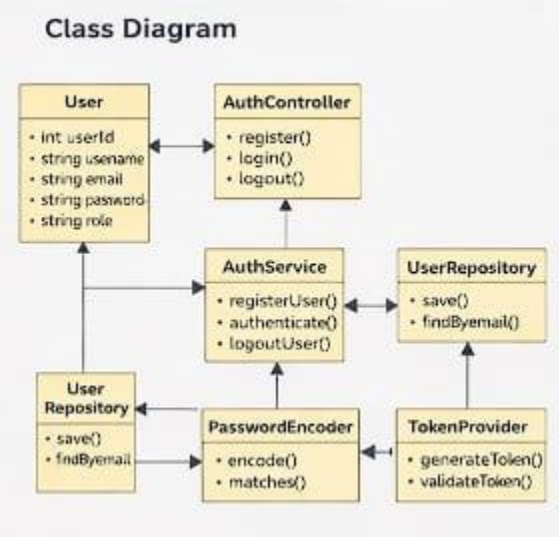
5.2. Use Case Diagram



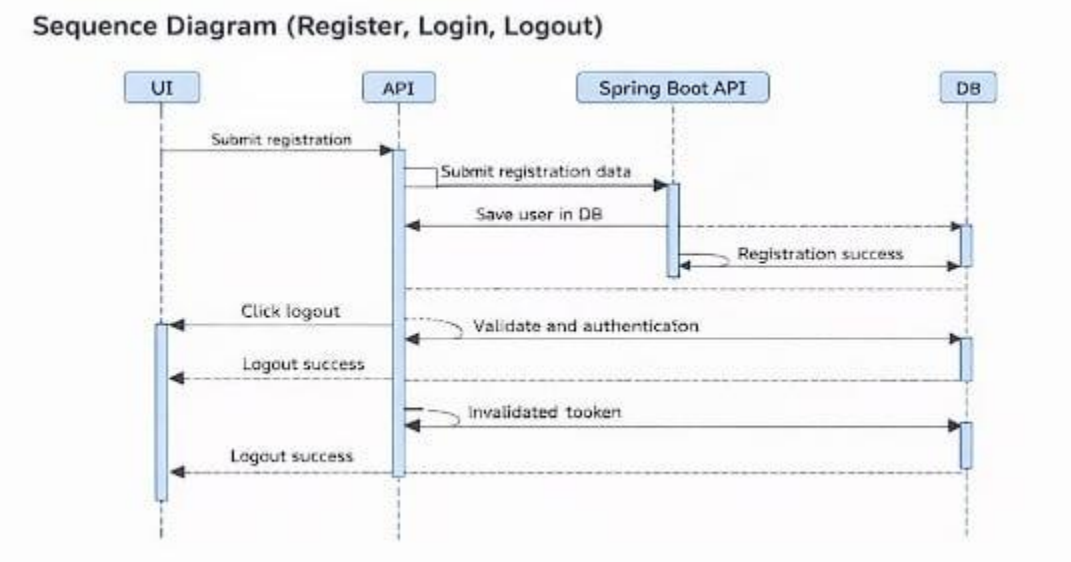
5.3. Activity Diagram



5.4. Class Diagram



5.5. Sequence Diagram



6. Appendices

This document supports the diagrams and design for the Mini App – User Registration and Authentication system. All diagrams were created using draw.io/diagrams.net as required.