IT342-Section SYSTEMS INTEGRATION AND ARCHITECTURE 1

FUNCTIONAL REQUIREMENTS SPECIFICATION (FRS)

Project Title: Mini App – User Registration & Authentication

Prepared By: Jefferus J. Resuera Jr.

Date of Submission: 02/02/2026

Version: 2

Table of Contents

1. Introduction

1.1. Purpose

The purpose of this document is to define the requirements for a User Authentication System that supports user registration, login, logout, and profile view. The intended audience for this document are instructors, students, developers, and other stakeholders who need a clear understanding of the system's functionality and limitations.

1.2. Scope

The User Authentication System provides basic access control for a web application. It allows users to create an account, authentication , log out , and view their personal profile.

1.3. Definitions, Acronyms, and Abbreviations

User - an individual who interacts with the system.
Guest User - A user who is not logged in.
Authenticated User - A user who can successfully log in.
API - Application Programming Interface

2. Overall Description

2.1. System Perspective

The system is a web-based application composed of a frontend and a backend. The frontend is developed using React and communicates with a Spring Boot backend through RESTful APIs. The backend manages authentication logic and communicates with a relational database to store and retrieve user information.

2.2. User Classes and Characteristics

- **Guest Users**
  - Can register and log in
  - Cannot access protected system features.
- **Authenticated users**
  - Can register and log in
  - Can view their profile
  - Can log out of the system

2.3. Operating Environment

Client Devices: Desktop, laptop, or mobile device
Web Browser: Chrome, Firefox, Edge, or similar
Frontend Technology: React UI
Backend Technology: Spring Boot API
Database: MySQL

2.4. Assumptions and Dependencies
- Users have a stable internet connection
- The backend server and database are operational
- The system depends on external libraries for encryption and security
- User credentials are stored securely using password hashing

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 a guest user to create a new account by providing required registration information.
Functional Requirements:
- The system shall allow users to submit registration details
- The system shall validate the entered information
- The system shall store user data securely in the database

3.2. Feature 2: User Log in
Description: This feature allows registered users to authenticate themselves and gain access to protected features.
Functional Requirements:
- The system shall accept user login credentials
- The system shall verify credentials against stored records
- The system shall grant access only to authenticated users

3.3. Feature 3: User Log out
Description: This feature allows authenticated users to securely log out of the system.
Functional Requirements:
- The system shall allow users to log out at any time
- The system shall invalidate the active session or token
- The system shall redirect the user to the login page

4. Non-Functional Requirements
Performance: The system shall respond to user requests within an acceptable time

Security: User passwords shall be encrypted and sessions protected

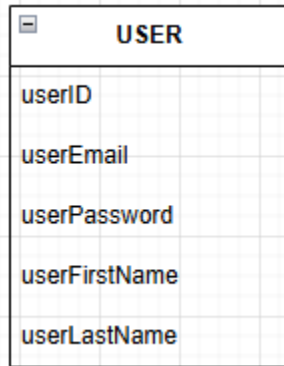**Usability**: The system shall be easy to use and understand

Reliability: The system shall function consistently without failure

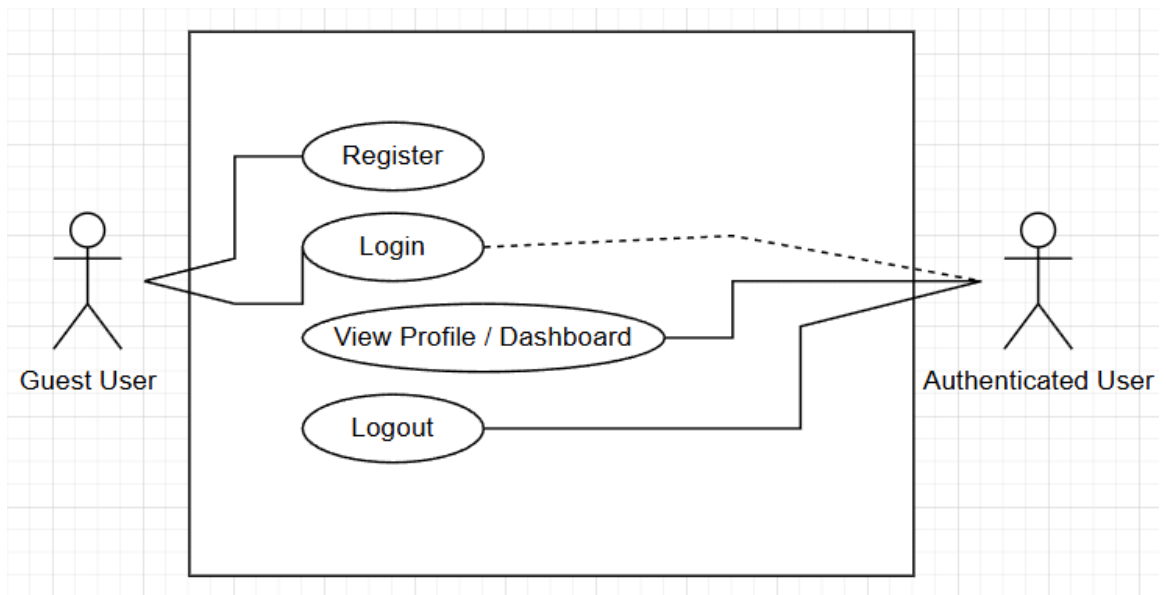**Maintainability**: The system shall be easy to update and extend

5. System Models (Diagrams)
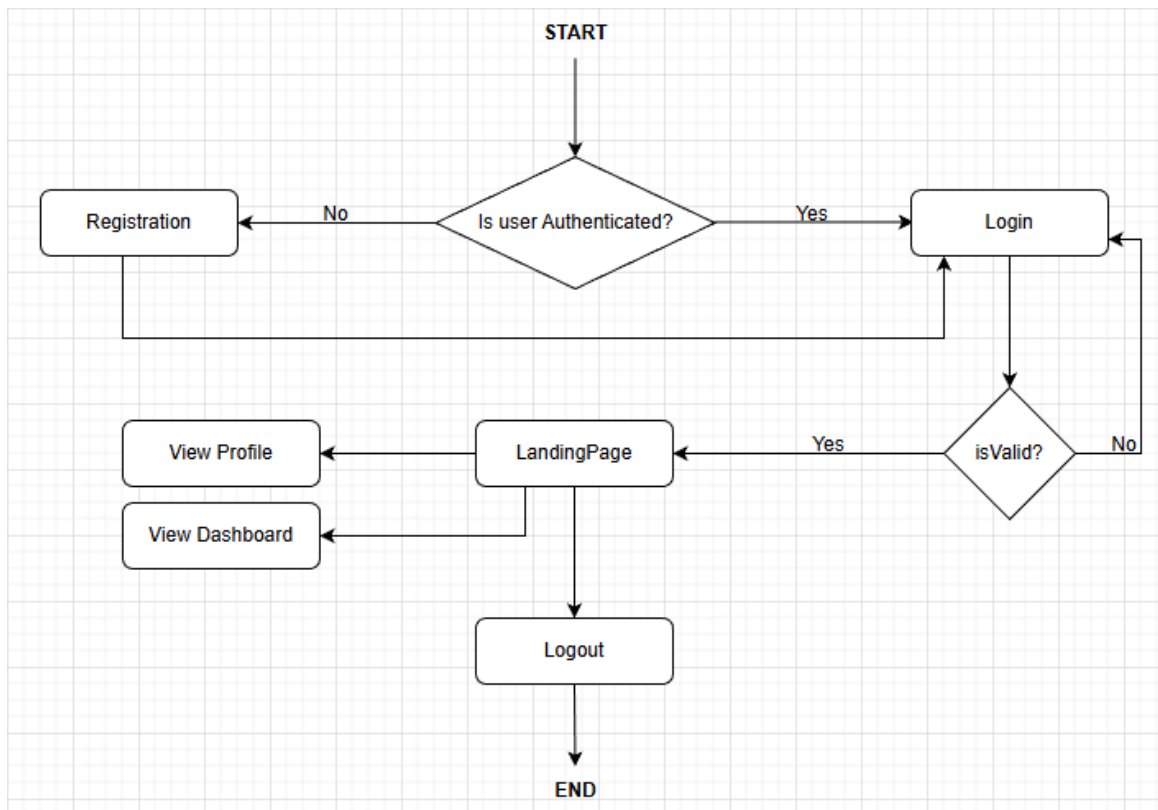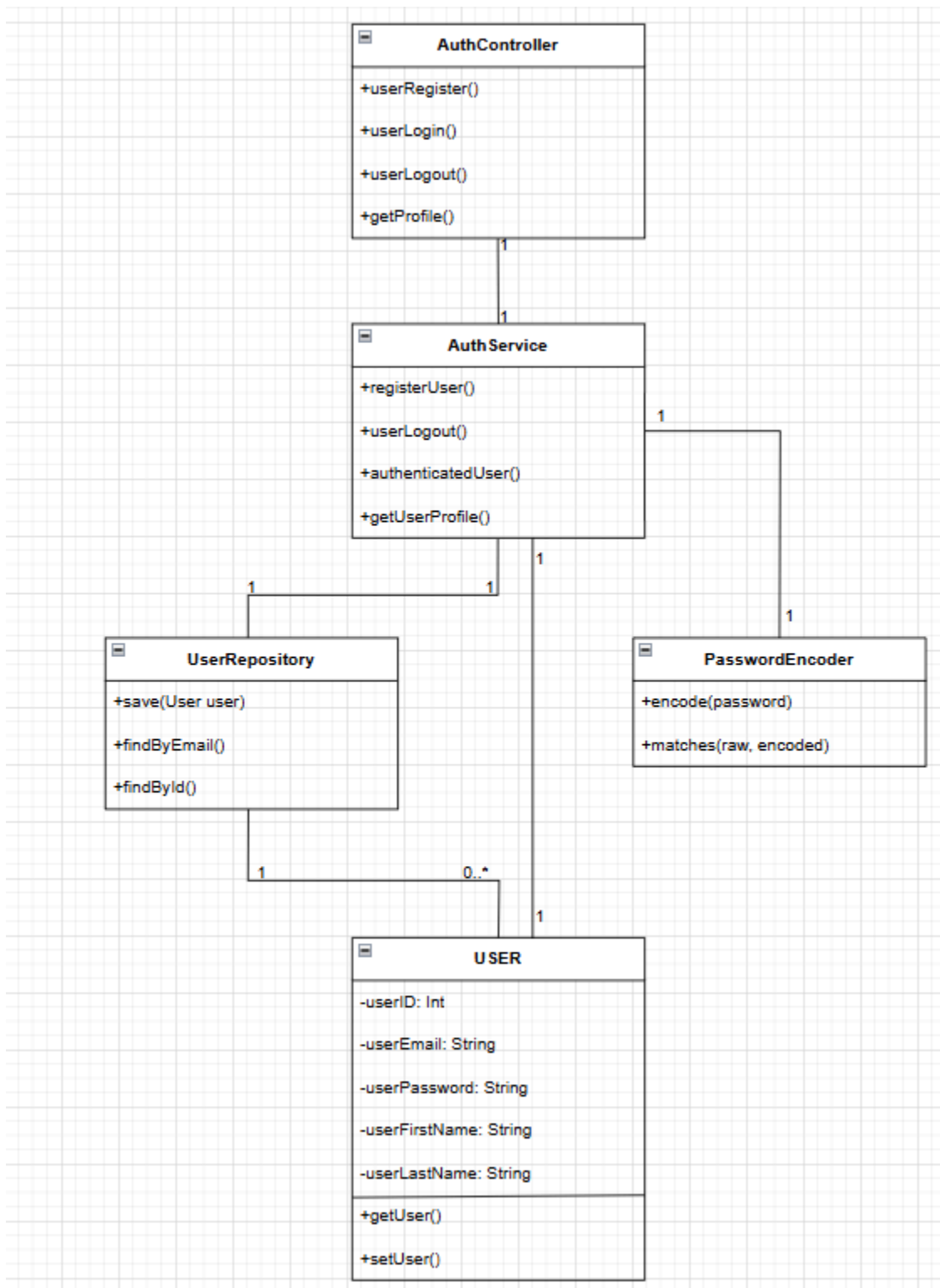
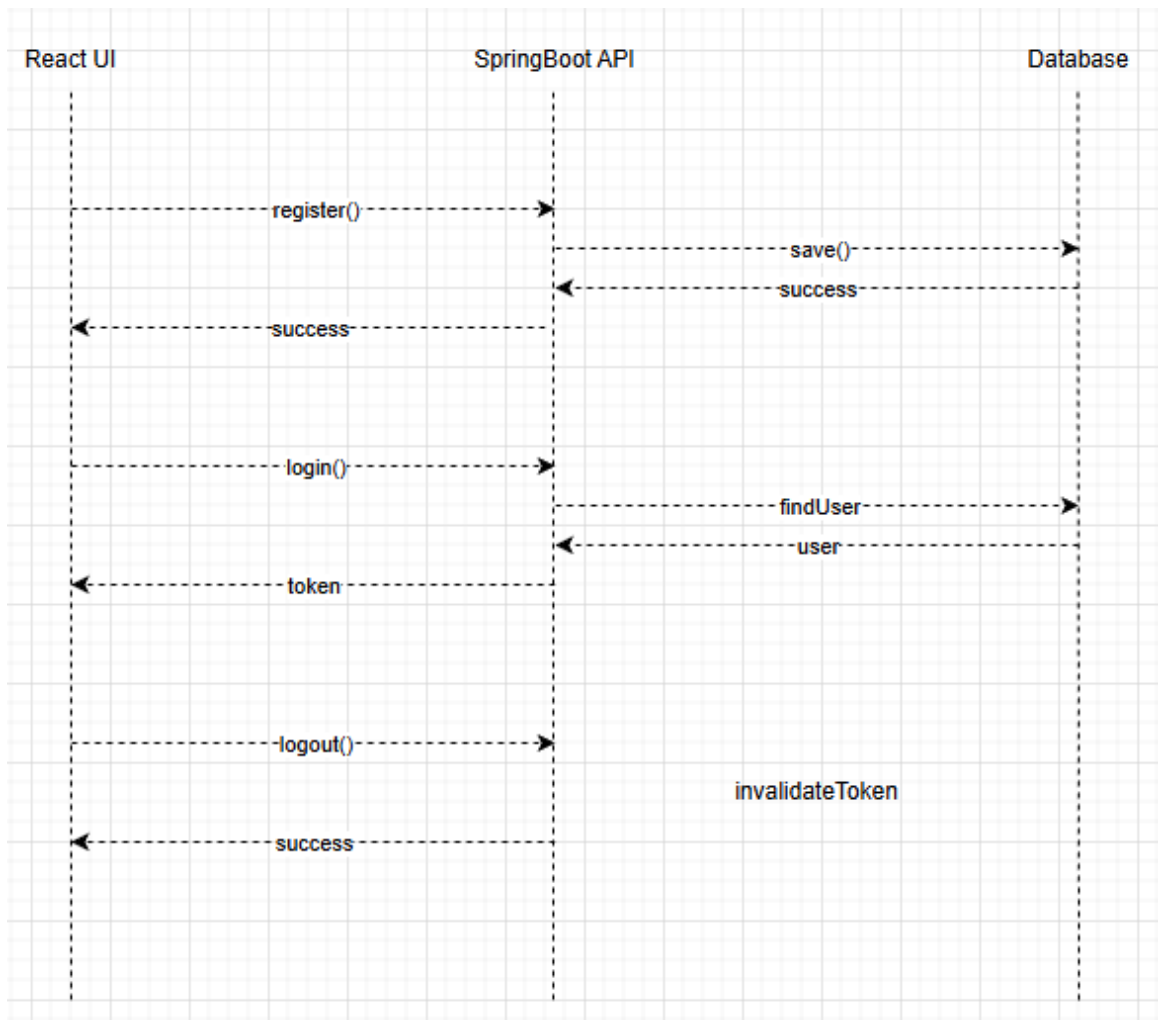*Insert the necessary diagrams for the system:*

5.1. ERD

USER
userID
userEmail
userPassword
userFirstName
userLastName

5.2. Use Case Diagram

Register

Login

View Profile / Dashboard

Logout

Guest User

Authenticated User

5.3. Activity Diagram

5.4. Class Diagram

AuthController

+userRegister()

+userLogin()

+userLogout()

+getProfile()

Auth Service

+registerUser()

+userLogout()

+authenticatedUser()

+getUserProfile()

UserRepository

+save(User user)

+findByEmail()

+findById()

PasswordEncoder

+encode(password)

+matches(raw, encoded)

USER

-userID: Int

-userEmail: String

-userPassword: String

-userFirstName: String

-userLastName: String

+getUser()

+setUser()

5.5. Sequence Diagram


```
   React UI                    SpringBoot API                    Database

      |                             |                               |
      |--------register()---------->|                               |
      |                             |----------save()-------------->|
      |                             |<---------success--------------|
      |<-------success--------------|                               |
      |                             |                               |
      |                             |                               |
      |---------login()------------>|                               |
      |                             |----------findUser------------>|
      |                             |<----------user----------------|
      |<--------token---------------|                               |
      |                             |                               |
      |                             |                               |
      |--------logout()------------>|                               |
      |                             |       invalidateToken         |
      |<-------success--------------|                               |
      |                             |                               |
```

6. Appendices

This section contains additional information and supporting materials related to the system, including diagrams, references, and other relevant documents used to support the design and development of the User Authentication System.

7. Web Screenshots
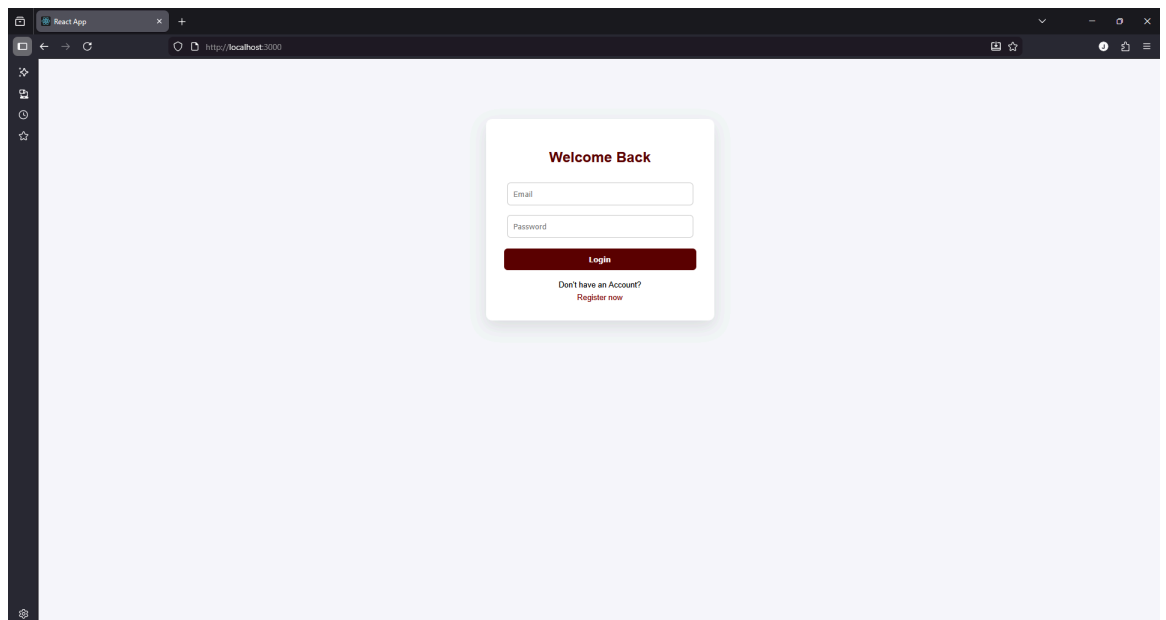
7.1. Register Page

7.2. Login Page

7.3. Dashboard Page w/ Logout Button
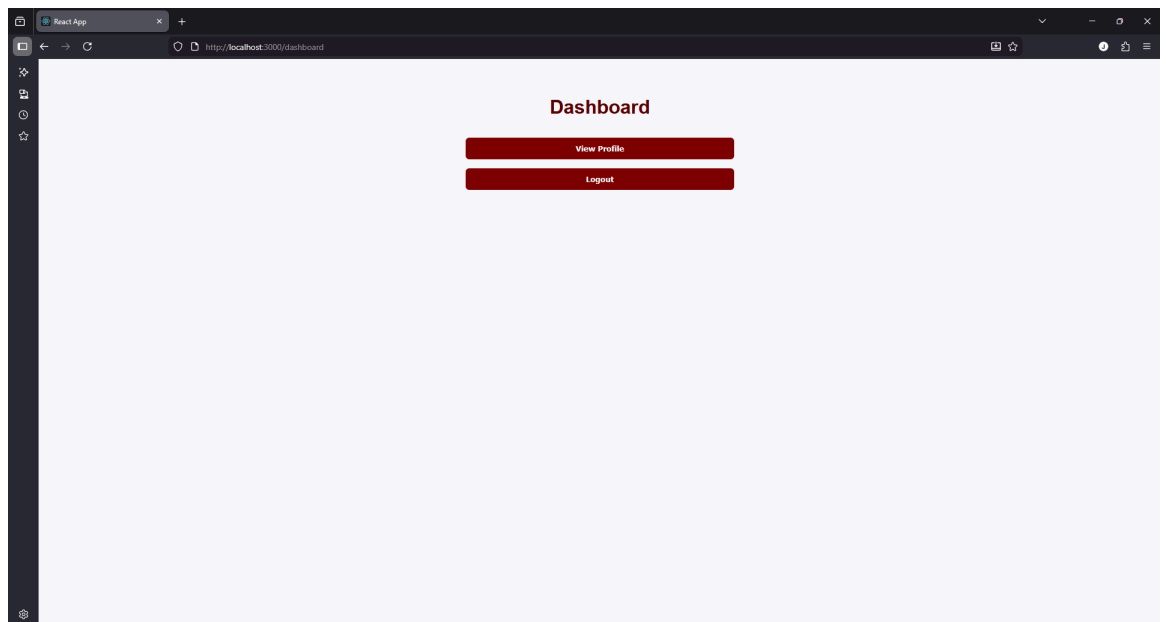
7.4. Profile Page
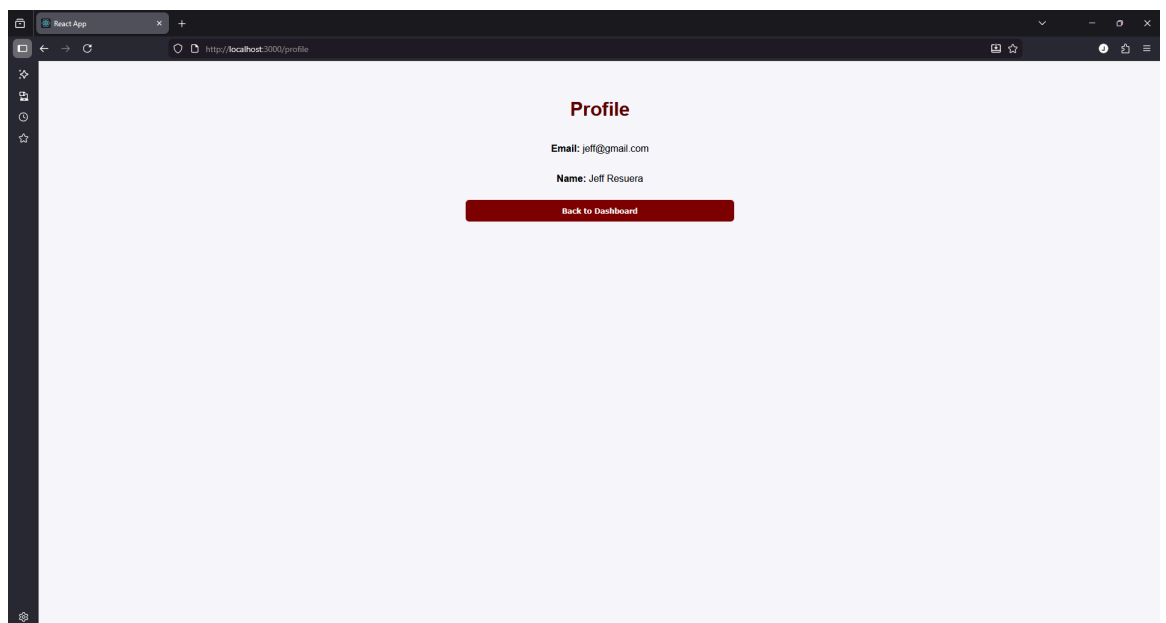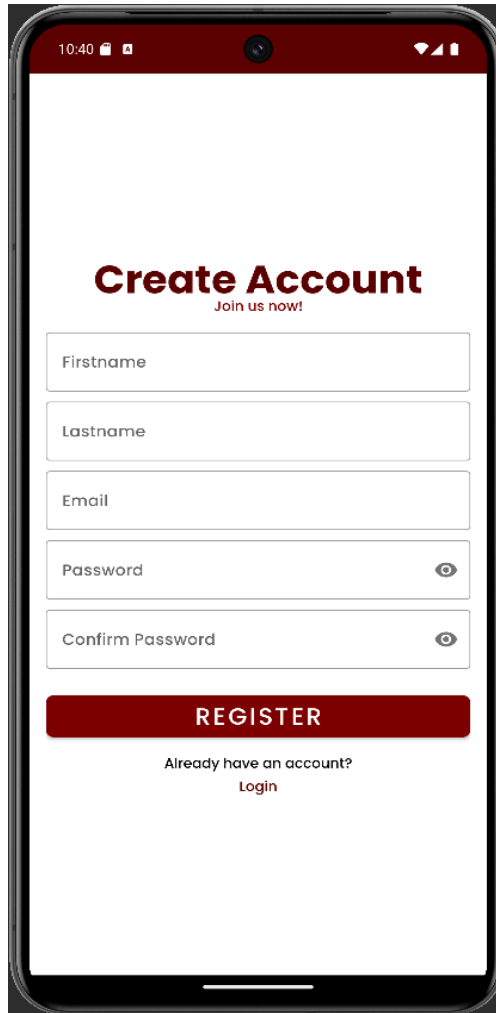
8. Mobile Screenshots

8.1. Login page
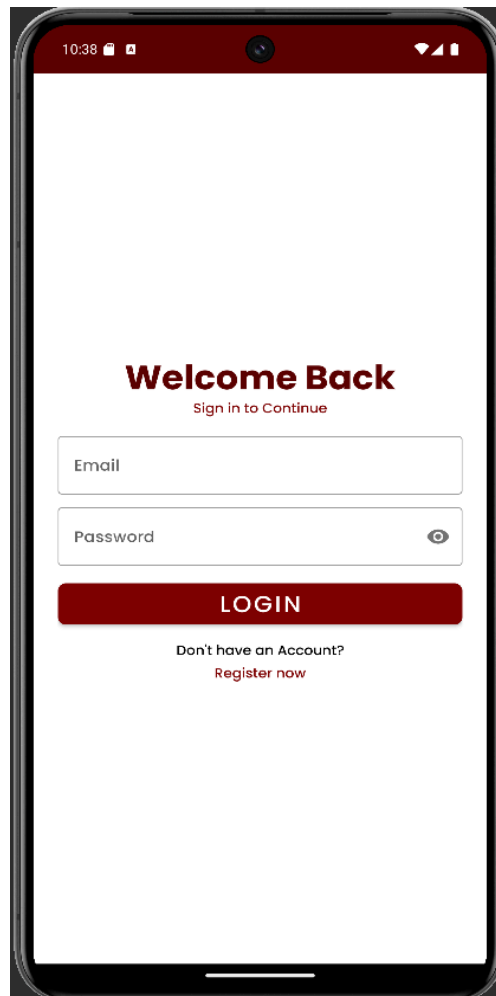
8.2. Register page

Create Account
Join us now!

Firstname

Lastname

Email

Password

Confirm Password

REGISTER

Already have an account?
Login

8.3. Dashboard Page w/ Logout & Profile Button

8.4. Profile page