Project Proposal: Mobile Application for Employee Record Management for the Criminal Investigation Department

1. Project Overview

The Criminal Investigation Department (CID) requires a robust and secure mobile application for managing employee records. This application will facilitate efficient storage, retrieval, and management of employee data, ensuring that all critical information is accessible in a secure and organized manner. The proposed solution will be a user-friendly mobile application that enables CID to manage employee records effectively, while also providing functionalities like search, update, and report generation.

2. Objectives

- A secure mobile application that stores employee records, including personal details, employment history, ranks, and departmental assignments.

- It provides functionalities for adding, updating, and deleting employee records as needed.

- A search feature that allows users to quickly find specific employee records based on various criteria (e.g., name, rank, department).

- The application employs robust security measures to protect sensitive employee data.

- A role-based access control to ensure that only authorized personnel can access or modify employee records.

1.4 Target Audience

The target audience includes administrators and human resource personnel within the Criminal Investigation Department.

1.5 Technologies Used

- **Frontend:** React Native, JavaScript

- **Backend:** C#, ASP.NET Core

- **Database:** SQL Server

- **Authentication:** Fingerprint biometric for secure user authentication

- **Data Storage:** Encrypted storage for sensitive data

2. System Design

2.1 Architecture

The system follows a client-server architecture with the React Native mobile application serving as the client and ASP.NET Core providing the server-side logic. The database is hosted on SQL Server with secured access.

2.2 Data Flow

The data flow involve input from administrators, processing on the server-side to store the data in the database. The mobile client will retrieve and display information as needed.

2.3 Security Measures

- **Encryption:** All sensitive data is encrypted both at rest and in transit.

- **Authentication:** Users is authenticated using mobile fingerprint biometric.

- **Audit Trails:** Comprehensive logging of all actions for audit purposes.

3. Development

3.1 Setup

The development environment is setup with React Native for the client and ASP.NET Core for the backend. Initialize a new project and configure the SQL Server database.

3.2 Components

- **Employee Management:** CRUD operations for employee records.

- **User Authentication:** Biometric secure login and session management.

3.3 State Management

State management will be handled using Redux for predictable state updates in the React Native application.

3.4 Data Storage

Utilize SQL Server for storing all related data, ensuring data integrity and security.
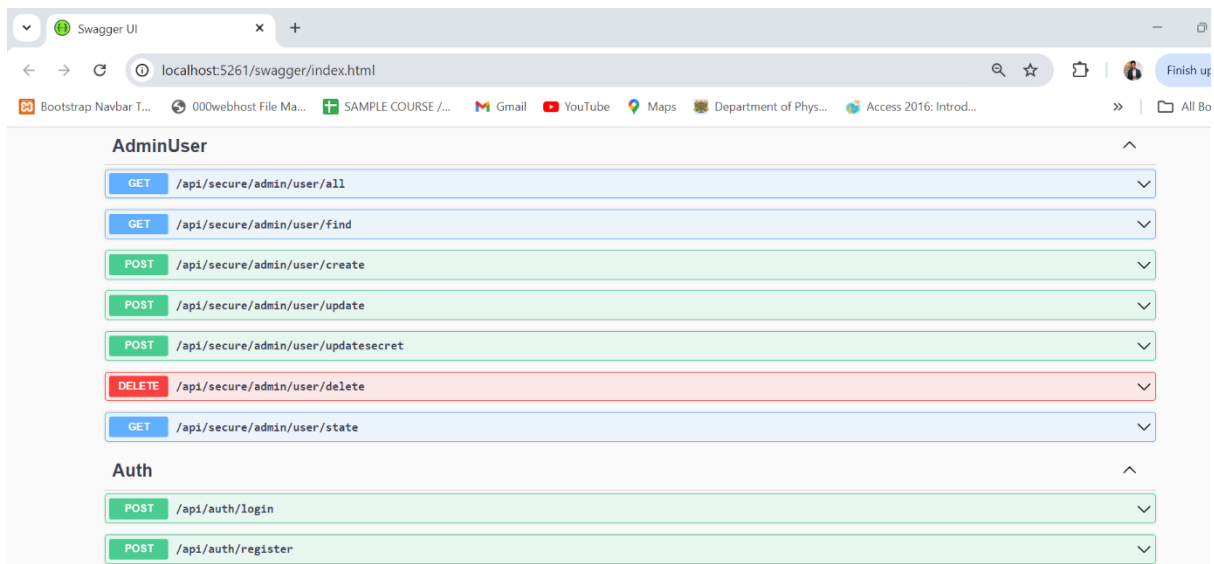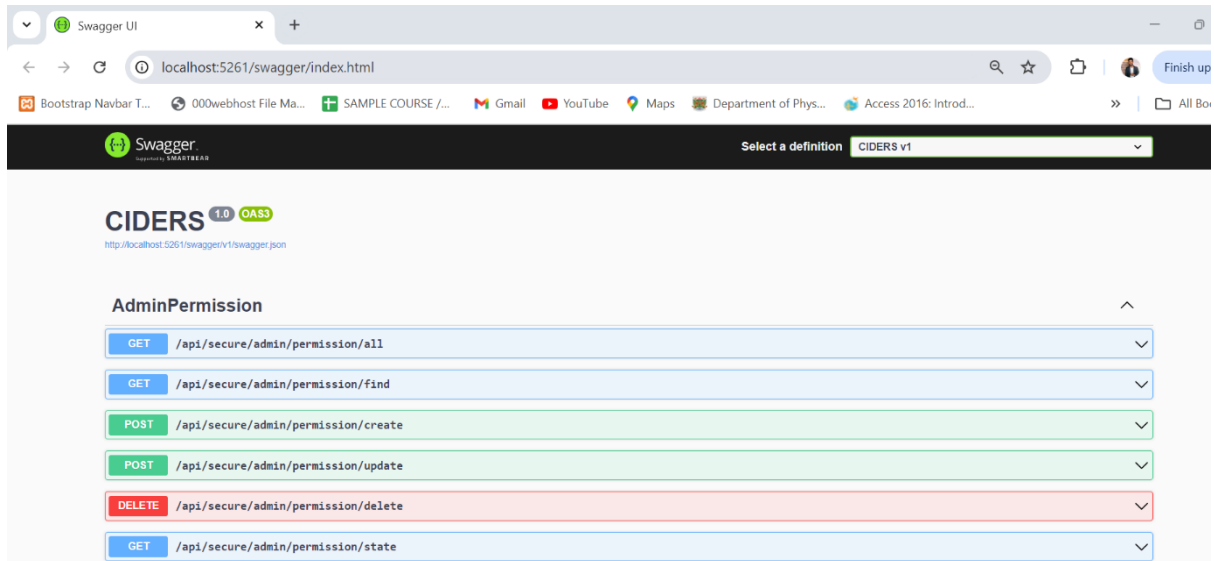
3.7 Error Handling

Implement a comprehensive error handling on both the client and server sides, including validation of user inputs

4. Low Fidelity Wireframes

We created a low fidelity design with Figma. Below is the link to access the design:

https://www.figma.com/design/oawnkXka1CEhvA8Sp1RgEF/Untitled?node-id=2-372&t=VwnRfbIbrRUU2Q8k-1

5. Screenshot of the web API End Point

Department   ⌃

GET	/api/secure/admin/dept/all	⌄
GET	/api/secure/admin/dept/find	⌄
POST	/api/secure/admin/dept/create	⌄
POST	/api/secure/admin/dept/update	⌄
DELETE	/api/secure/admin/dept/delete	⌄
GET	/api/secure/admin/dept/state	⌄

Employee   ⌃

GET	/api/secure/admin/emp/all	⌄
GET	/api/secure/admin/emp/find	⌄
POST	/api/secure/admin/emp/create	⌄
POST	/api/secure/admin/emp/update	⌄
DELETE	/api/secure/admin/emp/delete	⌄
GET	/api/secure/admin/emp/state	⌄

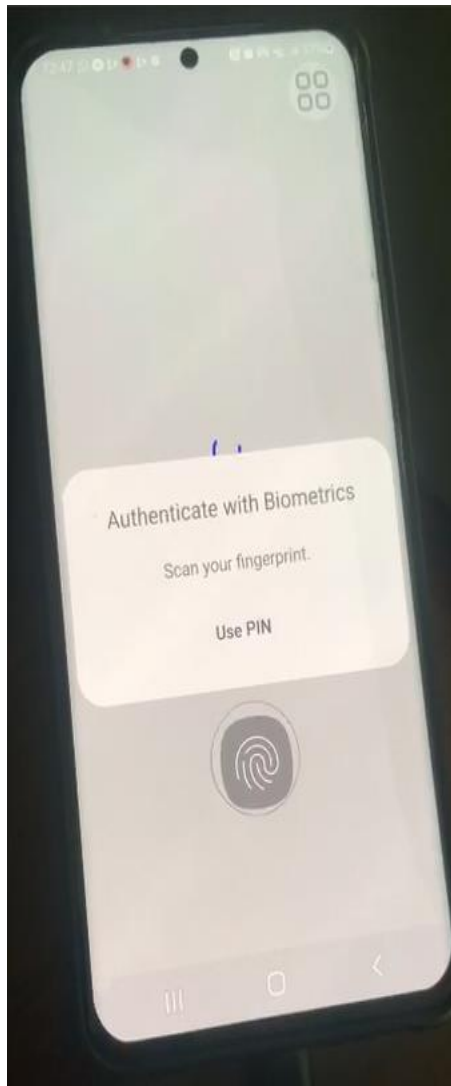## Location   ⌃

GET	/api/secure/admin/location/all	⌄
GET	/api/secure/admin/location/find	⌄
POST	/api/secure/admin/location/create	⌄
POST	/api/secure/admin/location/update	⌄
DELETE	/api/secure/admin/location/delete	⌄
GET	/api/secure/admin/location/state	⌄

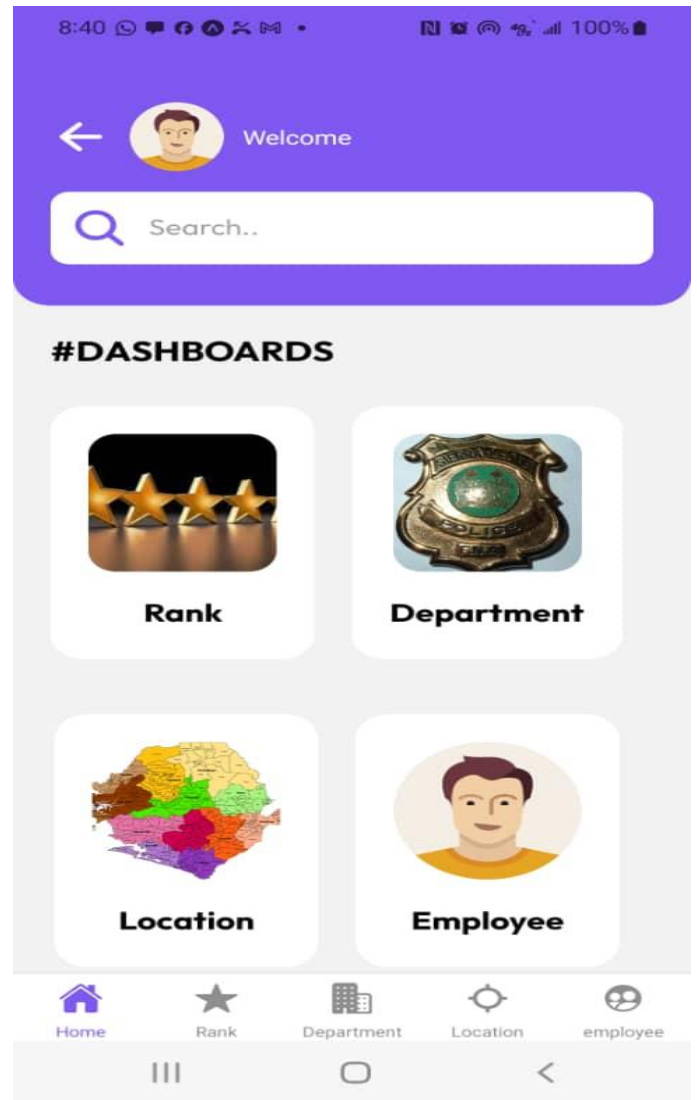## Rank   ⌃

GET	/api/secure/admin/rank/all	⌄
GET	/api/secure/admin/rank/find	⌄
POST	/api/secure/admin/rank/create	⌄
POST	/api/secure/admin/rank/update	⌄
DELETE	/api/secure/admin/rank/delete	⌄
GET	/api/secure/admin/rank/state	⌄

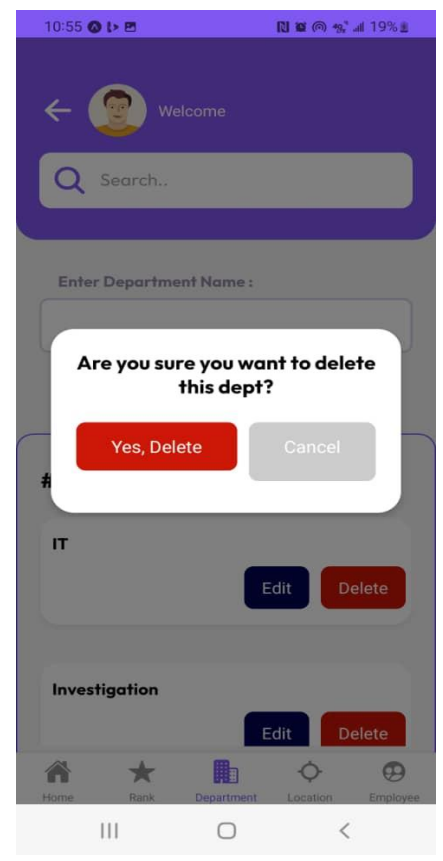## 6. Screenshot of the React Native UI
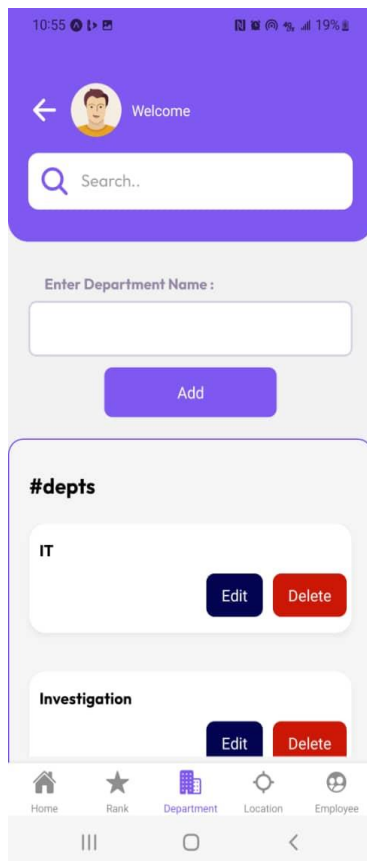
Login   **Dashboard**

Rank
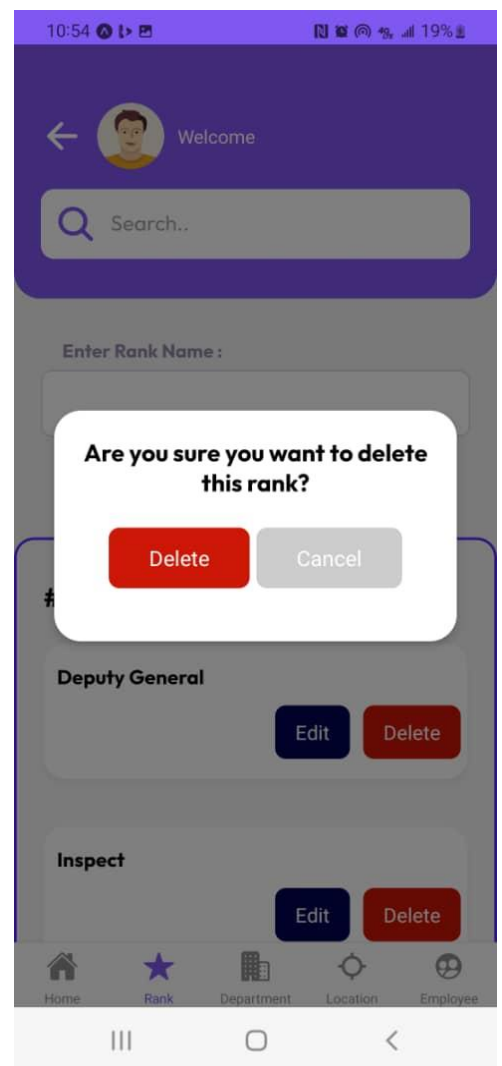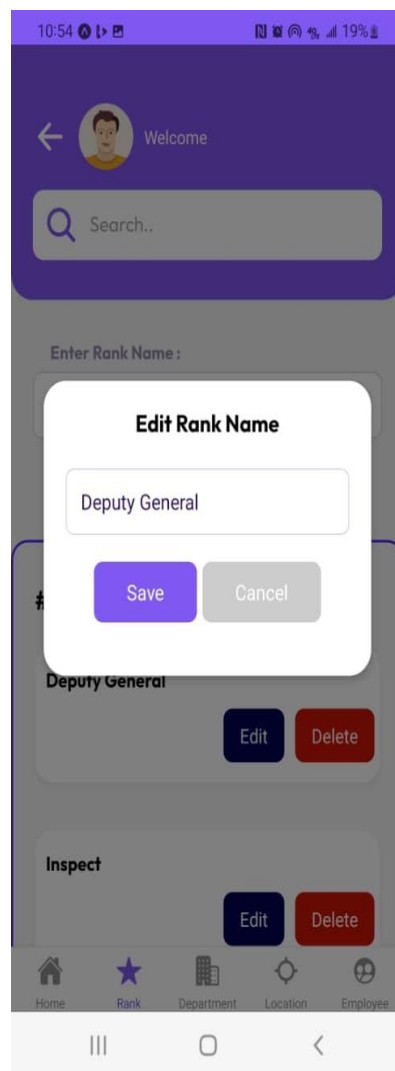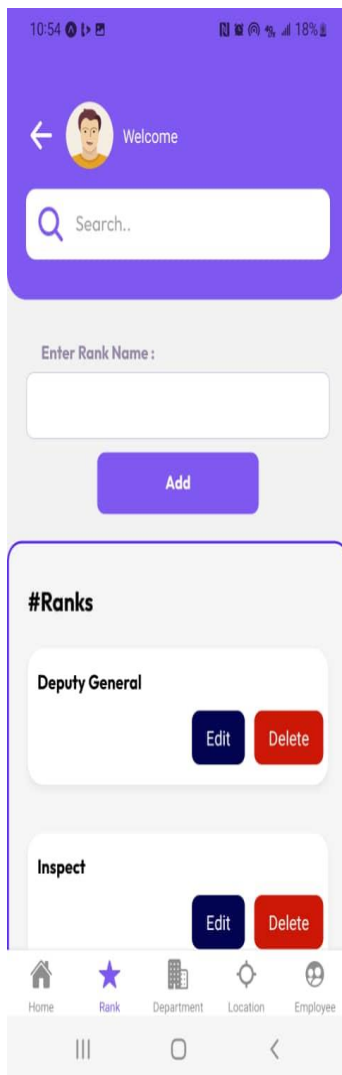
Department
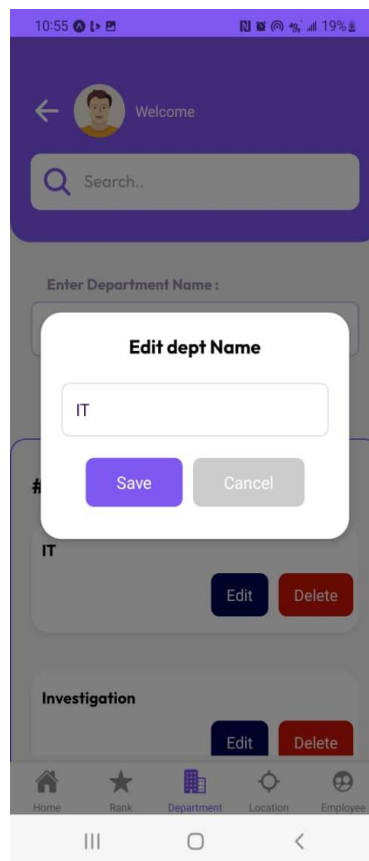
District Where Employee are Deployed

Employee

Screen 1
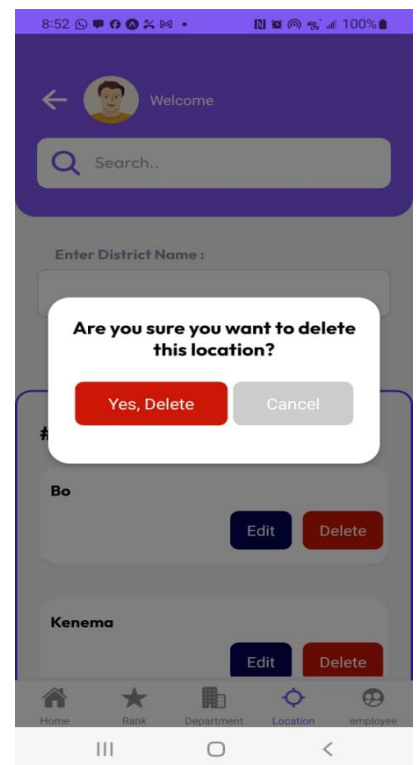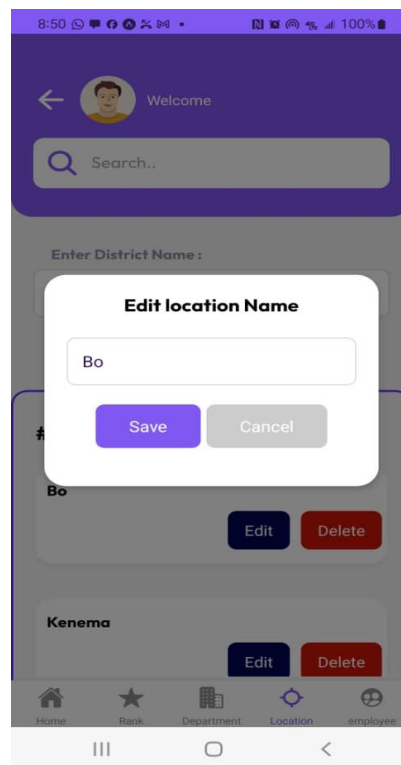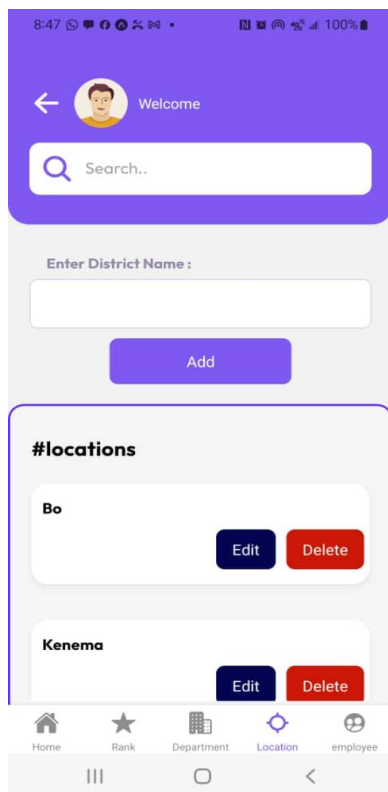
10:56   19% 🔋

← 👤 Welcome

🔍 Search..

Step 1: Basic Information

Force Number:
CID6823

Pin Number:

NEXT

View Employees

🏠 Home   ⭐ Rank   🏢 Department   ◇ Location   😊 Employee

Screen 2

10:15   100% 🔋

← 👤 Welcome

🔍 Search..

Step 2: Personal Details

First Name:

Last Name:

Middle Name:

DOB:
Date of Birth

PREVIOUS   **NEXT**

View Employees

🏠 Home   ⭐ Rank   🏢 Department   ◇ Location   😊 Employee

Screen 3

10:15   100% 🔋

← 👤 Welcome

🔍 Search..

Step 3: Contact Information

Enter Email:

Enter Mobile:

PREVIOUS   **NEXT**

View Employees

🏠 Home   ⭐ Rank   🏢 Department   ◇ Location   😊 Employee

Screen 4

10:57   19% 🔋

← 👤 Welcome

🔍 Search..

Step 4: Date Information

Employed Date:
Employed Date

Departure Date:
Departure Date

PREVIOUS   **NEXT**

View Employees

🏠 Home   ⭐ Rank   🏢 Department   ◇ Location   😊 Employee

Screen 5

10:58   19% 🔋

← 👤 Welcome

🔍 Search..

Step 5: Employment Information

Rank:
Deputy General ▼

Department:
IT ▼

District Deployed:
Bo ▼

PREVIOUS   **NEXT**

View Employees

🏠 Home   ⭐ Rank   🏢 Department   ◇ Location   😊 Employee

Screen 6

10:16   100% 🔋

← 👤 Welcome

🔍 Search..

Step 6:

Departure Reason

Profile Picture : Enter image url

PREVIOUS

Add

View Employees

🏠 Home   ⭐ Rank   🏢 Department   ◇ Location   😊 Employee
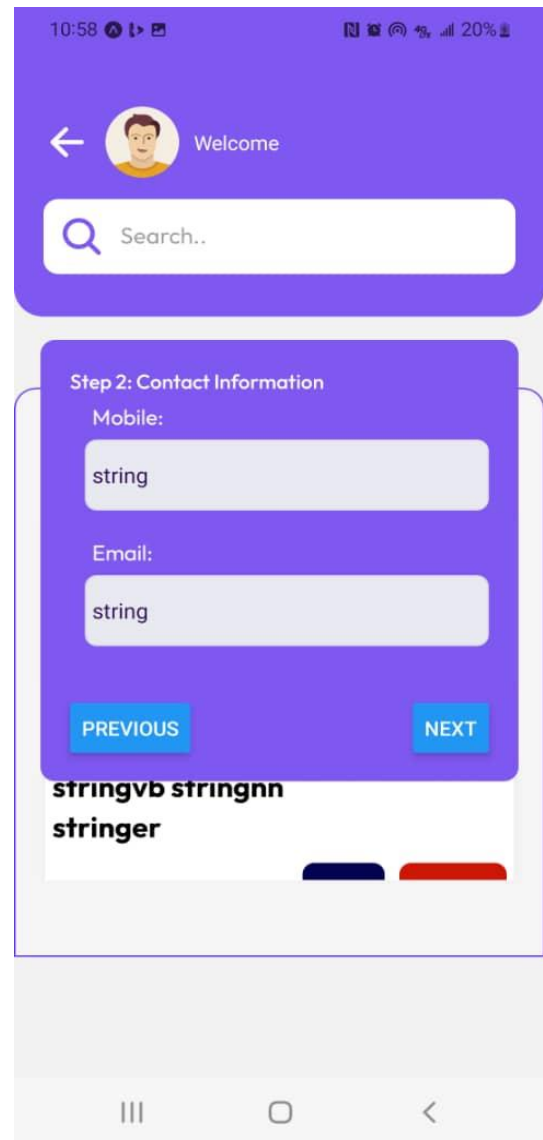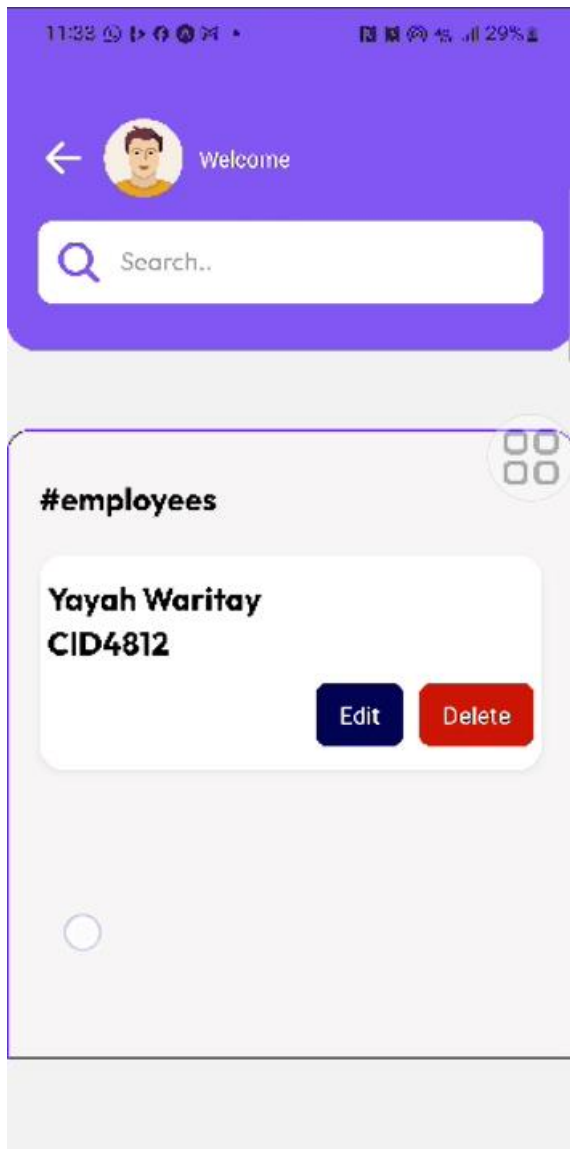
Link to the UI Project on GitHub

https://github.com/hywaritay/CIDEmp.git

Link to the web API Project on GitHub

https://github.com/hywaritay/CIDERS.git