# Full-Stack User Management Application

## Overview

This project is a full-stack web application built with Angular (frontend) and .NET (backend), connected to a MySQL database. The application is designed to manage user data with a single-page interface that contains three tabs:

- Add New User: A form to input and submit user details to the backend, where they are stored in the MySQL database.
- View All Users: A table displaying all user data fetched from the backend.
- Search, Update, and Delete Users: A search functionality that allows users to find, update, or delete existing user records.

# **Key Features**

- Single Page Application (SPA): The frontend is developed using Angular with smooth navigation between tabs without reloading the page.
- Backend with .NET: The backend provides RESTful API endpoints for creating, retrieving, updating, and deleting user data.
- MySQL Database: All user data is stored and managed in a MySQL database.
- Material UI: The design is clean and responsive, using Material UI components to ensure a user-friendly interface.

#### Tabs Overview

#### 1. Add New User

A form with fields: Name, Date of Birth, and Address. Includes a submit button to send data to the backend API for saving in the database.

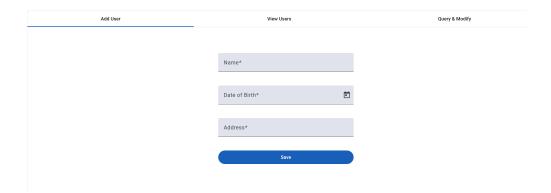


Figure 1: Add User Tab

## 2. View All Users

A table that lists all users retrieved from the backend database. Each row displays the user's name, date of birth, and address.



Figure 2: View Users Tab

## 3. Search, Update, and Delete Users

The third tab contains search functionality as well as the ability to update or delete users. Below are screenshots for different search criteria and the editing feature.

#### Search by Name Only

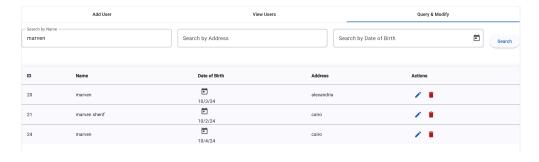


Figure 3: Search by Name Only

## Search by Address Only

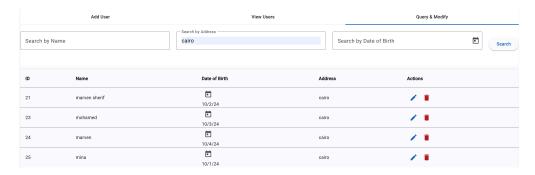


Figure 4: Search by Address Only

#### Search by Name and Address



Figure 5: Search by Name and Address

## Editing a User



Figure 6: Editing a User

# Architecture

- Frontend: Angular with Material UI for design and user interaction.
- Backend: .NET with a RESTful API serving user data.
- Database: MySQL for storing and managing user records.
- Communication: The frontend communicates with the backend API through HTTP requests to create, update, delete, or fetch user data.

# Conclusion

This project provides a robust, full-stack solution for user management with a clean user interface. The application can easily be extended with additional features like authentication, validation, or enhanced user filtering.