User Management System (Multi-Tier Architecture)

We will build a User Management System with the following core features:

- ✓ User login
- ✓ User dashboard to list all users.
- ✓ User Add/Edit/Delete operations

This solution will be multi-tier architecture

Tier 1: Frontend (Angular)

- ✓ Developed using Angular
- ✓ Key pages:
  - ▪ Login Page
  - ▪ User Dashboard With user’s list and Add, Edit and Delete button
  - ▪ Add/Edit page
- ✓ Authenticated via UserId + Password
- ✓ Connects to backend via HTTP services
- ✓ Dockerized for containerized deployment
- ✓ Will be accessible on http://localhost:8080

Tier 2: Backend API (Node.js + Express)

- ✓ RESTful API built using Node.js and Express
- ✓ Validate user credentials for login and business logic
- ✓ Routes for:
  - ▪ POST /login - user authentication
  - ▪ POST /add - create user
  - ▪ GET /users - list all users
  - ▪ GET /users:{id} – fetch user by id
  - ▪ DELETE /delete/:id - delete user by id
- ✓ Dockerized for containerized deployment
- ✓ Will be expose on http://localhost:3000

Tier 3: Database (MongoDB)

- ✓ Stores user information and roles
- ✓ Sample schema:
- ✓ User: { userid, password (hashed), userType, isActive, id }
- ✓ Dockerized MongoDB container for containerized deployment
- ✓ Will be expose on http://localhost:27017

Features

Login/Auth flow using JWT

User CRUD from UI with REST API integration

Docker Compose setup for full application configuration and run container.