Maharashtra Education Society’s

Institute of Management and Career Courses (IMCC), Pune

AUTONOMOUS

(A.Y. 2024-25)

F.Y. MCA Semester II Course-ITC 21-Mini Project

**1. Project Title and Scope**

**1.1. Project Title: Interview Preparation Guide (MERN Stack)**

**1.2. Project Scope:**

This project aims to create a **dynamic, role-based interview preparation platform** that provides users with structured learning paths, quizzes, mock interviews, and progress tracking. It will help students and professionals systematically prepare for job interviews in various technology roles.

**2. User Requirements:**

**2.1. Students/Users**:

* Sign up and log in to access personalized content.
* Select a career roadmap (Frontend, Backend, Full Stack, etc.).
* Access topic-wise interview questions and aptitude practice.
* Take quizzes and mock interviews.
* Track learning progress and completed tasks.

**2.2. Admins:**

* + Manage roadmaps (add, update, delete steps).
  + Add topic-wise questions (Technical, Aptitude, HR) to the Question Bank.
  + Add HR-based interview questions .

**3. Module Specifications:**

**3.1. User Authentication Module**

* Signup/Login (JWT authentication).
* Role-based access (Admin, Student).

**3.2. Roadmap Module**

* learning paths based on technology roles.

**3.3. Quizzes & Assessments**

* Organized topic-wise (DSA, CS Fundamentals, etc.).
* Multiple-choice quizzes with instant scoring.
* Performance tracking for users.

**3**.**4. Aptitude Module:**

* Aptitude practice for users
* Questions integrated within the Question Bank Collection.
* Solutions and explanations are shown.

**3.5. Mock Interview Module**

* HR-based interview question bank.

**3.6. Progress Tracking Module**

* User dashboard to track roadmap
* Track quiz progress and to review quiz attempted.
* To review mock interview given .
* To review aptitude questions solved.

**3.7. Admin Panel**

* Add/update roadmaps, question banks.

**4. Technologies to be Used:**

**Frontend:**

* **React.js** (Component-based UI).
* **CSS** (Styling).

**Backend:**

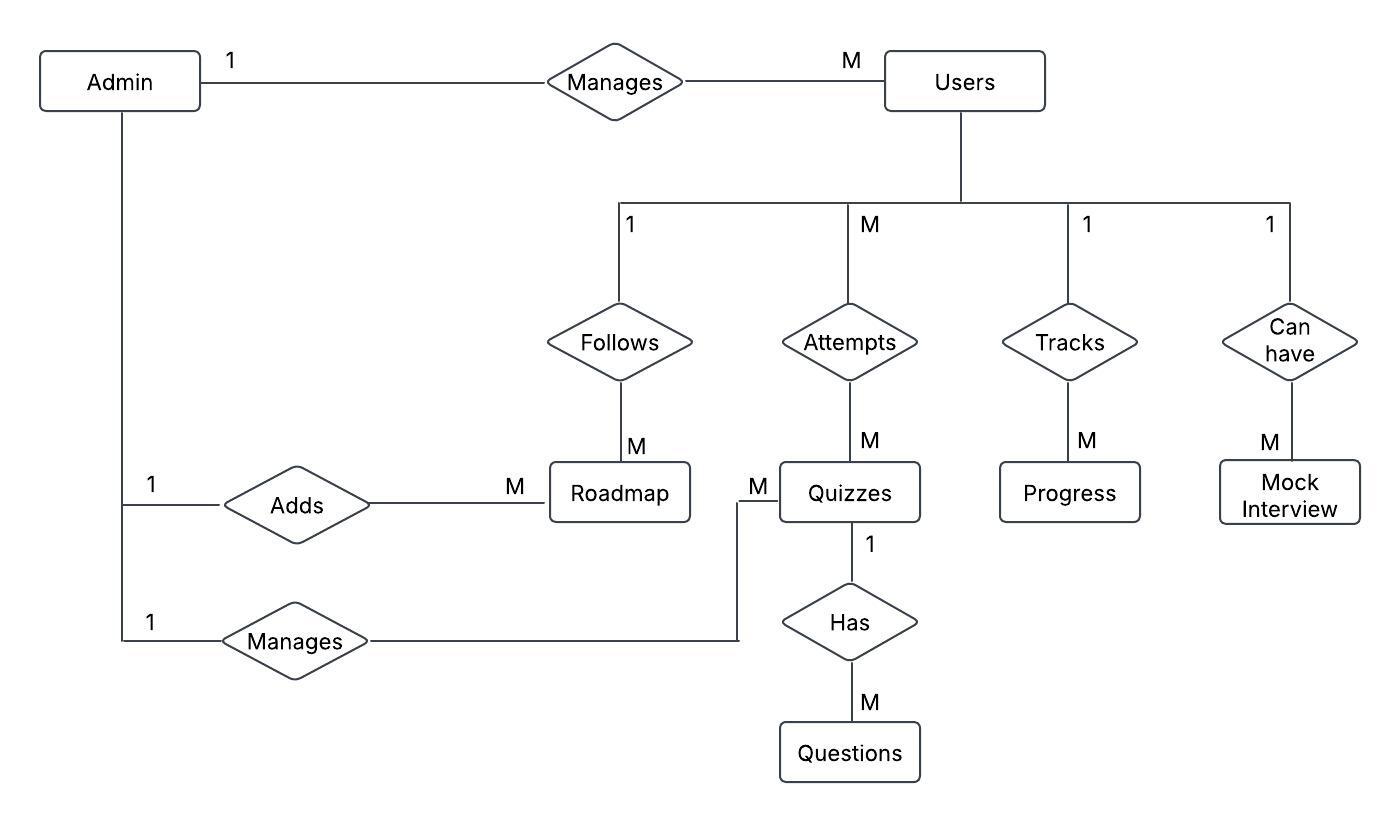
* **Node.js & Express.js** (Server-side API).
* **MongoDB (Mongoose)** (Database for dynamic content storage).5.

**Other Tools & Libraries:**

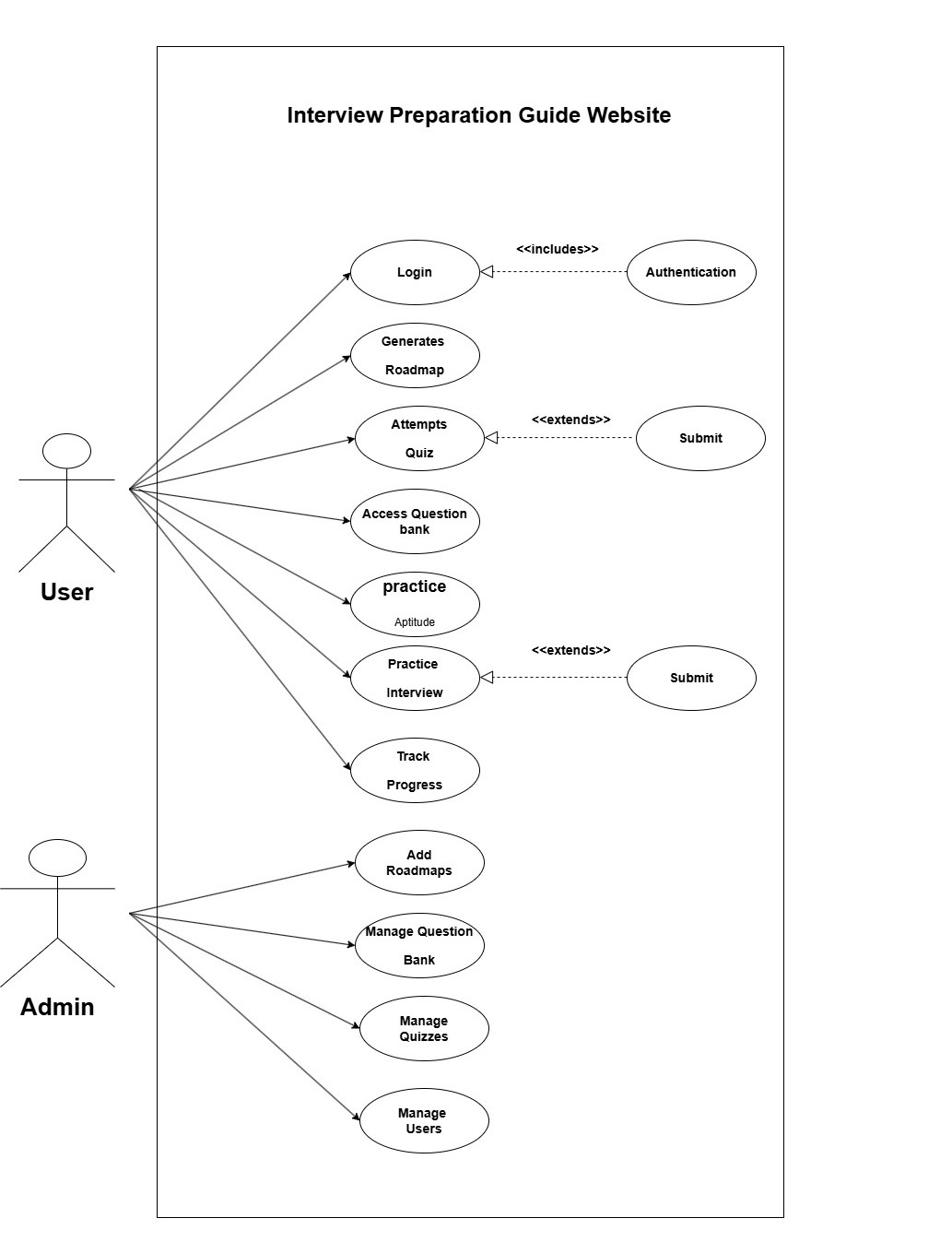
* **JWT & bcrypt.js** (Authentication & security).
* **Axios** (API calls).

**5. Diagrams**

**5.1. ER Diagram:**

****

**5.2. Use Case Diagram:**

****

**6. Data Dictionary**

### 1. ****User Collection (users)****

Stores user details and authentication data.

| **Entity** | **Attribute** | **Data Type** | **Constraints** |
| --- | --- | --- | --- |
| User | \_id | ObjectId | Primary Key, Auto-generated |
|  | name | String | Required |
|  | email | String | Unique, Required |
|  | password | String | Required, Hashed |
|  | createdAt | Date | Auto-generated Timestamp |
|  | updatedAt | Date | Auto-updated on modification |

### 2. ****Roadmap Collection (roadmaps)****

Stores different learning paths based on technology roles.

| **Entity** | **Attribute** | **Data Type** | **Constraints** |
| --- | --- | --- | --- |
| Roadmap | \_id | ObjectId | Primary Key, Auto-generated |
|  | role | String | Required (e.g., Frontend Developer) |
|  | title | String | Required |
|  | description | String | Optional |
|  | steps | Array | Ordered list of learning steps |
|  | resources | Array | Links to articles, videos, etc. |

### 3. ****Question Bank Collection (questions)****

Unified for Quizzes & Practice Problems.

| **Entity** | **Attribute** | **Data Type** | **Constraints** |
| --- | --- | --- | --- |
| Question Bank | \_id | ObjectId | Primary Key, Auto-generated |
|  | topic | String | Required (e.g., Data Structures) |
|  | category | String | Enum: Technical, Aptitude, HR |
|  | difficulty | String | Enum: Easy, Medium, Hard |
|  | questionText | String | Required |
|  | options | Array | Contains multiple-choice options |
|  | correctAnswer | String | Required |
|  | explanation | String | Optional |

### 4. ****Progress Tracking Collection (progress)****

Centralized progress tracking for users.

| **Entity** | **Attribute** | **Data Type** | **Constraints** |
| --- | --- | --- | --- |
| Progress | \_id | ObjectId | Primary Key, Auto-generated |
|  | userId | ObjectId | Foreign Key (References Users) |
|  | roadmapId | ObjectId | Foreign Key (References Roadmaps) |
|  | completedSteps | Array | List of completed roadmap steps |
|  | quizScores | Array | Stores quiz results (Topic, Score, Time Taken) |
|  | interviewResults | Array | Stores HR interview performance data |

### 5. ****Admin Collection (admins)****

| **Entity** | **Attribute** | **Data Type** | **Constraints** |
| --- | --- | --- | --- |
| Admin | \_id | ObjectId | Primary Key, Auto-generated |
|  | name | String | Required |
|  | email | String | Unique, Required |
|  | password | String | Required, Hashed |

### 6. ****Mock Interview Collection (mock\_interviews)****

Stores text-based HR questions handled by the AI chatbot.

| **Entity** | **Attribute** | **Data Type** | **Constraints** |
| --- | --- | --- | --- |
| Mock Interview | \_id | ObjectId | Primary Key, Auto-generated |
|  | category | String | Enum: HR Interview |
|  | questions | Array | List of question IDs (References Question Bank) |
|  | createdAt | Date | Auto-generated Timestamp |