**Project Documentation: Alumni Tracker System**

**1. Introduction**

The **Alumni Tracker System** is a web-based platform designed to bridge the gap between alumni and institutions. It provides an interactive interface where alumni can stay connected, share job opportunities, participate in discussions, and engage in upcoming events. The system comprises an **Admin Panel** for managing content and an **Alumni Panel** for user interactions.

**2. Objective**

The primary objective of the **Alumni Tracker System** is to create an interactive and efficient platform that enables institutions to maintain a strong connection with their alumni. It allows alumni to share job opportunities, participate in discussions, stay informed about institutional events, and contribute to the growth of the alumni network. The system also helps administrators manage alumni data, events, and forums effectively.

**3. Technologies Used**

**Front-End:**

* HTML
* CSS
* Bootstrap
* EJS (Embedded JavaScript)

**Back-End:**

* Node.js
* Express.js

**Database:**

* MongoDB (Mongoose ODM)

**4. System Requirements**

**Hardware Requirements:**

* **RAM:** 4GB
* **Storage:** 128GB / 256GB / 512GB
* **Processor:** Intel i3 / i5 / i7

**Software Requirements:**

* **Operating System:** Windows, macOS, or Linux
* **Node.js:** Version 14 or later
* **MongoDB:** Community or Atlas version
* **Package Manager:** npm or yarn
* **Version Control:** Git and GitHub
* **Development Tools:**
  + Visual Studio Code (or any preferred IDE)
  + Postman for API testing

**5. Modules and Functionalities**

**i. Admin Panel**

The Admin Panel is responsible for managing key functionalities of the system.

**Admin Functionalities:**

1. View the list of job postings.
2. Maintain the alumni database.
3. Manage and create events.
4. Oversee forum topics and also Remove Forum.
5. Manage the image gallery.

**ii. Alumni Panel**

The Alumni Panel allows users to interact with the system after registration.

**Alumni Functionalities:**

1. Register on the platform (verified by Admin).
2. Post job opportunities.
3. Assignment: View List of Posted Jobs, Update Posted Jobs and also delete posted jobs
4. Create and participate in forum discussions.
5. Join existing forum topics.
6. Commit to participating in upcoming events.

**6. Practical Implementation**

**Front-End Implementation:**

* Develop responsive web pages using HTML, CSS, and Bootstrap.
* Implement dynamic content rendering using EJS.
* Use AJAX for seamless data retrieval and updates.

**Back-End Implementation:**

* Set up an Express.js server with RESTful APIs.
* Implement authentication and role-based access control using JWT or session-based authentication.
* Use Mongoose to define database schemas and interact with MongoDB.
* Handle file uploads for the gallery using Multer.

**Database Implementation:**

* Design schemas for alumni, jobs, events, and forum discussions.
* Use indexing for optimized searches and queries.
* Implement data validation and constraints.

**Deployment and Maintenance:**

* Deploy the system on a cloud platform such as AWS, Heroku, or Render.

**7. Conclusion**

The **Alumni Tracker System** provides an efficient platform for maintaining alumni engagement and networking. With dedicated modules for job postings, events, and discussions, the system ensures smooth interaction between alumni and the institution. The structured implementation ensures reliability, scalability, and ease of use for both administrators and alumni.