CSE471: System Analysis and Design
Assignment on Functional Requirements
Proposed Project Title: Job Portal Application (MERN Stack)

Group No: 10, CSE471 Lab Section: 01, Fall 2025	
ID	**Name**
21201149	Mahmuda Akter Alif
22301769	Shuvashis Basak
22101409	Tahseen Tanha
22101385	Aneela Musarrat

Submission Date:14.11.2025

Tech Stack: MERN

- Language: JavaScript
- Framework: React.js
- Styling: TailwindCSS
- Database: MongoDB
- ORM: Prisma
- Deployment: Vercel

Functional requirements:

Module 1

1. Users can securely log in to access their features and log out, utilizing JWT authentication.Users can create an account by entering their details and specifying their role (Student/Recruiter).

2. Users can search and filter job listings (title, location, keywords) on the public Home/Browse page.

3. Implement advanced route protection and role-based access control on frontend (React Context) and backend (Express middleware) to restrict unauthorized access to dashboards. *Unique: Session timeout with auto-save drafts for unfinished actions.*]

4. Users can view personalized job recommendations on the public side based on search history or basic profile keywords (using simple collaborative filtering logic). Unique: Trend-based suggestions pulled from job market news.[api]

Module 2

1. Users can view and update their profile information, including name, email, phone number, bio, and resume/avatar upload. Recruiters can update company information (website, description, location) and change the company logo after

registration. Users can view their applied job status dashboard, showing the status (Pending/Accepted/Rejected) for each application.

2. Users can apply to a job listing via the "Apply Now" button, linking their profile to the job post.

3. Users can add a job to a "Wishlist" or "Saved Jobs" list for later review.

4. Implement a Smart Job Alert system that uses AI to send weekly email summaries of new jobs matching user preferences (via SendGrid free tier API).[api]

Module 3

1. Recruiters can post new job listings with all details (title, description, salary, requirements). Implement a Company Analytics Dashboard showing the total applicants, accepted/rejected ratio, and active job posts, visualized with Chart.js.

2. Recruiters can do a side-by-side comparison of up to 4 candidates showing skills, experience, and compatibility scores to help recruiters make decisions.

3. An AI tool that scans uploaded resumes and provides improvement suggestions (missing keywords, formatting issues, skill gaps for target jobs) with a compatibility score for specific job listings.[api]

4. Enable integrated feedback collection: after every interview or application, recruiters and applicants can rate the interaction, suggest improvements, and this feedback becomes available as anonymized insights for system improvements.

5. Allow recruiters to build a repository of frequently asked interview questions for each posted job, so each future recruiter or team member can quickly reuse or edit them for later rounds.

6. Add a login activity tracker using the IPinfo free API to detect unusual login patterns and prompt users with a security alert.[api]

7. Let users upload and manage multiple resumes or portfolio documents, assign custom titles to each, and select which one to use for each job application.Provide an "Achievements and Milestones" tracker where users can record past career accomplishments, display them visually on their profile, and update them any time.

8. Offer an in-platform note-taking feature for each job post, allowing users to jot down thoughts, reminders, or follow-up plans which are then saved to their account.(floating)