**Recruitment System Software**

**1. About the Project:**

The project is to develop a recruitment system that makes hiring easier by keeping candidate details, job postings, and interview schedules in a database. It assists HR teams in filtering applications, scheduling interviews, and sending automatic notifications, making hiring easier and minimizing manual effort.

*Example: A university may have thousands of graduating students annually. The system assists companies in finding the correct candidates by giving a structured area for job postings and student applications, removing manual steps and wastage of time.*

**2. DBMS (Database Management System):**

Software stores and keeps all the data in the recruitment system. It assists in managing:

• Candidate details (name, qualifications, experience, contact numbers)

• Job postings (job names, descriptions, salaries)

• Interview schedules (date, time, interviewers)

• Selection process details (shortlisted, rejected, hired)

**3. Software Requirements:**

The recruitment system software requires some tools and technologies to operate efficiently. They are:

1. Operating System – Windows for the time being

2. Programming Language – Python, Java, or C++ (any other too can be used according to preference here we are using C++)

3. Database – MySQL, PostgreSQL, or MongoDB

4. Web Server – Apache or Nginx (if it is a web-based system)

5. Front-end Framework – React.js or HTML/CSS for user interface

**4. Advantages of the Project**

This recruitment system will bring numerous advantages, such as:

✔ Time-Saving – Automates routine processes like sorting applications and sending messages.

✔ Efficient Hiring – Assists recruiters in finding the best candidates quickly.

✔ Better Organization – Keeps all candidate and job details in one place.

✔ Reduces Errors – Avoids errors due to manual entry of data.

✔ Improves Communication – Gives automatic notifications to candidates and recruiters.

**5. Scope:**

The project scope outlines what it can do and how it can be utilized. The system can be utilized by:

• Small and large firms to automate hiring.

• Recruitment agencies to handle multiple job openings and candidates.

• Job applicants to apply for positions and monitor application status.

• HR departments to monitor and streamline hiring.

**Note:** In the future, the software can further be enhanced to include AI-driven resume filtering, video interviewing features, and integration with LinkedIn.

**6. Functional Requirements:**

The functional requirements define what the software should do. Some of the most important functions are:

1. User Registration & Login – Job applicants and HR departments have to be able to register and log in.

2. Job Posting Management – Recruiters can add, edit, and remove job postings.

3. Candidate Application Process – Applicants can apply and monitor applications.

4. Resume Screening – The system has to screen candidates on the basis of experience and qualification.

5. Interview Scheduling – HR can schedule and monitor interview dates and times.

6. Status Updates – Candidates are notified of application status.

7. Report Generation – HR can produce reports on hiring trends and candidate performance.

**6.1. Main Features:**

**6.1.1 Job Posting**

• Recruiters can add job postings with detailed descriptions, requirements, and deadlines.

• Companies can edit, delete, and update job postings.

**6.1.2 Candidate Registration**

• Students first log in on the website

• then Students can register, create profiles, and upload resumes.

• Students can apply for multiple openings.

**6.1.3 Application Tracking**

• Recruiters can view and filter applications on the basis of qualifications.

• Candidates can monitor the status of their applications.

• Admins can monitor overall recruitment statistics.

*Example:*

*A recruiter shortlists five candidates for an interview and updates their application status to "Interview Scheduled". Students are notified of the next steps.*

**6.1.4 Communication Features:**

• Automated job update notifications

• Automated application status notifications

• Automated interview schedule notifications.

• Messaging or chat capability between recruiters and candidates.

• Email notifications for significant updates.

*Example:*

*A student gets an email notification that reads: "Your application for Data Analyst Intern at XYZ Corp has been shortlisted. Check your dashboard for the next steps."*

**6.2 User Roles:**

**6.2.1 Recruiters**

• Publish job postings.

• See candidate applications.

• Schedule and administer interviews.

*Example:*

*A Google recruiter logs in and finds 50 applications for a Data Science internship and schedules interviews for the top 10 candidates.*

**6.2.2 Candidates (Students):**

• Register and create profiles.

• Apply for job postings.

• See application status.

• Get notifications for job updates and interview schedules.

*Example:*

*A student logs in and finds that the status of their application for a Marketing Associate role is "Under Review."*

**6.2.3 Admins**

• Control user access and permissions.

• Control system security and data integrity.

• Create reports on recruitment activity.

*Example:*

*An admin runs a report that indicates 500 students have applied for jobs this semester with an average of 5 applications per student.*

**6.3 Key Non-Functional Requirements:**

• Security: Data encryption, role-based access control, and secure login.

• Usability: Easy-to-use interface and easy navigation.

• Scalability: Support for increasing number of users and job postings.

• Reliability: High availability and low downtime.

*Example:*

*A company's HR team accesses thousands of applications without performance slowdown, ensuring smooth operation even in heavy hiring periods.*

**7. dataflow diagram:**

The diagram below describes the interaction between the User Interface and the Campus Recruitment System:

• User Interface: Has three primary types of users – Admin, Student, and Company (Recruiters).

• User Login Info: Users log in with credentials.

• Success Message: On successful login, a success message is returned.

• User Raw Data: Data entered by users, such as profiles and job details, is passed to the system.

• Formatted Data: The system processes and formats the data before storing it.

• Input Message: Queries or applications are sent by the users to the system.

• Output Message: The system produces appropriate information, like job postings or application statuses.

**8. Deliverables:**

• System Requirement Specification (SRS) Document

• Detailed System Design Document

• Database Schema and ER Diagrams

• Prototype User Interface Screens

• Test Cases for System Validation

*Example:*

*A team of software developers applies the ER Diagram to design the database, thereby providing a formatted way of holding student and recruiter information. The document is a starting point to create a reliable Campus Recruitment System that improves student and employer hiring experience.*