

Job portal System



***Introduction of Job Portal System***

A **Job Portal System** is a web-based or mobile platform designed to connect job seekers with potential employers. It serves as a bridge between candidates looking for employment opportunities and organizations seeking to hire the right talent. The system provides features for job posting, resume submission, job search, applicant tracking, and communication between employers and job seekers.

The primary goal of a Job Portal System is to streamline the recruitment process, making it faster, more efficient, and accessible. By digitizing job search and hiring operations, it reduces the dependency on traditional methods like newspaper ads or recruitment agencies. Employers can post job vacancies, manage applications, and filter candidates based on qualifications and experience, while job seekers can browse openings, submit applications, and track their application status.

Modern job portals may also include advanced features such as AI-based job matching, interview scheduling, employer branding, and skill assessments. These tools enhance the user experience and improve the chances of finding the right match between job roles and can

**Purpose**

The purpose of this document is to specify the requirements for a Job Portal System, which enables job seekers to search and apply for jobs, and employers to post vacancies and manage applications.

**Scope**

The system will allow:

* Job seekers to create profiles, upload resumes, search and apply for jobs.
* Employers to post job openings, view applications, and contact candidates.
* Admins to manage users, categories, and monitor activity.



**Objectives of the Job Portal System**

The main objectives of the Job Portal System are:

1. **To bridge the gap between job seekers and employers**  
   Facilitate a platform where employers can post job vacancies and job seekers can apply for suitable jobs.
2. **To streamline the recruitment process**  
   Provide automated tools for job posting, resume screening, candidate shortlisting, interview scheduling, and communication.
3. **To provide personalized job recommendations**  
   Use user profiles, skills, and preferences to suggest relevant job openings to applicants.
4. **To manage user data efficiently and securely**  
   Ensure secure storage and retrieval of sensitive user information like resumes, personal details, and employment history.
5. **To support advanced search and filtering options**  
   Allow both employers and job seekers to find the most relevant results based on location, skills, experience, salary range, etc.
6. **To ensure a responsive and user-friendly interface**  
   Create a seamless experience across all devices (web and mobile) for users of varying technical expertise.
7. **To enable admin-level monitoring and control**  
   Provide an administrative dashboard to manage users, postings, reports, and handle disputes or misuse of the platform.
8. **To support notifications and communication**  
   Enable email/SMS/app notifications for job updates, application status, interview calls, and employer responses.
9. **To maintain a log of user activity and system operations**  
   Track user interactions and system events for auditing, analytics, and performance monitoring.
10. **To provide analytics and reporting tools**  
    Generate insights for both employers (e.g., number of views on job posts) and platform admins (e.g., usage statistics).

**Functional Requirements**

Functional requirements for a **job portal** specify the core features and operations the system must provide to meet the needs of users such as job seekers, employers, and administrators. These requirements describe what the system should do.

Here are the main user roles (actors) that a job portal typically has, along with their responsibilities, typical actions, and what they expect from the system.

| **Actor / Role** | **Description / Goals** | **Typical Functions / Use Cases** | **Key Requirements from System** |
| --- | --- | --- | --- |
| **Job Seeker** (Candidate) | Someone looking for employment. Their goal is to find and apply for suitable jobs. | - Register / create profile / upload resume / portfolio  - Search / browse job listings (by location, salary, skills, job type)  - Filter, sort, save jobs, get notifications/alerts  - Apply for jobs (submit resume, cover letter, answer questions)  - Possibly track application status  - Manage personal information and preferences | - Easy and powerful job search (good filters, usability)  - Secure and private profile / resume handling  - Alerts / notifications when matching jobs appear  - Transparency in application process (updates)  - Mobile‑friendly interface, good UX |
| **Employer / Recruiter** | Organizations or persons posting job vacancies. Their goal is to find suitable candidates efficiently. | - Register as employer  - Create / post job openings (define job description, criteria, deadlines)  - Search / browse candidate resumes / profiles  - Filter and shortlist candidates  - Communicate with candidates (in‑portal or external)  - Manage job postings (edit, close, renew)  - View/report metrics (number of applications, time to fill, etc.) | - Ability to reach relevant quality candidates  - Tools for filtering / sorting / ranking  - Good dashboard / reporting  - Smooth process for posting and managing jobs  - Security and privacy (of employer data, candidate data) |
| **Administrator (Admin / System Manager)** | Manages and oversees the system itself. Ensures quality, security, compliance, maintenance. | - Manage users (job seekers and employers)  - Approve / moderate job postings or employer registrations (if needed)  - Manage content categories, settings, site‑wide notifications  - Manage payments / subscriptions (if paid model)  - Monitor system performance and usage metrics  - Handle disputes, feedback, support  - Generate administrative reports | - Tools for moderation and management  - Good visibility into system health and usage data  - Ability to enforce policies (anti-spam, misuse, fraud)  - Secure access controls and audit trails  - Scalability, reliability, uptime |

**Primary actors of system**

**Job seeker**

**🎯 Job Seeker Module – Job Portal System**

The **Job Seeker** is a key user role in any **job portal system**. This module enables individuals searching for employment opportunities to register, create profiles, and apply for jobs.

**🧑‍💼 Key Features for Job Seekers**

**1. Registration & Login**

* Sign up with email or social login (Google, LinkedIn, etc.)
* Secure login with username and password

**2. Profile Management**

* Personal Information (Name, Contact, Address)
* Professional Summary
* Education Details
* Work Experience
* Skills
* Certifications
* Upload Resume (PDF/DOC)
* Profile Picture

**3. Job Search & Filter**

* Search by keyword, location, salary, company, industry
* Advanced filters: job type (Full-time, Part-time, Internship, Remote), experience level

**4. Apply for Jobs**

* One-click apply
* Attach resume/cover letter
* Application tracking (view applied jobs, status updates)

**5. Job Alerts / Notifications**

* Set job preferences
* Receive alerts via email or SMS
* Notifications for job matches or application status

**6. Bookmark or Save Jobs**

* Save interesting job listings to apply later

**7. View Company Profiles**

* Company information
* Posted jobs
* Company ratings and reviews

**8. Communication**

* Messaging system to communicate with employers
* Interview call notifications or scheduling

**9. Dashboard**

* Summary of recent activity (applications, saved jobs, alerts)
* Recommendations based on profile

**10. Resume Builder (optional)**

* In-built tool to create a professional resume

Job provider

**✅ Role of a Job Provider in a Job Portal System**

**1. Registration / Login**

* Create an account as an employer.
* Provide company details (name, industry, location, etc.).
* Verify identity (email/SMS verification or business documents).

**2. Create and Manage Company Profile**

* Upload logo and description.
* Add contact info, company culture, and benefits.
* Manage branding for job postings.

**3. Post Job Openings**

* Add job title, description, responsibilities, and requirements.
* Set location, salary range, employment type (full-time, part-time, etc.).
* Define application deadline.

**4. Manage Job Listings**

* Edit or delete job posts.
* Renew expired listings.
* Mark positions as filled.

**5. View & Manage Applications**

* View a list of applicants for each job.
* Access resumes and cover letters.
* Shortlist, reject, or contact candidates.
* Add notes or feedback.

**6. Search and Invite Candidates**

* Use filters to search the candidate database.
* Send interview invites or direct messages.

**7. Schedule Interviews**

* Schedule and manage interviews.
* Send email/SMS notifications to candidates.

**8. Dashboard & Analytics**

* View metrics like:
  + Number of views per job post.
  + Number of applications.
  + Candidate quality ratings.
* Improve future postings based on insights.

**Admin**

1. **You want to build a job portal system**, and you're referring to the **Admin module** (i.e., admin dashboard, features, etc.).
2. **You're the admin of an existing job portal** and need help managing it.
3. You're looking for **sample code** or **project ideas** for an Admin role in a job portal system (e.g., for a school project or software demo).
4. You're asking about **responsibilities or tasks of an admin** in a job portal system

**User stories**

**Jobseeker**

|  |  |  |
| --- | --- | --- |
| As a{type of user} | I NEED TO{do some tasks} | SO THAT I CAN GET {get some result} |
| Job seeker | Register into the application by entering details | Make any date available to job providers and use the system capability reserved for registered users. |
| JOB seeker | Logged in securely with my email and password. Reset my password if needed | I can view and edit my profile |
| Job seeker | I can create my profile with my personal and professional detail. | Make any data available to job providers. |
| JOB seeker | Search for a job according to my preferences | Easily find desired job vacancy and apply directly |
| Job seeker | Apply for a job from the job portal. | Be considered for the position and potentially secure employment |
| Job seeker | Chat option with the seeker provider and admin. | Possible communication between provider admin and seeker. |
| Job seeker | Logout from my account | Protect account from illegal access |

**Job provider**

|  |  |  |
| --- | --- | --- |
| As a {type of user} | I NEED TO[DO SOME TASK] | SO THAT I CAN[get some result] |
| Job provider | Register into application by entering details | Make my data available to job seekers and admin.use the system capabilities reserved for registered users. |
| Job provider | Logged in securely with my email and password and reset my password if needed. | I can view and edit my profile |
| Job Provider | Post a job on the job portal. | Advertise job openings and attract potential candidates. |
| Job Provider | Review accept/reject candidates applications on the job portal. | Assess their qualification and determine whether to move forward with the hiring process. |
| Job provider | Schedule interviews with the selected candidates. | Select the desired candidate. |
| Job provider | Discuss salary expectations. | Select the desired candidate |
| Job Provider | Access the selected candidates skill. | Confirm that the candidate is suitable for the job. |
| Job provider | Communicate with the job seeker and admin | Provide communication with job seekers and admin. |

**Admin**

|  |  |  |
| --- | --- | --- |
| AS A[type of user] | I NEED TO[do some task] | SO THAT I CAN[get some result] |
| Admin | Logged in securely with my email and password.reset my password if needed. | I can view and edit my profile |
| Admin | Create my profile with my personal and professional details. | Make my data available to job providers. |
| Admin | Manage the users | Manage users activity. |
| Admin | Add skill and location | Make sure to obtain the correct data. |
| Admin | Publish advertisement | Make the advertisement visible to the user |
| Admin | Make reports about activities | Improve user acceptance. |

**Use case diagrams**

**Job seeker**

Job seeker

v

Registration

Profile

creation

Job

search

Apply job

Chat

option

Notification

Login

Text

Log out

**Job Provider**

Provider

company research

rEGISTER

LOGIN

Job post

Selection

process

interview

schedule

salary

range

Job alert

and

notification

skill

assesment

chat

option

**Admin**

ADMIN

login

profile

user

management

add skill

add

location

adds

**BPMN**

**BPMN — Business Process Model and Notation**

**BPMN** stands for **Business Process Model and Notation**. It's a **standardized graphical notation** used to model **business processes** in a workflow. It helps stakeholders understand the flow of a process using visual diagrams.

**🔹 Purpose of BPMN**

* To **visualize** business processes clearly.
* To allow **technical and non-technical users** to understand workflows.
* To support **process analysis, automation, and optimization**.

**NON Functional Requirements**

**non-Functional Requirements (NFRs)** define **how** a system performs its functions, rather than **what** it does. They set the **quality attributes** or **constraints** for a system.

**Common Types of Non-Functional Requirements**

Here are the major categories of NFRs:

| **Category** | **Description** |
| --- | --- |
| **Performance** | Response time, throughput, latency (e.g., “system must respond in <1s”) |
| **Scalability** | Ability to handle growth in users or data |
| **Security** | Authentication, authorization, data protection (e.g., encryption) |
| **Availability** | Uptime requirements (e.g., “99.9% uptime”) |
| **Reliability** | System stability and fault tolerance |
| **Maintainability** | Ease of updates, debugging, and code changes |
| **Usability** | User-friendliness, accessibility, UI standards |
| **Portability** | Ability to run on different platforms or environments |
| **Compliance** | Adherence to legal or regulatory standards (e.g., GDPR, HIPAA) |
| **Recoverability** | How quickly the system can recover from failures |
| **Capacity** | Maximum load the system can handle (e.g., 10,000 users simultaneously) |
| **Auditability** | Support for logging, tracking changes, and audit trails |

**Technical Requirements and environment** The **Technical Environment** refers to the **infrastructure and tools** used to run, test, and maintain the application — both during development and in production.

**📌 Components of a Technical Environment:**

| **Environment Layer** | **Example** |
| --- | --- |
| **Operating System** | Ubuntu 22.04 LTS, Windows Server 2022 |
| **Servers / Cloud** | AWS EC2, Azure App Service, Google Cloud Platform (GCP) |
| **Containers** | Docker, Kubernetes |
| **Development Tools** | VS Code, IntelliJ, Postman, Get |
| **Build Tools** | Maven, Grade, Web pack |
| **Testing Tools** | JUnit, Selenium, Postman, Jest |
| **Monitoring & Logging** | Prometheus, Granma, ELK, Data dog |
| **Databases** | MySQL, PostgreSQL, MongoDB |
| **Network & Security** | Firewalls, SSL/TLS, VPNs, OAuth |
| **Dev/Test/Prod Environments** | Separate environments for development, testing, and production |

**🔹 Why This Matters**

Understanding and defining the technical requirements and environment:

* Ensures **compatibility** and **performance**
* Supports **deployment planning**
* Helps avoid costly rework
* Guides **DevOps**, **QA**, and **development** teams

**✅ Quick Example**

If you're building a **web application**, a basic technical requirement and environment summary might look like:

**Technical Requirements:**

* Frontend: React 18
* Backend: Node.js 20 with Express
* Database: PostgreSQL 15
* API: RESTful APIs with JWT auth
* Code Repository: Git (GitHub)
* CI/CD: GitHub Actions

**Technical Environment:**

* OS: Ubuntu 22.04 LTS
* Dev Tools: VS Code, Docker Desktop
* Servers: AWS EC2
* Container: Docker
* Logging: ELK Stack
* Testing: Jest, Cypress
* Monitoring: Prometheus + Grafana

**Conclusion**

This Software Requirements Specification (SRS) provides a detailed and structured outline of the functional and non-functional requirements of the system. It serves as a foundational reference for all stakeholders, including developers, testers, designers, and project managers, ensuring a shared understanding of the system's objectives, constraints, and expected behavior.

By clearly defining the system's scope, interfaces, user interactions, performance expectations, and technical environment, this document reduces ambiguity, supports planning, and lays the groundwork for effective design, implementation, and testing phases.

Successful development and delivery of the system will depend on adherence to these specified requirements, as well as ongoing collaboration and validation throughout the software development lifecycle.

Top of Form

Bottom of Form