

FULL STACK ENGINEERING REPORT

On

Job Era

Submitted in partial fulfilment of the requirement for the
Course Full Stack Engineering BEE (22CS037) of

**COMPUTER SCIENCE AND ENGINEERING
B.E. Batch-2022**

in

Jan -2025



Under the Guidance of
Mr. Rahul

Submitted By:

Madhav Garg: 2210991863
Kashish Gupta: 2210991768
Khushi: 2210991793
Kavita Trivedi: 2210991774
Lovish Bhateja: 2210991857

CERTIFICATE

This is to be certified that the project entitled “Job Era” has been submitted for the Bachelor of Computer Science Engineering at Chitkara University, Punjab during the academic semester January 2025- May-2025 is a bonafide piece of project work carried out by Madhav Garg (2210991863), Kashish Gupta (2210991768), Kavita Trivedi (2210991774), Khushi (2210991793), Lovish Bhateja (2210991857) of the group G-24 towards the partial fulfillment for the award of the course Full Stack Engineering (22CS037) under the guidance of “Project Guide Name” and supervision.

Sign. of Project Guide :

Name of Project Guide

(Designation & Department)

CANDIDATE'S DECLARATION

We, Madhav Garg (2210991863), Kashish Gupta (2210991768), Kavita Trivedi (2210991774), Khushi (2210991793), Lovish Bhateja (2210991857) of the student group G-24, B.E.-2022 of the Chitkara University, Punjab hereby declare that the Integrated Project Report entitled “**Job Era**” is an original work and data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other Institute for the award of any other course.

Sign. of Student 1
Madhav Garg
ID No 2210991863

Sign. of Student 2
Kashish Gupta
ID No 2210991768

Sign. of Student 3
Khushi
ID No 2210991793

Sign. of Student 4
Kavita Trivedi
ID No 2210991774

Sign. of Student 5
Lovish Bhateja
ID No 2210991857

Place:

Date:

ACKNOWLEDGEMENT

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced my thinking, behavior and acts during the course of study.

We express our sincere gratitude to all for providing me an opportunity to undergo Integrated Project as the part of the curriculum.

We are thankful to “Mr. Rahul” for his support, cooperation, and motivation provided to us during the training for constant inspiration, presence and blessings.

We also extend our sincere appreciation to “Mr. Rahul” who provided his valuable suggestions and precious time in accomplishing our Integrated project report.

Lastly, We would like to thank the almighty and our parents for their moral support and friends with whom we shared our day-to day experience and received lots of suggestions that improve our quality of work.

Madhav Garg	Kashish Gupta	Kavita Trivedi	Khushi	Lovish Bhateja
ID No 2210991863	ID No 2210991768	ID No 2210991774	ID No 2210991793	ID No 2210991857

(Annexure –F)

The report must consist of following chapters:

1. Abstract/Keywords
2. Introduction to the project
 - 2.1 Background
 - 2.2 Problem Statement
3. Software and Hardware Requirement Specification
 - 3.1 Methods
 - 3.2 Programming/Working Environment
 - 3.3 Requirements to run the application
4. Database Analyzing, design and implementation (If any)
5. Program's Structure Analyzing and GUI Constructing (Project Snapshots)
6. Code-Implementation and Database Connections (If any)
7. System Testing (if any)
8. Limitations (if any)
9. Conclusion
10. Future Scope
11. Bibliography/References

ABSTRACT

The Job Era is a comprehensive platform designed to simplify the job search and hiring process. By providing a user-friendly interface, efficient job listings, and seamless application tracking, the portal helps job seekers connect with potential employers while enabling recruiters to find qualified candidates quickly. With features like resume uploads, job alerts, company profiles, and direct employer-candidate communication, the platform enhances the hiring experience. By offering personalized job recommendations, secure authentication, and an intuitive dashboard, the Job Portal Website empowers users to navigate the job market efficiently and achieve their career goals.

INTRODUCTION

2.1 Background

With the increasing demand for efficient and accessible job search platforms, traditional hiring methods often fall short in providing seamless job applications, recruiter-candidate interactions, and real-time job updates. Many job seekers face challenges such as limited access to relevant job opportunities, delayed application responses, and a lack of transparency in the hiring process. Similarly, recruiters struggle with managing large volumes of applications, filtering suitable candidates, and streamlining the hiring workflow. The Job Portal Website addresses these challenges by offering a centralized, user-friendly, and scalable platform that simplifies the job search and recruitment process, ensuring convenience, efficiency, and accessibility for both job seekers and employers.

2.2 Problem Statement

The increasing demand for efficient and accessible employment opportunities has highlighted the need for a reliable job portal that ensures a seamless hiring experience for both job seekers and recruiters. Traditional job search methods often face challenges such as lack of transparency, delayed responses, and limited accessibility. A user-friendly job portal can provide a flexible, real-time, and secure environment for job searching and hiring, offering features such as automated job recommendations, real-time application tracking and employer-job seeker feedback mechanisms.

SOFTWARE AND HARDWARE REQUIREMENT SPECIFICATION

3.1 Methods

- **Frontend Development:** React.js for a dynamic user interface with Redux and CSS for styling.
- **Backend Development:** Node.js with Express.js for handling API requests and real-time interactions.
- **Database Management:** MongoDB for scalable and secure data storage.

3.2 Programming/Working Environment

- Development Tools: Visual Studio Code, GitHub for version control.
- Deployment: Database on MongoDB Atlas.

3.3 Requirements to Run the Application

- Node.js environment for backend execution.
- React-compatible browser for frontend access.
- API keys for third-party integrations (Google Books API, YouTube Data API, Spotify API).

DATABASE ANALYSIS, DESIGN, AND IMPLEMENTATION

- **User Profiles:** Stores job seeker and recruiter information with secure authentication, including login credentials, personal details, resume uploads, and saved job listings.
- **Job Listings:** Maintains a database of job postings, including job descriptions, company details, required qualifications, application deadlines, and status updates.
- **Company Profiles:** Stores employer details, including company information, job postings, and recruiter contacts.
- **Messaging & Notifications:** Manages communication between recruiters and job seekers, including interview invitations, job alerts, and real-time notifications.



PROJECT SNAPSHOTS

Job Portal

Home Browse Jobs

Login

Register

No.1 Job Hunt Website

Search Apply & Get Your Dream Job

Start your hunt for the best, life-changing career opportunities from here in your selected areas conveniently and get hired quickly.

Find Your Dream Job



Categories

Explore our extensive job market.



Frontend Developer

Backend Developer



Latest & Top Job Openings

Job Portal

Home Browse Jobs

Login

Register

Login

Email

johndoe@gmail.com

Password

Adhar Card Number

123456789012

☐ Student ☐ Recruiter

Login

Create new Account

Register



Register

Fullname

John Doe

Email

johndoe@gmail.com

Password

PAN Card Number

ABCDEF1234G

Adhar Card Number

123456789012

Phone Number

+1234567890

☐ Student ☐ Recruiter

Profile
Photo

Choose File No file chosen

Register

Already have an account? [Login](#)



```
...  ← →  JOB-PORTAL-master  [Profile]  [Icons]

.env  JS application.controller.js X  Login.jsx

JOB-PORTAL-master > Backend > controllers > JS application.controller.js > ...

1  import { Application } from "../models/application.model.js";
2  import { Job } from "../models/job.model.js";
3
4  export const applyJob = async (req, res) => {
5    try {
6      const userId = req.id;
7      const jobId = req.params.id;
8      if (!jobId) {
9        return res
10         .status(400)
11         .json({ message: "Invalid job id", success: false });
12      }
13      // check if the user already has applied for this job
14      const existingApplication = await Application.findOne({
15        job: jobId,
16        applicant: userId,
17      });
18      if (existingApplication) {
19        return res.status(400).json({
20          message: "You have already applied for this job",
21          success: false,
22        });
23      }
24      //check if the job exists or not
25      const job = await Job.findById(jobId);
26      if (!job) {
```



```
File Edit Selection View ... < > JOB-PORTAL-master
.env main.jsx Login.jsx X
JOB-PORTAL-master > Frontend > src > components > authentication > Login.jsx > Login
14 const Login = () => {
29   };
30
31   const submitHandler = async (e) => {
32     e.preventDefault();
33
34     try {
35       dispatch(setLoading(true)); // Start loading
36       const res = await axios.post(`${USER_API_ENDPOINT}/login`, input, {
37         headers: { "Content-Type": "application/json" },
38         withCredentials: true,
39       });
40       if (res.data.success) {
41         dispatch(setUser(res.data.user));
42         navigate("/");
43         toast.success(res.data.message);
44       }
45     } catch (error) {
46       toast.error("Login failed");
47     } finally {
48       dispatch(setLoading(false)); // End loading
49     }
50   };
51
52   useEffect(() => {
53     if (user) {
```

Progress Till Now

1. Core Features Implemented:

- Job posting and application system is fully functional, allowing recruiters to post jobs and job seekers to apply seamlessly.
- User authentication (JWT-based login/registration) has been successfully integrated for secure access.
- Resume upload and profile management features enable job seekers to maintain a professional portfolio.

2. Dynamic Functionality Implemented:

- Real-time job alerts and notifications allow users to stay updated on new job postings and application statuses.
- Search and filtering options have been optimized to help candidates find relevant jobs efficiently.
- Employer dashboard allows recruiters to track job applications and manage candidate interactions.

3. Backend Infrastructure Established:

- The backend, powered by Express.js and Node.js, provides secure and scalable API endpoints for frontend communication.
- Database (MongoDB) integration ensures efficient storage and retrieval of user data, job listings, and applications.
- Security measures like JWT authentication, CORS handling, and encryption have been implemented to protect user data.

4. Data Models Developed:

- Structured database models for user profiles, job postings, applications, and recruiter dashboards are in place.
- The system is designed for future scalability, allowing for potential integrations such as third-party APIs (LinkedIn, Google Jobs API) and premium job posting features.

SYSTEM TESTING

- **Integration Testing:** Postman utilized to verify API endpoints.
- **User Testing:** Conducted surveys to improve UX based on feedback.

LIMITATIONS

- **Scalability Constraints:** High traffic may require additional server resources.
- **Limited Offline Accessibility:** Platform requires an internet connection for full functionality.

CONCLUSION

Job Portal websites are evolving rapidly with advancements in AI. These platforms play a crucial role in connecting job seekers with the right opportunities while helping employers find the best talent efficiently. Features like AI-powered job matching and skill-based assessments are transforming the recruitment process. With the rise of remote work and personalized job recommendations, job portals must adapt to changing market trends. Ensuring user-friendly experiences and data security will be key to their long-term success. As technology continues to advance, job portals will become even more intelligent, making hiring faster, fairer, and more effective for everyone.

FUTURE SCOPE

- Intelligent algorithms can match candidates to job based on skills, experience and performance.
- AI can automatically analyze resume and shortlist candidates, reducing recruiter and workload.
- AI can suggest peer recommended jobs based on mutual connections.
- Detailed company profiles with work culture insights, employee testimonials, and virtual office tours.

BIBLIOGRAPHY

- [Google Books API Documentation](#)
- [YouTube Data API Documentation](#)
- [MongoDB Documentation](#)