

A
Project Report
On
Job Portal

Developed at
S.M Soni Infosoft
Rudra Complex, First Floor,
Parivar crossroads, waghodia, Vadodara,
Gujarat 390018

Developed by
Vrund Patel – Department of IT, DD University

Guided By

Internal Guide: Prof. Nilamba Vala Department of Information Technology Faculty of technology Dharmsinh Desai University	External Guide: Satyesh Soni S.M Soni Infosoft, Vadodara
---	---



**Department of Information Technology Faculty of
Technology, Dharmsinh Desai University College Road,
Nadiad-387001 April - 2025**



S.M. Soni Infosoft

everyone gets Education but important thing is Knowledge

INTERNSHIP COMPLETION CERTIFICATE

Date: 07th April 2025

This is to certify that **Vrund Harishbhai Patel** has interned as a **Python Development** at **S.M.INFOSOFT** from **02nd December 2024 to 07th April 2025**.

He worked on project, designing and developing websites for various clients using the latest technologies.

During the internship, he demonstrated good design skills with a self-motivated attitude to learn new things. His performance exceeded expectations and he was very punctual throughout.

We wish him every success in his life and career.

For, **S.M.SONI INFOSOFT**



Mr. Satyesh Soni

[Managing Director]

CANDIDATE'S DECLARATION

I declare that the final semester report entitled **Job Finder** is my own work conducted under the supervision of the external guide **Mr. Satyesh Soni**

I further declare that to the best of my knowledge the report for B.Tech. final semester does not contain part of the work which has been submitted for the award of B.Tech. Degree either in this or any other university without proper citation.

Also, I declare that following students also worked in this project:

Vrund Patel – Department of IT, Dharmsinh Desai University

Candidate's Signature

Candidate's Name: Vrund Patel

Branch: **Information Technology**

Student ID: **21ITUON155**

DHARMSINH DESAI UNIVERSITY NADIAD-

387001, GUJARAT



CERTIFICATE

This is to certify that the project entitled "Job Finder" is a Bonafide report of the work carried out by **Mr. Patel Vrund Harishbhai** Student ID No: **21ITUON155** of Department of Information Technology, semester VIII, under the guidance and supervision for the award of the degree of Bachelor of Technology at Dharmsinh Desai University, Nadiad. (Gujarat). He was involved in Project training during the academic year 2024-2025.

Prof. Nilamba Vala,
(Project Guide)
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

Prof. (Dr.) V. K. Dabhi,
Head, Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to all those who have helped me directly or indirectly during this training period. I am more thankful to the department and institute who have given us the opportunity to work in industry.

I am thankful to **Mr. Satyesh Soni**, CEO, for giving me the opportunity to work at their esteemed organization and encouraging me to push my boundaries during this training period. I am indeed thankful to Mr. **Karan Patel**, Senior Developer who has been a great mentor during this time and helped me a lot during the project work.

My sincere thanks to the internal guide **Prof. Nilamba Vala** and Head of the Department **Prof. (Dr.) V.K. Dabhi**, who gave me the opportunity to gain practical knowledge in the industry. I am grateful to them for their guidance, encouragement and insightful support during the training period.

Also, I would like to thank the employees of **S.M Soni Infosoft**, all the faculty members of our IT department, friends, and family for encouraging us throughout the project.

With sincere regards,
Patel Vrund H.

TABLE OF CONTENTS

ABSTRACT.....	i
COMPANY PROFILE	ii
TABLES.....	iii
1. List of Figures.....	iii
2. List of Tables.....	iii
ABBREVIATIONS.....	iv
0.Traning Taken at Industry	1
0.1. Objectives of different types of training taken	1
0.2. Duration and schedule of each training phase	1
0.3. Tools and platform used	1
0.4. Week wise list of topics covered during training	2
0.5. Conclusion and feedback of training	2
1. Introduction	3
1.1.Project Details	3
1.1.1. Overview	3
1.1.2. User Roles and Functionalities	4
1.2.Purpose	4
1.3.Scope	4
1.3.1. User Segmentation	4
1.4.Objective	5
1.4.1. what it can do	6
1.4.2. what it can't do	6
1.5.Technology and Literature Review	7
2. Project Management	8
2.1 Feasibility Study	8
2.1.1. Technical feasibility	8
2.1.2. Time schedule feasibility	8
2.1.3. Economic feasibility	8
2.1.4. Implementation feasibility	8
2.2 Project Planning	9
2.2.1. Project Development Approach and Justification	9
2.2.2. Project Plan	10
2.2.3. Work Breakdown Structure	10
2.2.4. Timeline and Milestones	11
2.2.5. Roles and Responsibility	11

3. System Requirements Study	12
3.1 Study of Current System	12
3.1.1 Overview of Existing System	12
3.1.2 Current System Workflow	12
3.1.3 User Feedback pain points	12
3.1.4 Technology Stack and Infrastructure	12
3.1.5 Data Management and security	12
3.1.6 Reporting and analytics	13
3.2 Problems and Weaknesses of Current System	13
3.3 User Characteristics	13
3.4 Hardware and Software Requirements	13
3.4.1 Hardware Requirements	13
3.4.2 Software Requirements	13
3.4.3 Third-Party Integrations	13
3.5 Constraints	13
3.5.1 Hardware Limitations	13
3.5.2 Network Limitations	14
4. System Analysis	15
4.1. Study of Current System	15
4.1.1. User Requirements	15
4.1.2 . System Requirements	15
5. System Design	20
5.1 System Architecture Design	20
5.1.1. Class Diagram	20
5.1.2. Use case Diagram	21
5.1.3. Sequence Diagrams	21
5.1.4. Activity Diagram	26
5.1.5. Deployment Diagram	28
6. Implementation Planning	37
6.1. Implementation Environment (Single vs Multi User, GUI vs Non-GUI)	29
6.2. Program/Modules Specification	29
6.3. Coding Standards	29
7. Testing	31
7.1. Testing Plan	31
7.2. Integration Testing Plan	31
7.3. System Testing Plan	32
7.4. Acceptance Testing Plan	33

8. User Manual	34
9. Limitation and Future Enhancement	45
10. Conclusion and Discussion	51
10.1. Conclusions	45
10.2. Discussion	45
10.2.1. Self-Analysis of Project Viabilities	46
10.2.2. Problem Encountered and Possible Solutions	46
10.2.3. Summary of Project work	47
Reference	48
Experience	49

ABSTRACT

This **Job Finder** platform is designed to streamline the recruitment process by connecting job seekers with potential employers in a seamless and efficient manner. The system provides two types of user roles: Job Seekers and Recruiters, each with a dedicated dashboard to manage their respective activities.

Job Seekers can **register, create profiles, upload resumes**, and apply for jobs that match their skills and experience. They can also track their application status in real-time. Recruiters, on the other hand, can create company profiles, post job openings, review candidate applications, and either accept or reject applicants based on their qualifications and resumes.

The platform is built using the **MERN stack (MongoDB, Express.js, React, and Node.js)**, ensuring high performance and scalability. The frontend is developed using React and JSX for a dynamic and responsive user experience, while the backend leverages Express.js and Node.js for robust API handling. MongoDB is used for efficient data storage and retrieval, providing structured management of users, jobs, and applications.

To ensure smooth communication between the frontend and backend, RESTful APIs have been implemented and tested using **Postman**. **Secure authentication and authorization mechanisms** are in place to protect user data and maintain privacy.

With a **user-friendly interface and real-time updates**, this job portal simplifies the hiring process, making job search and recruitment more efficient and accessible for both job seekers and employers.

COMPANY PROFILE

About Us

S.M. Soni Infosoft is a dynamic IT solutions provider based in India, dedicated to delivering comprehensive software development, web design, and digital transformation services. With a client-first approach, we specialize in developing custom solutions that align perfectly with our clients' goals and business requirements.

Mission Statement

To empower businesses through innovative, efficient, and scalable technology solutions that drive growth and digital excellence.

Core Services

- **Website Design & Development**
- **Mobile App Development** (Android & iOS)
- **Software Development** (Custom & Enterprise)
-  **E-Commerce Solutions**
- **Digital Marketing & SEO Services**
-  **Cloud Integration & Hosting**
- **Security & Maintenance Services**

Technologies We Work With

- HTML5, CSS3, JavaScript, React.js
- PHP, Laravel, .NET
- MySQL, MongoDB
- Android Studio, Flutter
- AWS, Azure, Cloudflare

Contact Information

Website: <https://smsoniinfosoft.in>

Email: info@smsoniinfosoft.in

Address: 2nd Floor, Vishwakarma Complex, Near Shital Park, Rajkot, Gujarat - 360005

Phone: +91-94289 86575

List Of Figures

	Name	Page No.
Fig 1	Agile model life cycle	18
Fig 2	Class diagram	28
Fig 3	Use case diagram	29
Fig 4 to 9	Sequence diagrams	29 to 33
Fig 10,11	Activity diagrams	34,35
Fig 12	Deployment diagram	36
Fig 13 to 24	Screenshots of our web application Admin panel, user panel etc	42 to 48

List of tables

	Name	Page No.
Table 1	Roles and Usability	20

ABBREVIATIONS

- **CRUD:** Create, Read, Update, Delete – Fundamental operations used to manage jobs, users, applications, etc.
- **React.js:** A JavaScript library used to build the frontend interface where users browse and apply for jobs.
- **Node.js & Express.js:** Used to build the backend API that handles logic like job posting, user registration, and applications.
- **MongoDB:** A NoSQL database used to store information such as user data, job listings, and application records.
- **JWT (JSON Web Token):** Used for securely authenticating users and managing sessions across the app.
- **Multer:** Middleware for handling file uploads like resumes and company logos.
- **Cloudinary:** A cloud service used to store and serve uploaded resumes and images.
- **Redux Toolkit:** Helps manage state across the app, like storing logged-in user data or job listings.
- **React Router:** Allows navigation between different pages such as Home, Jobs, Applications, and Dashboard.
- **SMTP / Nodemailer:** Used to send confirmation emails when users register or apply for jobs.

0.TRAINING TAKEN IN INDUSTRY

0.1. OBJECTIVE OF THE DIFFERENT TYPES OF TRAINING UNDERTAKEN

- I have undergone training in HTML, CSS, JavaScript, Bootstrap, SQL, PostgreSQL, and ASP.NET MVC.
- The primary objective of this training is to gain knowledge and hands-on experience in various technologies and platforms widely used in the corporate world today. This training enhances my understanding of web development, database management, and application development, equipping me with essential technical skills for real-world projects.

0.2. DURATION AND SCHEDULE OF EACH TRAINING PHASE

Training Undertaken	Duration
HTML, CSS, JS, Bootstrap with assignment	2 week
MongoDB with basic to advance concepts and assignments	1 week
React,Mern js with basic to advance concepts and assignments	4 week
Api calling,Postman server	1 week

0.3. TOOLS AND PLATFORMS USED

During the training, I utilized the following tools and platforms:

- **Visual Studio Code** – For coding and development.
- **SQL Server** – For learning database management and executing SQL queries.
- **PostgreSQL (PgAdmin)** – For managing and interacting with PostgreSQL databases.
- **MongoDB (MongoDB Compass)** – For handling NoSQL databases and performing CRUD operations.
- **Postman** – For testing and interacting with RESTful APIs.

0.4. WEEK WISE LIST OF TOPICS COVERED DURING TRAINING

Week	Module	Topics Covered
1	HTML, CSS, JS, Bootstrap	<ul style="list-style-type: none"> - Learn Tools - Implement Practice Module - Perform Assignments
2	SQL Server	<ul style="list-style-type: none"> - Learn Basics of SQL - Learn Advanced Concepts - Perform Assignment
3	RESTful APIs	<ul style="list-style-type: none"> - REST Principles - Create APIs in ASP.NET - Postman Testing - Assignments
4	React.js (Frontend)	<ul style="list-style-type: none"> - React Basics - Components - Props & State - Routing (React Router) - Assignments
5	Authentication & Auth	<ul style="list-style-type: none"> - Login/Signup with Identity - Role-Based Auth - JWT Auth - Practice Module
6	Deployment	<ul style="list-style-type: none"> - Host Frontend (Netlify/Vercel) - Host Backend (Azure/Render) - Env Vars - Final Integration

0.5. CONCLUSION AND FEEDBACK ON TRAINING

Attending these training sessions has provided me with a **comprehensive understanding of various technologies and platforms** used in our project, along with their real-time applications. Additionally, the training has enhanced my skills in **time management, group discussions, and team collaboration**, which are essential for effective project execution and professional growth.

1.INTRODUCTION

1.1. PROJECT DETAILS

The Job Portal Project entails the development of a digital platform to facilitate finding new jobs and employees efficiently. The system caters stakeholders which are part of any recruitment lifecycle which mainly includes employers and job seekers.

1.1.1 Overview

The Job Portal System is developed to revolutionize the recruitment process through a smart, digitally-driven platform. Its goal is to automate and simplify every stage of hiring from posting job openings to selecting candidates—while promoting efficiency, transparency, and accountability. By leveraging advanced technologies, the system enhances the experience for both job seekers and recruiters, making the entire process faster, more accurate, and user-friendly.

1.1.2 User Roles and Functionality

Student:

- Students are provided with a user-friendly web interface to log in using their credentials securely.
- If students are new to web site they can easily register with a user-friendly web interface.
- After successful Login, job seeker are directed to home page, which comprises basic information of website and a navbar to redirect the user to all jobs section, my application section and logout.
- In all jobs section job seeker are allowed to view all the open job's and get main details of the job like Job title, category and location. This section also have a button which say job details. On clicking the job details the user will be forwarded to new page where he can get more details and a button to apply for job.
- Gets confirmation mail after applying for jobs successfully.
- Students can't apply after deadline is passed.
- In my application section user will able to view all the applications of applied job. Here user can also delete the job application by clicking on the delete button near the application.
- The logout section will sign out the user and redirect the user to login page.

Recruiter:

- Recruiters are provided with a user-friendly web interface to log in using their credentials securely.
- If recruiters are new to web site, they can easily register with a user-friendly web interface.
- Upon successful authentication, Employer are directed to home page, which comprises basic information of website and a navbar to redirect the user to all jobs section and applicant's application, post new job, view your jobs and logout section.
- In all jobs section Employer are allowed to view all the open job's and get main details of the job like Job title, category and location. This section also have a button which say job details. On clicking the job details the user will be forwarded to new page where he can get more details.
- Applicant's application section the user can view all the application that have been submitted by the job seeker for the jobs posted by the employer. Beside every job seeker application there are 2 button available which are accept application when clicked send mail to the respected job seeker that his application for this job is been accepted and reject application when clicked send mail to the respected job seeker that his application for this job is been rejected.
- In post new job section user need to fill form which include details like job title, category, country, city, location, salary, job description and a button which say create job on clicking that button the job will be successfully created.
- In view your jobs section the user will able to view all the job posted by him and can edit the job by clicking the edit button aside each job and delete job by clicking delete button aside each job.
- The logout section will sign out the user and redirect the user to login page.

1.2. PURPOSE

The Job Portal System is built to transform and modernize the recruitment landscape through an intelligent, digitally-driven platform. It aims to streamline and automate the entire hiring process—from job postings to candidate selection—while ensuring efficiency, transparency, and accountability at every step. By leveraging cutting-edge technology, the system significantly enhances the experience for both job seekers and recruiters, making the recruitment journey faster, more accurate, and effortlessly accessible.

1.3. SCOPE

The Job Portal Application aims to provide a comprehensive digital platform for job seekers and employers to efficiently navigate the recruitment process. The scope of the application encompasses the following features and functionalities.

1.3.1 User Segmentation

The application caters to two primary user segments: job seekers and employers.

1.3.1.1 Student Feature:

- **User Registration:** Job seekers can register on the platform securely, providing necessary details.
- **Dashboard:** Upon logging in, job seekers access a dashboard displaying relevant information and navigation options.
- **View Job Listings:** Job seekers can browse through available job listings which has information such as job title, category, and location.
- **Job Details:** Detailed information about each job listing, including description, requirements, and application instructions.
- **Application Management:** Job seekers can submit applications for desired positions, delete the application.
- **Notifications:** Receive notifications regarding application status updates on mail.

1.3.1.2 Recruiter Features:

- **User Registration:** Employers can register on the platform securely, providing necessary details.
- **Dashboard:** Upon logging in, employers access a dashboard displaying relevant information and navigation options.
- **Job Posting:** Employers can create and manage job listings, including specifying job details, requirements, and application deadlines.
- **Application Management:** Review and manage job applications received from job seekers, including accepting or rejecting applications.
- **Job Management:** Employers can manage their posted jobs, including updating job and deleting jobs.
- **Deadline control:** Recruiter can restrict students from applying to job after deadline.

1.4 Objectives

1.4.1. What it can do

- **Effective Job Discovery and Matching:** The job portal aims to enhance job discovery for job seekers. This helps users discover relevant job opportunities that align with their skills and preferences.
- **Simplify Recruitment Process:** For employers, the platform simplifies the recruitment process by offering intuitive tools for posting job listings, managing applications, and ultimately reducing time-to-hire and improving recruitment efficiency.
- **Enhance User Experience:** The application prioritizes user experience by offering a user-friendly interface, ensuring a seamless and engaging experience for both job seekers and employers.

- **Facilitate Communication:** Effective communication channels between job seekers and employers are provided to facilitate interaction regarding job applications status updates.
- **Promote Transparency and Accountability:** The job portal promotes transparency and accountability by providing detailed information about job listings, including company profiles, job descriptions, and application status updates, empowering both job seekers and employers to make informed decisions.
- **Empower Job Seekers and Employers:** By empowering job seekers to manage their applications and enabling employers to efficiently manage job postings, applications, the platform aims to empower both stakeholders in the recruitment ecosystem.

1.4.2. What it can't do

- **Guarantee Job Placement:** While the job portal facilitates the job discovery and recruitment process, it cannot guarantee job placement for job seekers or hiring outcomes for employers. The final hiring decisions are at the discretion of the employers based on their specific criteria and requirements.
- **Ensure Compliance with Labor Laws:** The application does not ensure compliance with labor laws and regulations, as the responsibility for compliance lies with the users (both job seekers and employers) based on their respective jurisdictions and legal obligations.
- **Override Human Judgment:** Employers are still responsible for assessing candidates' qualifications, suitability, and cultural fit based on their unique organizational needs and preferences.
- **Ensure Accuracy of Information:** While the platform strives to maintain accurate and up-to-date information regarding job listings, company profiles, and candidate profiles, it cannot guarantee the accuracy or authenticity of the information provided by users. Users are encouraged to verify information independently before making decisions based on the platform's content.
- **Handle Physical Recruitment Activities:** The job portal focuses on digital recruitment activities and does not handle physical recruitment activities such as in-person interviews, job fairs, or on-site assessments. Users are responsible for organizing and conducting such activities outside the platform as needed.

1.5 Technology and Literature Review

The implementation of the Job Portal is grounded in the utilization of modern technologies, including React.js for the frontend, Node.js for server-side development, and MongoDB as the database management system. This technology stack was strategically chosen for its versatility, scalability, and effectiveness in developing dynamic web applications.

1.5.1. React.js:

React.js, a widely adopted JavaScript library for building user interfaces, forms the foundation of the system's frontend. Its component-based architecture facilitates the creation of interactive and responsive user interfaces, crucial for seamless and intuitive user experience. The virtual DOM (Document Object Model) implementation in React optimizes rendering, ensuring efficient updates and improved overall performance. The use of React.js aligns with contemporary frontend development practices, providing a robust and modular structure to the user interface.

1.5.2. Node.js:

Node.js serves as the backend runtime environment, enabling server-side JavaScript execution. Leveraging the event-driven, non-blocking I/O model of Node.js ensures high scalability and responsiveness. This is particularly beneficial for handling concurrent operations, such as managing user authentication, file uploads, and communication between the frontend and the database. The Node.js ecosystem, with its extensive package repository (npm), facilitates the integration of third-party libraries and tools, streamlining the development process.

1.5.3. MongoDB:

MongoDB, a NoSQL database, has been employed to store and manage the project's data. Its document-oriented structure aligns well with the dynamic and evolving nature of academic information. MongoDB's flexibility in handling complex data structures, scalability, and support for horizontal scaling make it an ideal choice for this project. The use of MongoDB allows for efficient querying and retrieval of data, crucial for generating reports, analyzing academic performance, and supporting dynamic user interactions.

1.5.4. Literature review:

The selection of React, Node.js, and MongoDB is informed by a review of contemporary literature and best practices in web development. React.js, widely adopted by leading companies, ensures maintainability and reusability of UI components. Node.js, with its asynchronous capabilities, aligns with the requirements of real-time communication and data processing in educational systems. MongoDB's document-oriented approach is well-documented as suitable for systems requiring flexible schema design and quick iterations.

2. PROJECT MANAGEMENT

2.1. Feasibility Study

2.1.1. Technical feasibility:

From a technical standpoint, the Job Portal System is feasible to develop using a combination of appropriate tools and technologies. The chosen technologies provide a solid foundation for building a robust and scalable web application. Below is an overview of the technical feasibility considerations:

For web app development: VS Code

Frontend: React JS

Backend: Express JS, Node JS

Database: Mongo database

For testing: Postman

2.1.2. Time Schedule feasibility:

We have planned the steps for the completion of our project in the given duration. Firstly, we will perform requirement gathering & analysis by the mid of January 2024. We will prepare the SRS document and the GUI design tentatively by January 2024 ending. The diagrams required for the design as well as the database design will be tentatively completed by mid of February 2024. For coding 4 weeks required. Hence tentatively by the end of March 2023. We will be able to complete the project and ready for its demonstration at starting of April 2024. Being a 2-member team, we will be able to complete our project in the estimated time.

2.1.3. Economic Feasibility:

The economic feasibility analysis for the Job Portal System involves evaluating the costs and benefits associated with the project to determine its viability and potential economic advantages. This assessment serves as a crucial step in decision-making, providing insights into the financial implications of investing in the development and implementation of the system.

2.1.4 Implementation Feasibility:

Since we have been taught about technology like React, JavaScript in our college syllabus, we just need to learn how to implement it according to our needs which will take around 2 or 3 weeks and be completed before starting implementation. As this is a MERN stack technology base project we also required to learn MongoDB, Also learn few stuff which are require because of the functionality like for sending mail for updates on job application we learn how to integrate our web application with node-mailer.

2.2. Project Planning:

2.2.1. Project Development Approach and Justification:

For the development of the Job Portal System, we propose the utilization of the Agile methodology. Agile methodologies are characterized by their iterative and incremental approach, which enables flexibility, adaptability, and continuous improvement throughout the development lifecycle. The Agile model aligns well with the dynamic nature of software development projects, allowing for rapid iterations and responsiveness to changing requirements and stakeholder feedback.

Following are the phases in the Agile model are as follows:

1. Requirements gathering
2. Design the requirements
3. Construction/ iteration
4. Testing/ Quality assurance
5. Deployment
6. Feedback



Fig 1 agile methodology life cycle

This diagram shows seven different phases of the agile methodology.

Justification:

- **Iterative Development:** Agile methods break down the project into smaller iterations or sprints, each focusing on delivering a potentially shippable product increment. This approach allows for early and frequent delivery of valuable features, ensuring continuous progress and stakeholder engagement.

- **Flexibility and Adaptability:** The Agile model emphasizes adaptability to changing requirements and priorities. By prioritizing collaboration and communication within cross-functional teams, Agile enables swift adjustments to project scope, priorities, and deliverables based on evolving business needs and user feedback.
- **Customer Collaboration:** Agile methodologies prioritize customer collaboration throughout the development process. Regular reviews and demonstrations enable stakeholders to provide feedback on each iteration, ensuring that the product meets their expectations and aligns with their goals.
- **Risk Mitigation:** Agile methodologies incorporate risk management practices by addressing potential issues early in the development process. By delivering working increments at the end of each iteration, Agile enables early identification and mitigation of risks, thereby reducing project uncertainty and enhancing overall project success.
- **Continuous Improvement:** The Agile model promotes a culture of continuous improvement through retrospective meetings at the end of each iteration. Team members reflect on their processes, identify areas for enhancement, and implement iterative changes to optimize project performance and efficiency.

2.2.2. Project plan:

2.2.2.1. Project Scope and Objectives:

The Job Portal System aims to develop a comprehensive web application for facilitating the job posting, application, and management process for employers and job seekers. The system will provide features such as job posting, job search, application submission, candidate management, and communication tools.

Create a user-friendly interface for employers to post job listings, manage applications, and communicate with candidates. Develop a platform for job seekers to search for relevant job listings, submit applications, and track application statuses. Implement backend functionality for storing job data, managing user profiles, and facilitating communication between employers and job seekers. Enhance the efficiency and effectiveness of the job search and recruitment process for both employers and job seekers.

2.2.3. Work Breakdown Structure:

- Requirements Gathering
- System Design
- Frontend Development
- Backend Development
- Database Setup and Configuration

- Integration with Email Notification System
- Deployment

2.2.4. Timeline and Milestone:

- **Timeline:**

- Requirements Gathering: 4 weeks
- System Design: 4 weeks
- Frontend Development: 4 weeks
- Backend Development: 4 weeks

- **Milestone:**

- Completion of Requirements Gathering
- Finalization of System Design
- Frontend Development Milestone
- Backend Development Milestone
- Database Setup Milestone
- Integration Milestone
- Completion of Testing Phase
- Deployment Milestone
- User Training and Documentation Completion

2.2.5. Roles and Responsibility:

Name	Analysis	Frontend	Backend	Testing	Report
Vrund Patel	Yes	Yes	Yes	Yes	Yes

3.SYSTEM REQUIREMENT STUDY

3.1 Study of Current System:

3.1.1. Overview of Existing System:

The current system, a Job Portal, serves as a platform for job seekers and employers to connect and facilitate the recruitment process. It comprises modules for job posting, job searching, application submission, candidate management, and communication between employers and job seekers.

3.1.2. Current System Workflow:

The workflow begins with employers posting job listings, specifying details such as job title, description, and requirements. Job seekers then search for relevant job listings, submit applications, and track application statuses. Employers review applications, communicate with candidates, and manage candidate profiles within the system.

3.1.3. User Feedback and Pain Points:

Feedback indicates challenges with limited job search filters, cumbersome application submission processes, and difficulties in tracking application statuses. Users also express a need for improved communication tools and enhanced candidate management features.

3.1.4. Technology Stack and Infrastructure:

The system utilizes React JS for frontend development, Express JS and Node.js for backend development, and MongoDB as the database. The infrastructure is scalable, but optimization is required in certain functionalities, such as search and data retrieval, to handle increasing user volumes.

3.1.5. Data Management and Security:

Data management involves secure storage of user profiles, job listings, and application data in MongoDB. However, there are concerns regarding data privacy compliance and potential vulnerabilities in data security protocols that need addressing.

3.1.6. Reporting and Analytics:

Reporting capabilities enable administrators to generate job posting and application reports. However, there is a need for comprehensive analytics to

identify recruitment trends, optimize job listings, and improve decision-making processes.

3.2. Problem and Weakness of Current System:

- Limited job search filters leading to inefficiencies in finding relevant job listings.
- Cumbersome application submission processes causing user frustration.
- Difficulty in tracking application statuses affecting user experience.
- Insufficient communication tools hindering effective employer-candidate interaction.
- Inadequate candidate management features impacting recruitment efficiency.
- Concerns regarding data privacy and security protocols.
- Optimization required to improve system performance and scalability.
- Lack of comprehensive analytics for informed decision-making.

3.3. User Characteristics:

- Recruiter
- Student

3.4. Hardware and Software Requirements:

3.4.1. Hardware Requirements:

- For users accessing the Job Portal System:
- No specific hardware requirements beyond standard devices capable of running a modern web browser.
- Ensure that client devices, such as desktops, laptops, tablets, or smartphones, meet basic specifications for optimal user experience.

3.4.2 Software Requirements:

- For users accessing the Job Portal System:
- The only essential software requirement is a modern web browser.
- Supported web browsers include Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari.
- Ensure that the web browser is up to date to guarantee compatibility and a seamless user experience.

3.4.3. Third-Party Integrations:

- No additional third-party tools or integrations are mandatory for basic functionality of the Job Portal System.

3.5. Constraints:

3.5.1 Hardware Limitations:

The primary constraint of the Job Portal System is that users must have access to a device with a modern web browser.

3.5.2. Network Limitations:

Another primary constraint of the Job Portal System is that users must have access to internet with a modern web browser.

4.SYSTEM ANALYSIS

4.1. Study of Current System:

4.1.1. User Requirements:

Student:

- List Jobs: Allow to view available job.
- Apply to Jobs: Allow to apply to available job.
- Delete Application: Allow user to delete application.
- List Application: Allow to view application.
- Mail confirmation : Students receives confirmation mail after applying for job.

Recruiter:

- Create jobs: Allow to create jobs.
- Update job: Allow to update jobs
- Delete job: Allow to delete job.
- Update application Status: Allow to update Status of Application.
- View Resume: Recruiter can see resume of applicants.
- Job deadline: Recruiter can add deadline to job and restrict students from applying to job after deadline.

4.1.2. System Requirements:

4.1.2.1. Functional Requirements:

Functional requirements are the following:

- **Student:**

S1: Registration:

Input: Student details including the email, password and select role as student.

Precondition: New student details get added to Database.

Output: On confirmation of registration status, the student will be redirected to the home page of the application.

S2: Login:

Precondition: Student details must exist in Database.

Input: Student details including the email, password and role.

Output: Student will be redirected to the home page of the application.

S2: Login:

Precondition: Student details must exist in Database

Input: Student details including the email, password and role.

Output: Student will be redirected to the home page of the application.

S3: Student can view available jobs:

Input: Click on view job.

Output: Student will be redirected to the view job page where they will be able to see all available jobs.

S4: Student can apply for jobs:

Input: Click on any job from the available job section. Student will be redirected to job/:id where they will fill in details and click on send application.

Precondition: New student application will be added in the database.

Output: Student will be redirected to the home page.

S5: Student can view their applications:

Input: Click on my application.

Output: Student will be redirected to the applications page where they will see all their applications.

S6: Student can delete their application:

Input: Click on my application.

Output: Student will be redirected to the application page where they will see all their applications. Now click the delete button of the application they want to delete.

Post-Condition: Student application gets deleted from the database.

- **Recruiter:**

R1: Registration:

Input: Recruiter details including the email and the password and select role as recruiter.

Precondition: New recruiter details get added to Database.

Output: On confirmation of registration status, it will redirect the recruiter to the home page of the application.

R2: Login:

Precondition: Recruiter details must exist in Database.

Output: Recruiter will be redirected to the appropriate home page of the application.

R2: Login:

Precondition: Recruiter details must exist in Database.

Input: Recruiter details including the email, password, and role.

Output: Recruiter will be redirected to the appropriate home page of the application.

R3: Recruiter can view available jobs:

Input: Click on view job.

Output: Recruiter will be redirected to the view job page where they will be able to see all available jobs.

R4: Recruiter can post new jobs:

Input: Click on post new job. The recruiter will be redirected to the add post page of the application, where they will fill in details and click on create job.

Post-Condition: New job will be added to the database.

Output: Recruiter will be redirected to the home page.

R5: Recruiter can view their jobs:

Input: Click on view your jobs.

Output: Recruiter will be redirected to the view your job page where they will get all their jobs.

R6: Recruiter can update their jobs:

Input: Click on view your jobs. The recruiter will be redirected to the view your job page where they will get all their jobs. Then click the edit button, make updates, and click on save changes.

Post-Condition: Recruiter job posting gets updated in the database.

Output: Recruiter will be redirected to the home page where they will see all their jobs.

R7: Recruiter can delete their jobs:

Input: Click on view your jobs. The recruiter will be redirected to the view your job page where they will get all their jobs. Then click on delete.

Post-Condition: Recruiter job posting gets deleted from the database.

Output: Recruiter will be redirected to the application home page.

R8: Recruiter can accept/reject applications:

Input: Click on applicant applications. The recruiter will be redirected to applicant applications, where they will get all applications for their jobs.

From there, they can click accept/reject, which will send an update status to the student via email.

Post-Condition: Email will be sent to the student.

4.1.2.2. Non-Functional Requirements:

4.1.2.2.1. Usability:

- The system must have an intuitive and user-friendly interface to ensure ease of use for both employers and job seekers.
- Navigation within the system should be clear and straightforward, with easily accessible menus and options.
- The system should provide clear instructions and guidance to users, especially during complex processes such as job application submission or posting.
- All forms and input fields should be properly labeled and validated to prevent user errors.

4.1.2.2.2. Reliability:

- The system should be reliable and available for use at all times, with minimal downtime for maintenance or updates.
- Data integrity and consistency should be maintained throughout the system, ensuring that information remains accurate and up-to-date.
- The system should have mechanisms in place to handle errors and exceptions gracefully, providing informative error messages to users when necessary.
- Backup and recovery procedures should be established to prevent data loss in the event of system failures or crashes.

4.1.2.2.3. Performance:

- The system should be responsive and provide quick load times for all pages and functionalities.
- It should be capable of handling a large number of concurrent users without experiencing significant slowdowns or performance degradation.
- Database operations should be optimized to ensure efficient data retrieval and processing.

4.1.2.2.4. Security:

- User authentication should be implemented securely to prevent unauthorized access to the system.
- Access controls should be enforced to ensure that users can only perform actions appropriate to their role and privileges.

- Data encryption should be used to protect sensitive information, such as user credentials and personal details.

5. SYSTEM DESIGN

5.1. System Architecture Design:

5.1.1. Class Diagram:

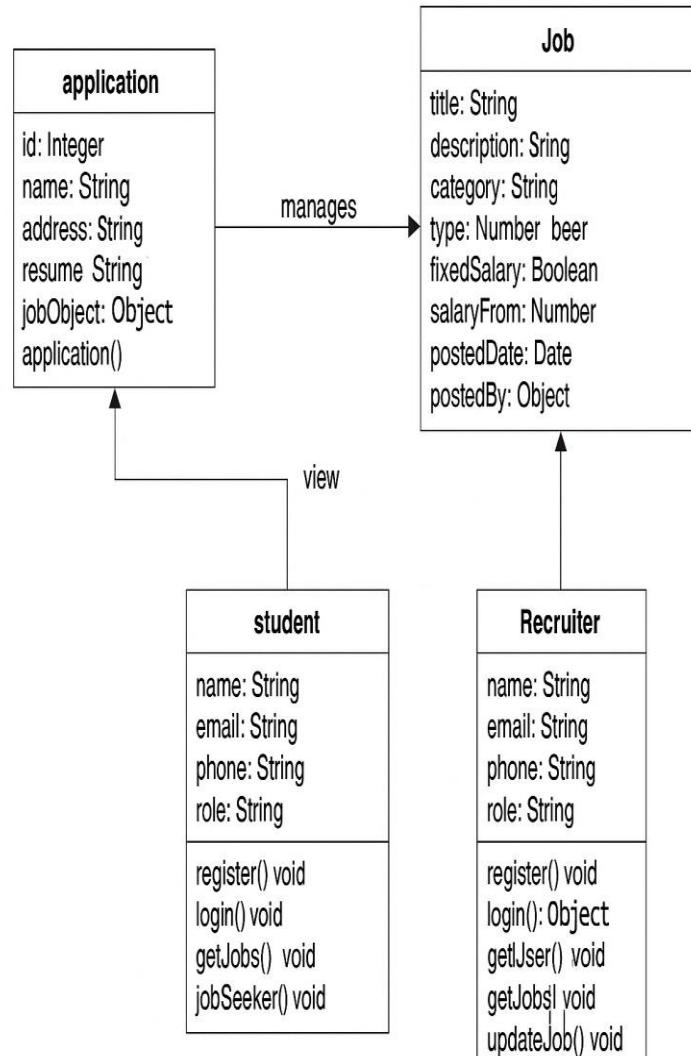


Fig 2 Class diagram

5.1.2. Use Case Diagram:

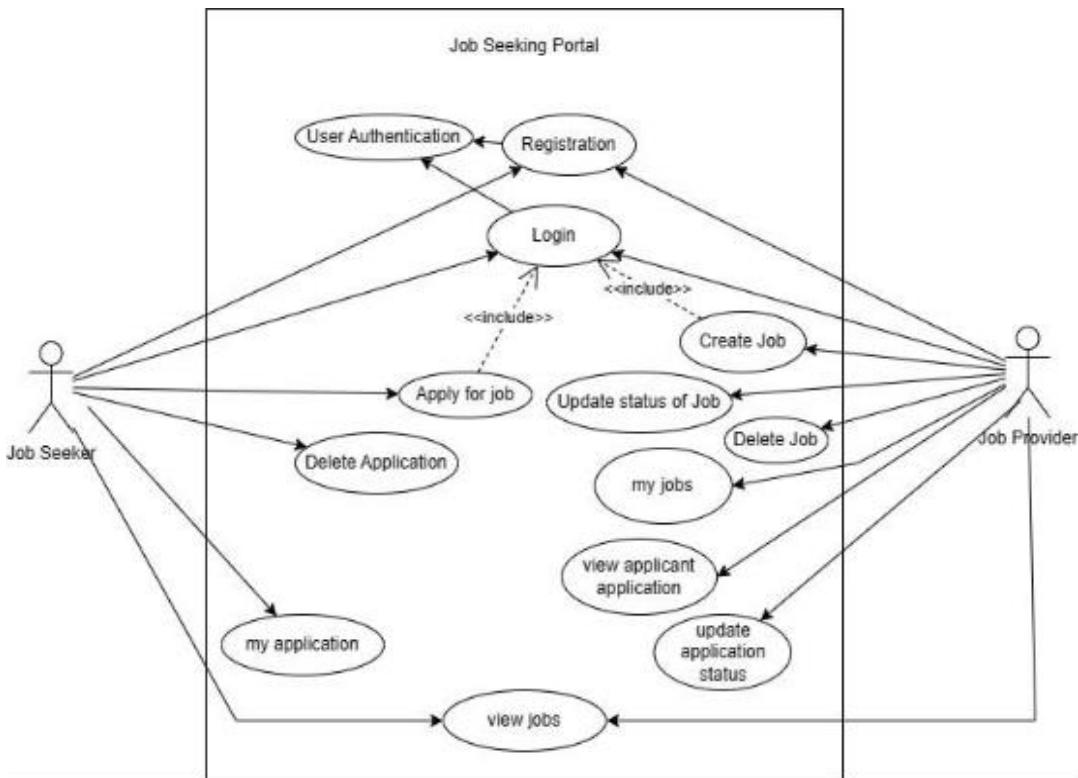


Fig 3 Use-case diagram

5.1.3. Sequence Diagram:

5.1.3.1. Recruiter Login:

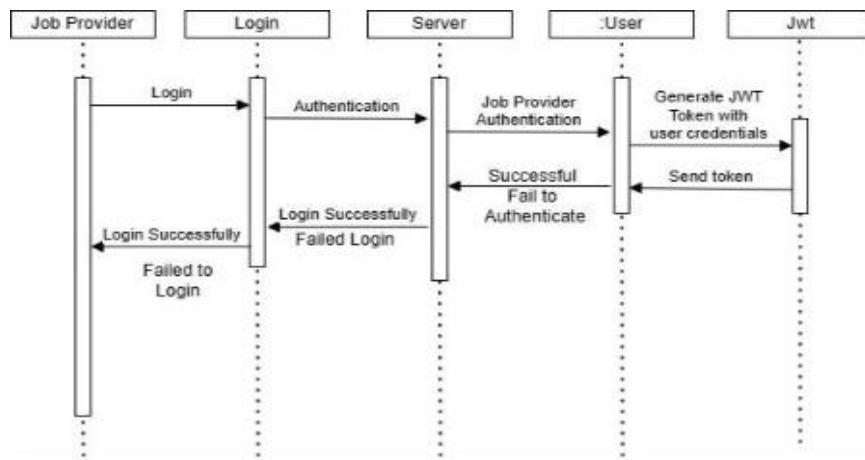


Fig 4 Sequence diagram for recruiter Login

5.1.3.2. Recruiter creating Job:

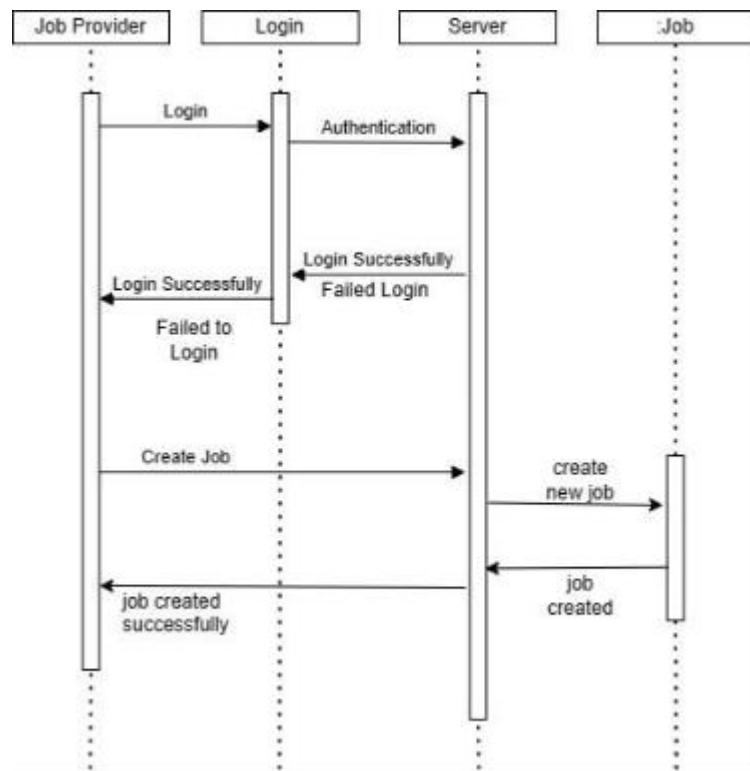


Fig 5 sequence diagram for recruiter creating job

5.1.3.3. Recruiter viewing application's:

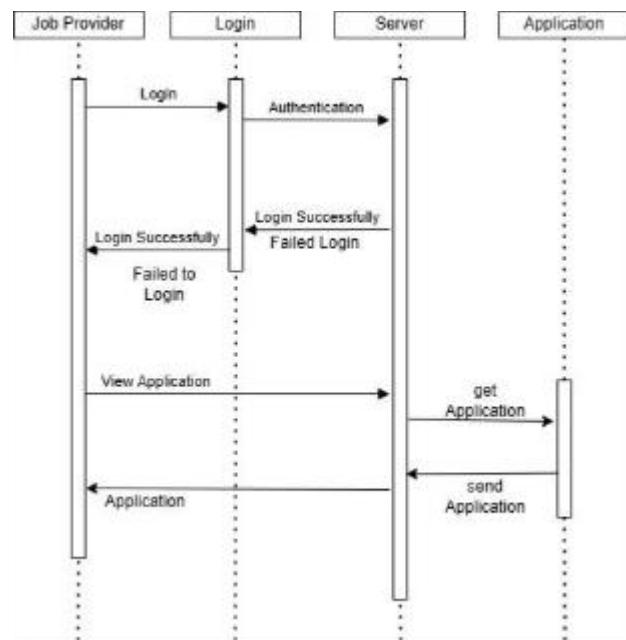


Fig 6 sequence diagram for recruiter viewing application's

5.1.3.4. Recruiter Updating status of application's:

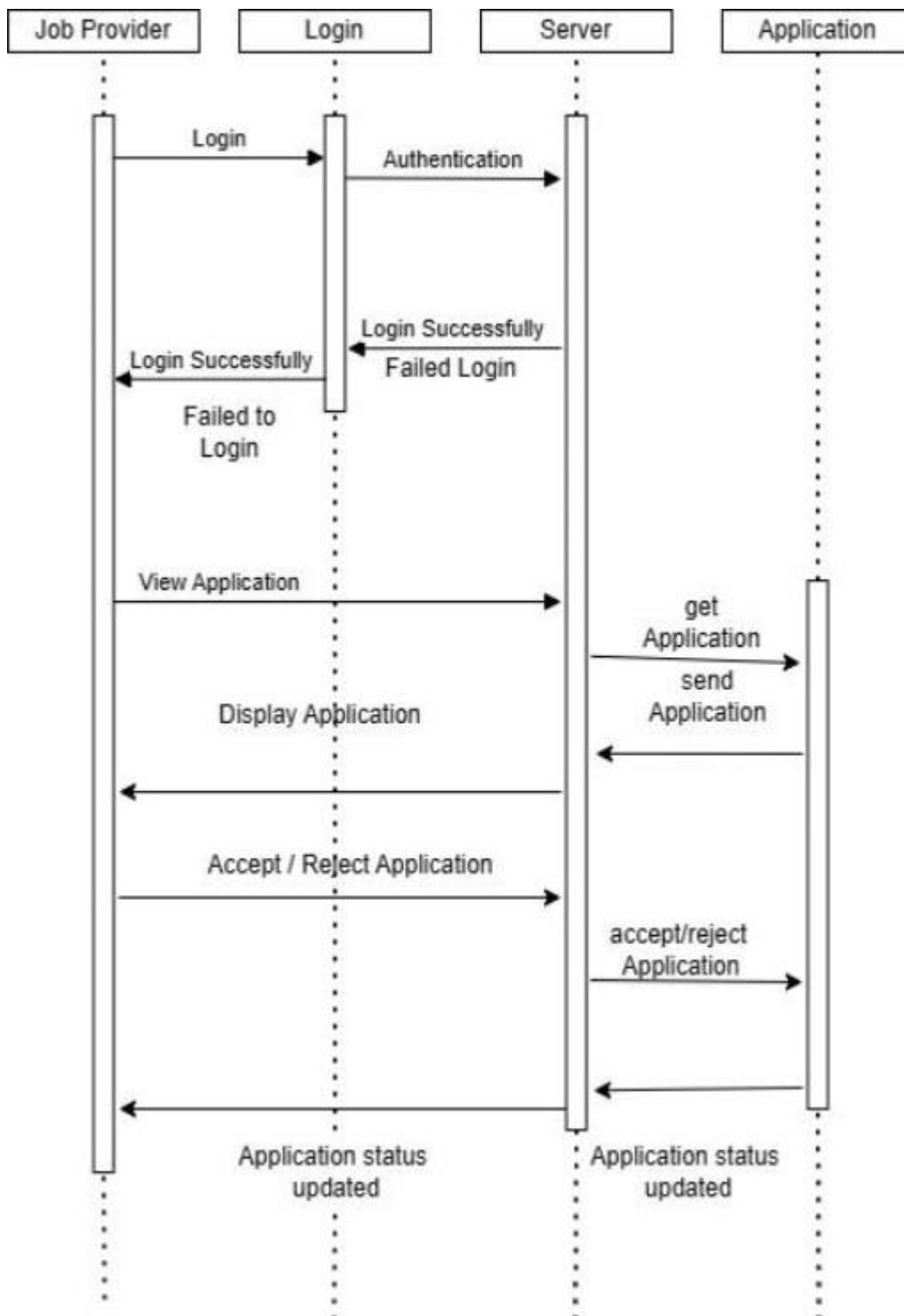


Fig 7 sequence diagram for recruiter updating status of application's

5.1.3.5. Student Applying for job:

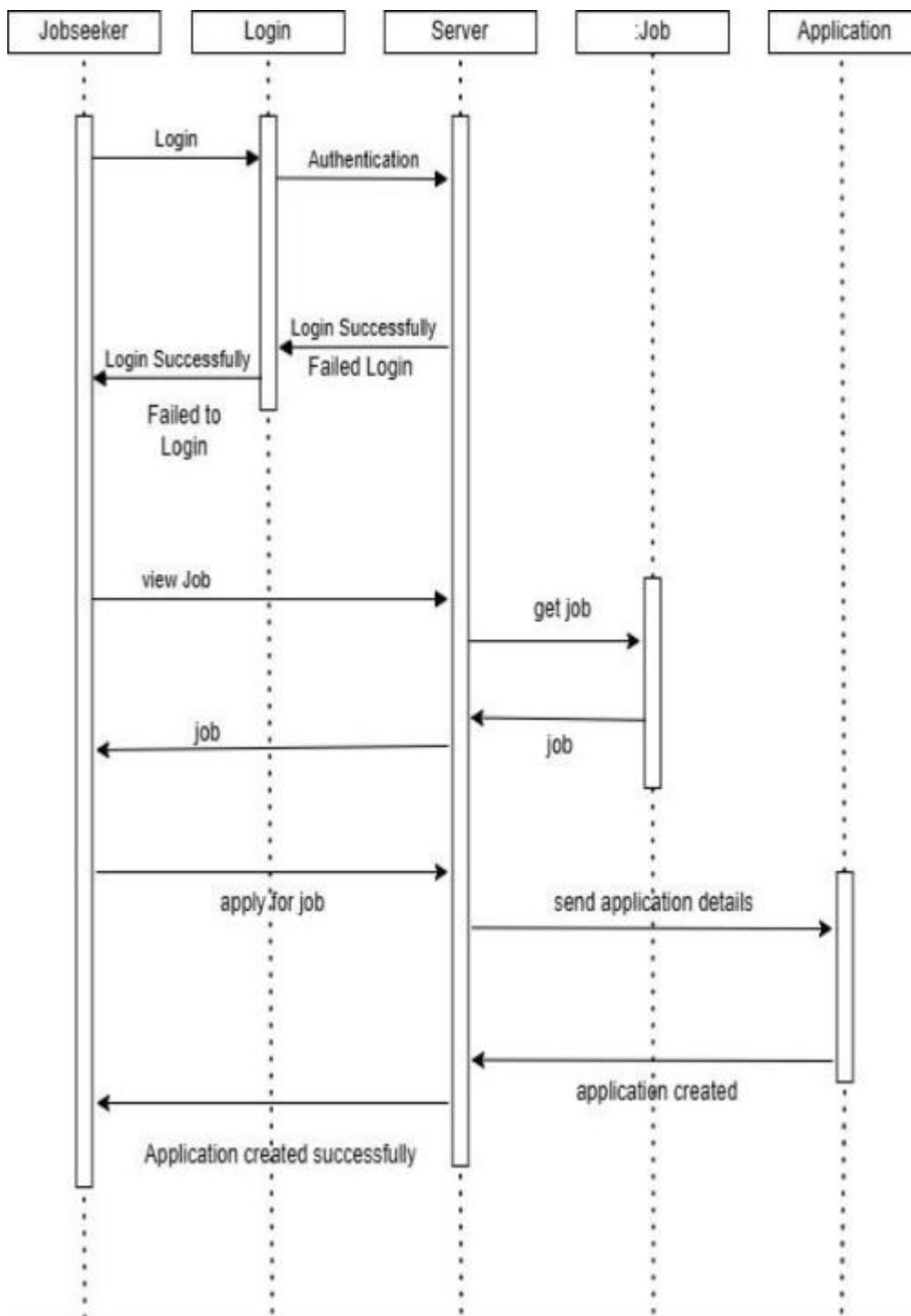


Fig 8 sequence diagram for student applying for job

5.1.3.6. Student deleting application's

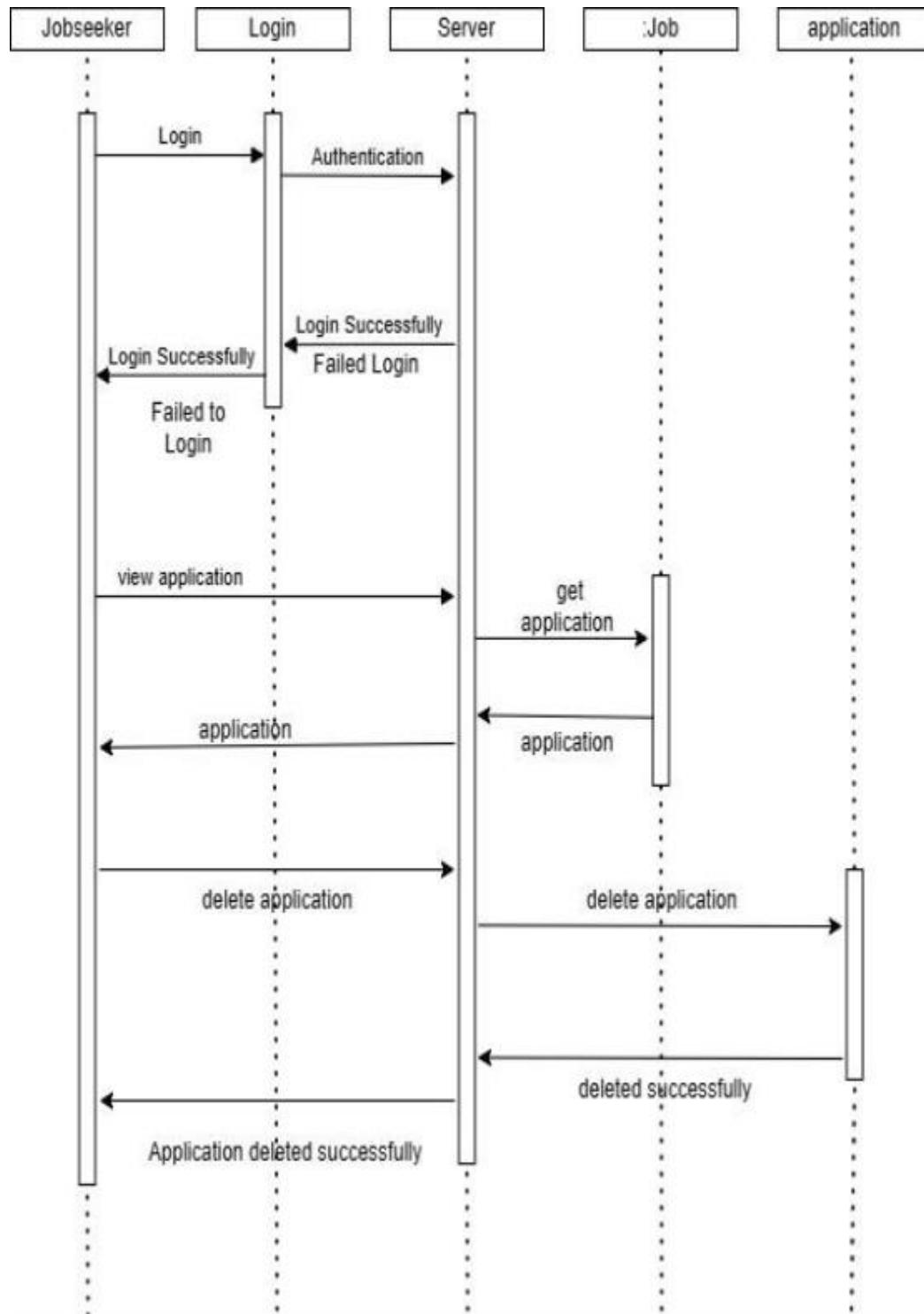


Fig 9 sequence diagram for jobseeker deleting application

5.1.4. Activity Diagram:

5.1.4.1. Activity diagram for recruiter:

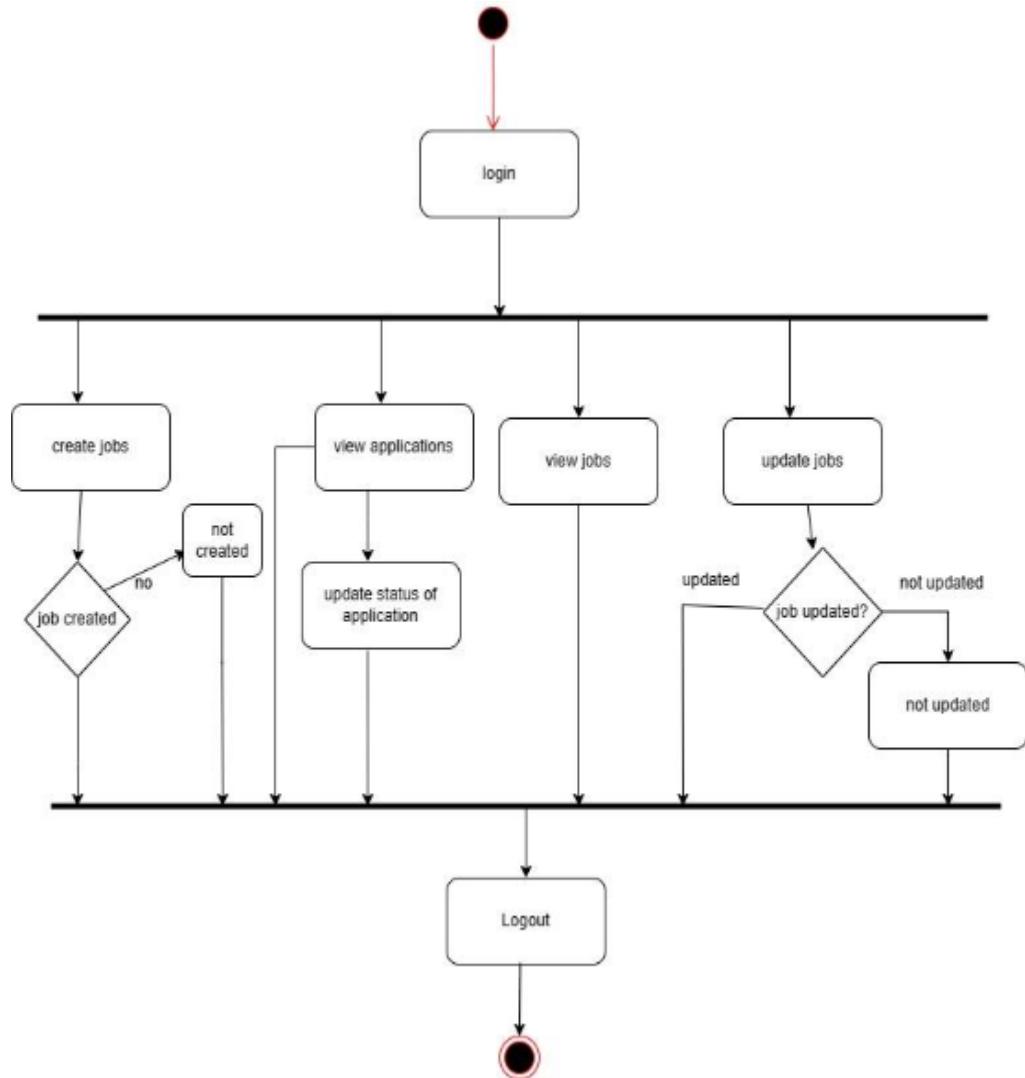


Fig 10 activity diagram for recruiter

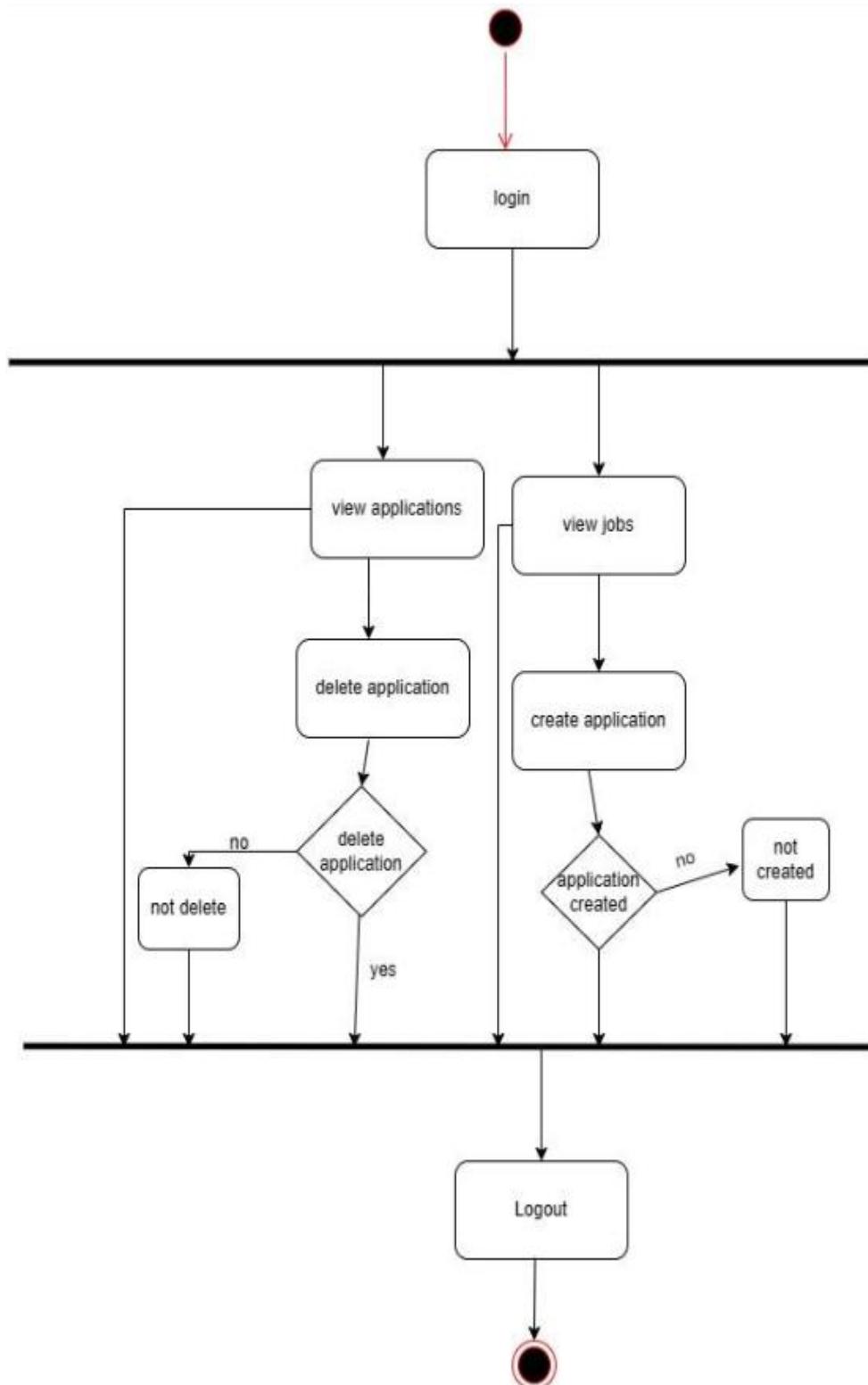
5.1.4.2. Activity diagram for student:

Fig 11 activity diagram for student

5.1.5. Deployment Diagram:

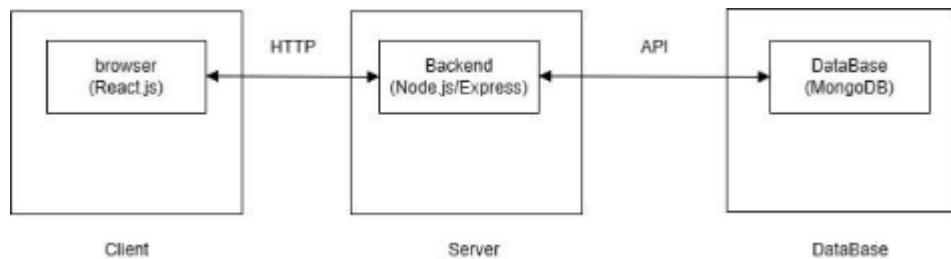


Fig 12 Deployment diagram for job portal

6.IMPLEMENTATION PLANNING

6.1. Implementation Environment (Single vs Multi-User, GUI vs Non-GUI):

Our job portal project is designed as a multi-user system with a graphical user interface (GUI) for enhanced user interaction. We chose Visual Studio Code as our primary development environment due to its robust features and seamless integration with web technologies. Visual Studio Code provides an intuitive interface and essential tools for web development, making it an ideal choice for building our web application.

Furthermore, our system relies on MongoDB as the database management system to store and manage data efficiently. MongoDB's flexibility and scalability ensure seamless operation of our web application, allowing it to run smoothly and handle data storage requirements effectively.

6.2. Program/Modules Specification:

6.2.1. Student:

- View available jobs: Job seekers can browse and view all open job listings.
- Apply for job: Job seekers can submit job applications for positions they are interested in.
- Delete application: Job seekers have the option to delete submitted job applications if needed.

6.2.2. Recruiter:

- View available jobs: Employers can access and view all job listings posted on the platform.
- Create new jobs: Employers can create and post new job listings, specifying job details such as title, description, and requirements.
- Update jobs: Employers can edit and update existing job listings as needed.
- Delete jobs: Employers have the ability to remove job listings from the platform.
- View applications: Employers can view and manage job applications submitted by job seekers for their posted positions.
- Update application status: Employers can update the status of job applications, such as accepting or rejecting applicants.

6.3. Coding Standards:

To make the system coding easy, easy to remember, and reduce the chances of errors, some techniques are used at the time of coding of the application which is called coding standard.

- The coding standard which we adopted during the coding is explained as follows:
- Each nested block should be properly indented and spaced.
- The code should be properly commented on for understanding easily.
- Comments regarding the statements increase the understandability of the code.
- Better to avoid the use of digits in variable names.
- The names of the function should be written in camel case starting with small letters.
- The name of the function must describe the reason for using the function clearly and briefly.

7.TESTING

7.1. Testing Plans:

7.1.1. Unit Testing Plan:

7.1.1.1. Objectives:

- Ensure individual units/modules perform as expected.
- Verify the correctness of functions and methods.

7.1.1.2. Scope:

- Unit tests for critical functions and methods.
- Identification of edge cases and boundary conditions.

7.1.1.3. Responsibilities:

- Developers responsible for writing and executing unit tests.
- Test cases reviewed by peers for accuracy.

7.1.1.4. Test Cases:

- Detailed test cases for each function/method.
- Edge cases and boundary conditions covered.

7.1.1.5. Tools:

- Detailed test cases for each function/method.
- Edge cases and boundary conditions covered.

7.1.1.6. Schedule:

- Unit testing integrated into the development process.
- Continuous testing during code development.

7.1.2. Integration Testing Plan:

7.1.2.1. Objectives:

- Verify the interaction between integrated components.
- Identify and address interface issues.

7.1.2.2. Scope:

- Testing interactions between various modules.
- Assessing data flow between components.

7.1.2.3. Responsibilities:

- Testing team responsible for integration testing.
- Collaboration with developers for issue resolution.

7.1.2.4. Test Cases:

- Scenario-based test cases for component interactions.
- Data consistency and integrity checks.

7.1.2.5. Tools:

- Integration testing frameworks.
- Debugging tools for identifying integration issues.

7.1.2.6. Schedule:

- Integration testing performed after unit testing.
- Iterative testing during system development.

7.1.3. System Training Plan:**7.1.3.1. Objectives:**

- Validate the system as a whole.
- Ensure all components work seamlessly together.

7.1.3.2. Scope:

- End-to-end testing of the entire system.
- Verifying system functionalities and features.

7.1.3.3. Responsibilities:

- Dedicated testing team for system testing.
- Collaboration with developers and stakeholders.

7.1.3.4. Test Cases:

- Comprehensive test cases covering system functionalities.
- Usability testing for user interfaces.

7.1.3.5. Tools:

- Automated testing tools for system-wide testing.
- Load testing tools for performance evaluation.

7.1.3.6. Schedule:

- System testing performed after integration testing.
- Multiple test cycles to ensure comprehensive coverage.

7.1.4. Acceptance Testing Plan:**7.1.4.1. Objectives:**

- Validate if the system meets user requirements.
- Obtain approval from stakeholders.

7.1.4.2. Scope:

- Real-world scenarios to validate user expectations.
- User acceptance testing (UAT) sessions.

7.1.4.3. Responsibilities:

- Users and stakeholders actively involved in acceptance testing.
- Testing team facilitates and records results.

7.1.4.4. Test Cases:

- User scenarios replicated for testing.
- Verification of system against initial requirements.

7.1.4.5. Tools:

- Issue tracking tools for feedback collection.
- User feedback surveys.

7.1.4.6. Schedule:

- Acceptance testing performed after system testing.
- Iterative testing based on user framework.

USER MANUAL

User Manuals are manuals that enable the user of a system or application to understand the working of the system and help them to use them efficiently. It is usually written by a technical writer, although user guides are written by programmers, product or project managers, or other technical staff, particularly in smaller companies.

Follow below step to work with app:

Registration Page:

The screenshot shows the 'Sign Up' form for the JobPortal. At the top right, there are links for Home, Jobs, Browse, Login, and Signup. The form itself has fields for Full Name (containing 'patel'), Email (containing 'patel@gmail.com'), Phone Number (containing '8080808080'), and Password (containing 'patel@gmail.com'). Below these fields are radio buttons for 'Student' and 'Recruiter', a 'Profile' link, and a file upload field labeled 'Choose file No file chosen'. A large black 'Signup' button is at the bottom, and a link 'Already have an account? Login' is at the very bottom.

Fig 13 Registration page for student and recruiter

- This Registration page has 2 role's one is for jobseeker and another one is for Employer.

Login Page:

Fig 14 Login page for student and recruiter

- This Login page has 2 role's one is for Student and another one is for Recruiter.

Forget and reset password Page:

fig 15. forgot password Page for student and recruiter

fig 16. reset password Page for student and recruiter

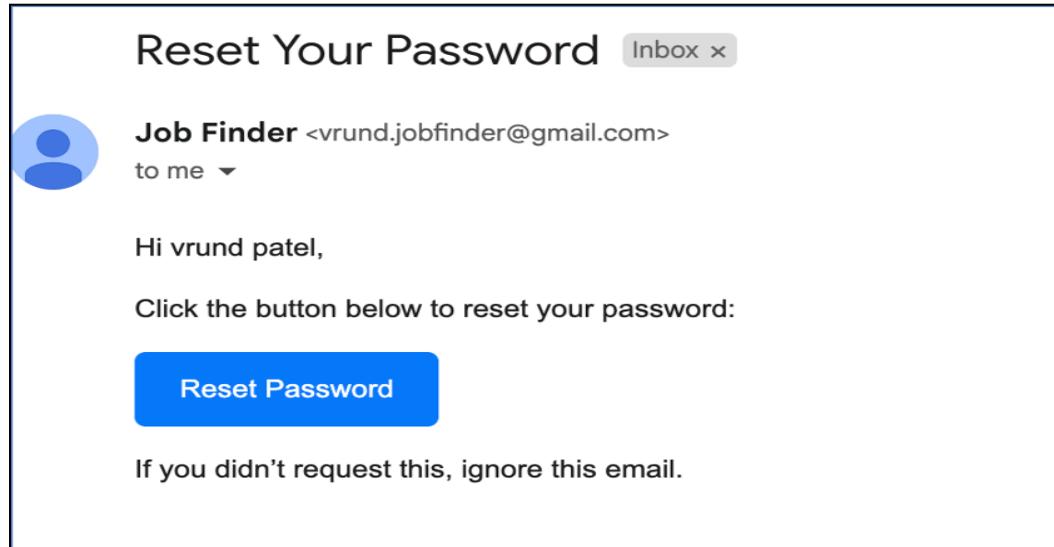


Fig 17. Received mail to reset the password

- This forgot password page and reset password page has 2 role's one is for Student and another one is for Recruiter which allows them to change the password.

Home Page:

JobPortal

No. 1 Job Hunt Website

Search, Apply & Get Your Dream Jobs

Frontend Developer Backend Developer Data Science

Latest & Top Job Openings

Paypal India Risk Analyst Monitor and assess transaction risks, implement fraud prevention strategies. 4 Positions Full Time 190000LPA	Netflix India UI/UX Designer Design engaging user interfaces and experiences for Netflix's streaming platform. 9 Positions Full Time 220000LPA	Google India Site Reliability Engineer Ensure reliable system uptime, manage alerts, and handle large-scale distributed systems. 1 Positions Full Time 300000LPA
---	---	---

Fig 18 Home page for student

- This Home page is for job seeker it contains some basic information of site and activity which job seeker role user can do such as view all job and my application.

Jobs Page:

The screenshot shows the 'JobPortal' interface. On the left, there's a sidebar with 'Filter Jobs' sections for Location (Delhi, Bangalore, Hyderabad, Pune, Mumbai), Industry (Frontend Developer, Backend Developer, FullStack Developer, Cloud Engineer, DevOps Engineer, Machine Learning Engineer, AI/ML Engineer, Data Scientist, Cybersecurity Engineer, Blockchain Developer, Game Developer, Mobile Developer, Product Manager, UI/UX Designer, QA Tester, Others), and Salary (0-40k, 42-1akh). The main area displays a grid of job cards. Each card includes the company logo, company name, job title, deadline, brief description, position count, full-time status, and salary. Buttons for 'Details' and 'Apply Now' are also present. The jobs listed are: Facebook Data Analyst (Deadline: 25/04/2025), Apple Security Analyst (Deadline: 10/05/2025), PayPal Risk Analyst (Deadline: 25/04/2025), Netflix UI/UX Designer (Deadline: 31/05/2025), Google Site Reliability Engineer (Deadline: 03/05/2025), and AMD Machine Learning Engineer (Deadline: 05/05/2025).

fig 19 Jobs page for student

- This all available jobs page is use by jobseeker role and employer role. They can get main information of job and if the user want more details then user can click on job details and also save their jobs.

Job details page:

The screenshot shows the 'Data Analyst' job details page. At the top, it shows the job title, 'Positions: Full Time', '400000LPA', and an 'Apply Now' button. Below that is a 'Job Description' section. Underneath is a table with job specifications: Role: Data Analyst, Location: Hyderabad, Description: Analyze user behavior to improve platform features and drive engagement., Experience: yrs, Salary: 400000LPA, Total Applicants: 0, and Posted Date: 2025-04-15.

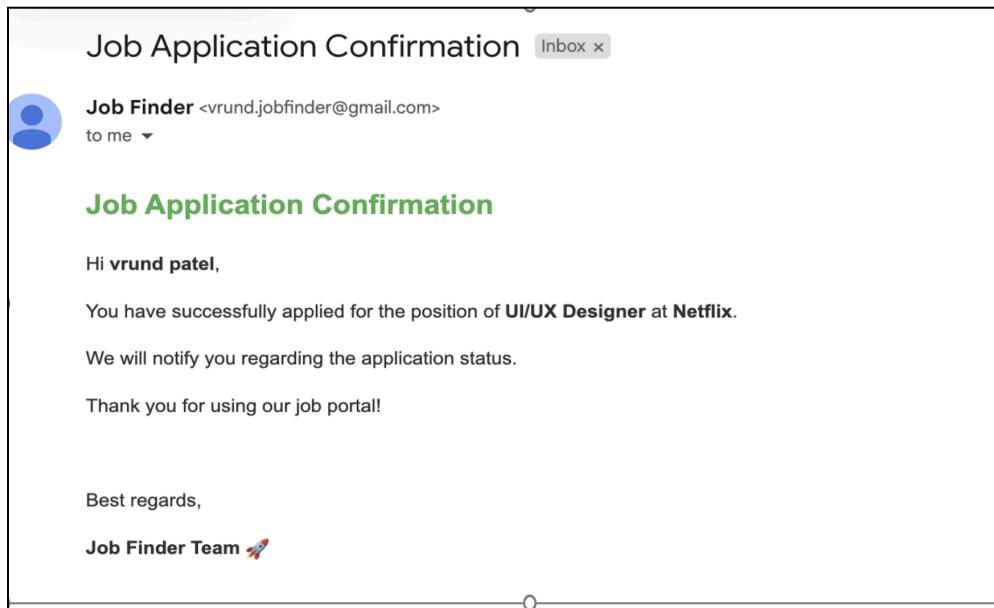


Fig 20-21 job details and received job confirmation mail

- This job details page is shown to user who has role jobseeker when jobseeker click on job details in all job page. If the user feel's interested in applying for this job then he can click on apply now and receives mail after applying for job.

Chatbot Support:

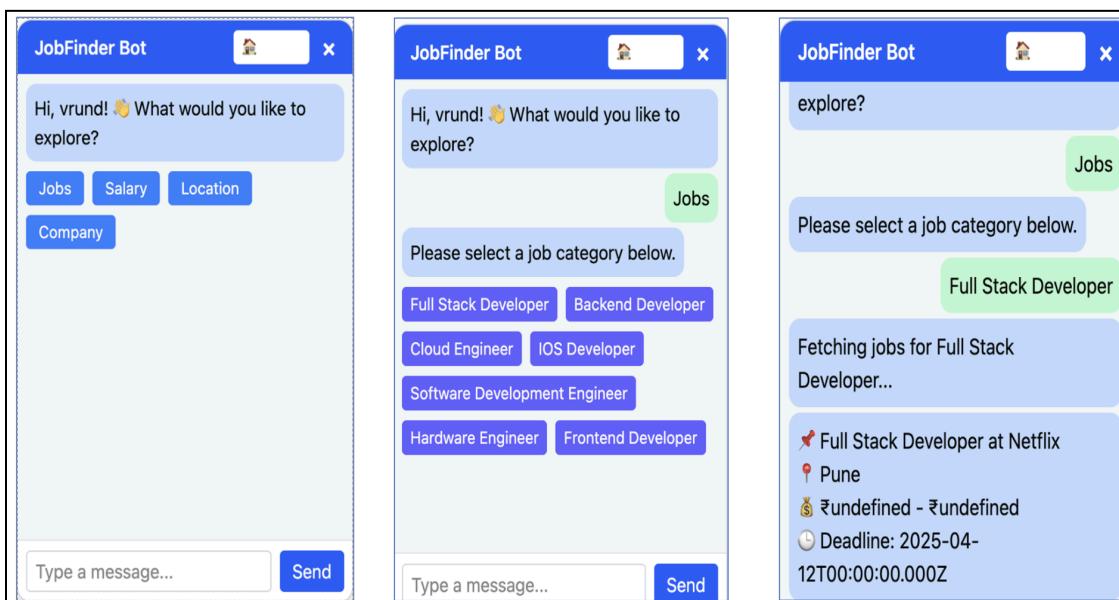


Fig 22 chatbot support

This page allows students to get jobs details according to jobs,salary,city for better user experience.

Browse page:

The screenshot shows the 'JobPortal' interface. At the top, there's a navigation bar with 'Home', 'Jobs', 'Browse', and a user profile icon. Below the navigation, the main content area is divided into two sections: 'Your Saved Jobs (2)' and 'All Jobs (12)'.

Your Saved Jobs (2):

- Facebook Hyderabad** (1 day(s) ago)

Data Analyst
Deadline: 25/04/2025
Analyze user behavior to improve platform features and drive engagement.
7 Position(s) Full Time 4.0 LPA

[Details](#) [Apply Now](#)
- Apple Pune** (1 day(s) ago)

Security Analyst
Deadline: 10/05/2025
Work on threat detection, system audits, and securing mobile ecosystems.
3 Position(s) Full Time 28.0 LPA

[Details](#) [Apply Now](#)

All Jobs (12):

- Paypal Bangalore** (1 day(s) ago)

Risk Analyst
Deadline: 25/04/2025
Monitor and assess transaction risks, implement fraud prevention strategies.
- Netflix Chennai** (1 day(s) ago)

UI/UX Designer
Deadline: 31/05/2025
Design engaging user interfaces and experiences for Netflix's streaming platform.
- Google Hyderabad** (1 day(s) ago)

Site Reliability Engineer
Deadline: 03/05/2025
Ensure reliable system uptime, manage alerts, and handle large-scale distributed systems.

Fig 23 student browse page

- Browse page allows user to see their saved jobs and redirecting page from home page after using searching service.

Student profile page:

Date	Job Role	Company	Status
2025-04-16	UI/UX Designer	Netflix	ACCEPTED
2025-04-15	Risk Analyst	Paypal	PENDING
2025-04-15	Cloud Engineer	Google	PENDING
2025-04-14	Backend Developer	Paypal	PENDING
2025-04-14			PENDING
2025-04-14			PENDING
2025-04-14			PENDING

Fig 25 student profile page

Allows user to view their details and resume and shows them for how many jobs they have applied and their current status.

Update student profile page:

Name	vrund patel
Email	vrundp254@gmail.com
Number	9265080027
Bio	
Skills	
Resume	Choose file No file chosen

Fig 26 update student profile page

- Allows students to update their current details.

Logout page for student and recruiter profile:

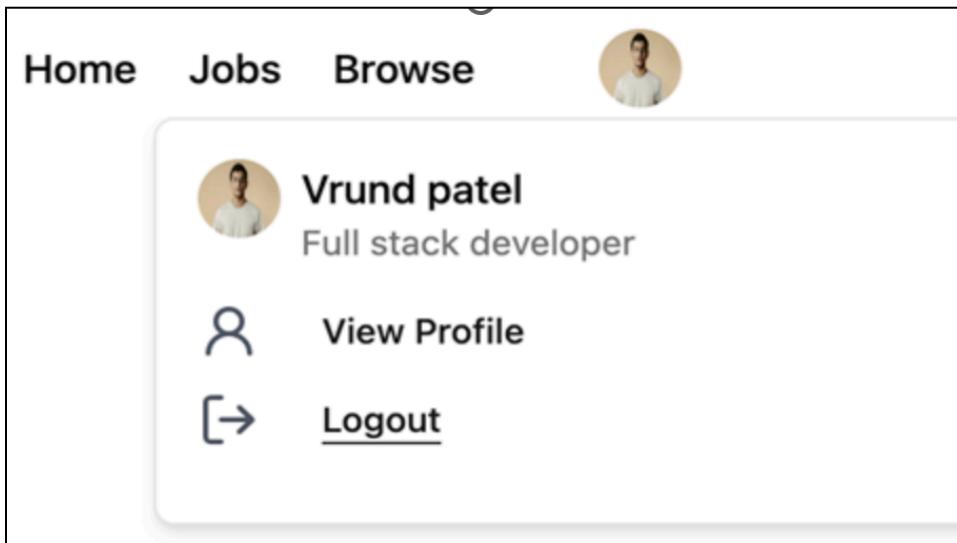


Fig 27 Logout page

Recruiter company page:

JobPortal			
			Companies Jobs
<input placeholder="Filter by name" type="text"/> New Company			
Logo	Name	Date	Action
	Amazon	2025-04-04	...
	AMD	2025-04-04	...
	Apple	2025-04-04	...
	Facebook	2025-04-04	...
	IBM	2025-04-04	...
	Microsoft	2025-04-04	...
	Paypal	2025-04-04	...
	Google	2025-04-04	...
	Netflix	2025-04-04	...
	amzn	2025-04-09	...
	snapchat	2025-04-10	...
	instagram	2025-04-11	...
	amd	2025-04-15	...

Fig 28 company page for recruiter page

- This Home page is for employer it contains some basic information of site and activity which employer role user can do such as view all job, see applicant's application, create new job's and view your job's.

Create New company page:

JobPortal

Companies Jobs

[← Back](#) Company Setup

Company Name	Description
Website	Location
Logo Choose file No file chosen	

Update

Fig 29 new company page

- The new company setup page allows recruiter to add new company.

All Jobs Page for recruiter:

JobPortal

Companies Jobs

Filter by name, role **New Jobs**

Company Name	Role	Date	Deadline	Action
Paypal	Risk Analyst	2025-04-15	25/04/2025	...
Netflix	UI/UX Designer	2025-04-15	31/05/2025	Edit Applicants
Google	Site Reliability Engineer	2025-04-15	03/05/2025	...
Facebook	Data Analyst	2025-04-15	25/04/2025	...
Apple	Security Analyst	2025-04-15	10/05/2025	...
AMD	Machine Learning Engineer	2025-04-15	05/05/2025	...
Amazon	DevOps Engineer	2025-04-15	18/04/2025	...
Netflix	Full Stack Developer	2025-04-14	12/04/2025	...
Paypal	Backend Developer	2025-04-14	09/05/2025	...
Google	Cloud Engineer	2025-04-14	30/04/2025	...
Apple	IOS Developer	2025-04-14	01/05/2025	...
Amazon	Software Development Engineer	2025-04-14	30/04/2025	...
AMD	Hardware Engineer	2025-04-14	20/04/2025	...
Facebook	Frontend Developer	2025-04-14	18/04/2025	...

Fig 30 All jobs page for recruiter

This all available jobs page is use by jobseeker role and employer role. They can get main information of job and if the user want more details then user can click on job details.

Edit Job page:

The screenshot shows the 'Edit Job' form with the following data:

Title	Risk Analyst	Description	Monitor and assess transaction risks, implement fraud prev.
Requirements	Risk modeling, SQL, Excel, Python	Salary	1900000
Location	Banglore	Job Type	Full Time
Experience Level	2	No. of Positions	4
Industry	QA Tester	Deadline	25/04/2025

Update Job

fig 31 edit jobs page

- Edit job Page allows recruiter to edit their existing jobs and post it.

Applicants page:

The screenshot shows the 'Applicants 1' section with the following data:

Full Name	Email	Contact	Resume	Date	Action
vrund patel	vrundp254@gmail.com	9265080027	vrund_resume.pdf	2025-04-14	...

A list of your recent applied users

Action buttons: Accepted, Rejected

Fig 32 view applicants page

- View Applicants page allows recruiter to view thier details and resume and recruiter can accept or reject job application on basis of their analysis.

9.LIMITATIONS AND FUTURE ENHANCEMENT

9.1. Limitations:

- Limited Job Listings: The availability of job listings is dependent on employers posting job openings on the platform. As a result, there may be periods with fewer job opportunities available to job seekers.
- Geographic Limitations: The job portal's reach may be limited to specific geographic regions or industries. Job seekers and employers outside of these regions or industries may find fewer relevant opportunities on the platform.
- Dependency on Internet Connectivity: Since the job portal is a web-based application, users require a stable internet connection to access its features. Limited or unstable internet connectivity may hinder users' ability to browse job listings or submit applications.
- Incomplete Information: Job listings and user application may sometimes lack comprehensive information. Employers may not provide detailed job descriptions, and job seekers may not fully complete their application, leading to ambiguity and uncertainty for both parties.
- Competition: Job seekers may face stiff competition from other applicants for desirable positions. The high volume of applicants for popular job listings can make it challenging for individual candidates to stand out.
- Limited Employer Engagement: Employers may not actively engage with job seekers or provide timely updates on the status of job applications. This lack of communication can lead to frustration and uncertainty for applicants.
- Security Concerns: As with any online platform, there are inherent security risks associated with sharing personal and sensitive information. While efforts are made to secure user data, the risk of data breaches or unauthorized access cannot be entirely eliminated.
- Technical Issues: Users may encounter technical issues such as website downtime, slow loading times, or glitches in the user interface. These technical challenges can disrupt the user experience and affect productivity.
- Compatibility: The job portal may not be fully compatible with all web browsers or devices. Users may experience formatting issues or limited functionality when accessing the platform from certain browsers or mobile devices.
- Lack of Guarantee: While the job portal aims to connect job seekers with employment opportunities, it cannot guarantee job placement or success in securing a job. The final hiring decision lies with the employers, and there is no assurance of job offers for applicants.

9.2. Future Enhancements:

- Mobile Application Development: Develop native mobile applications for iOS and Android platforms to extend accessibility and convenience for users who prefer mobile devices.
- Integration with Social Media Platforms: Enable users to import their professional profiles from platforms like LinkedIn, allowing for seamless registration and profile completion.
- Real-Time Notifications: Implement push notifications to alert users about new job listings matching their criteria, application status updates, and other relevant notifications.
- Skill Assessment Tools: Integrate skill assessment tests or quizzes to help job seekers evaluate their skills and receive recommendations for relevant job opportunities based on their assessment results.
- Enhanced Employer Dashboard: Provide employers with more advanced tools for managing job listings, including analytics dashboards to track applicant demographics, application status, and hiring metrics.
- Networking Features: Introduce networking functionalities such as chat or messaging systems to facilitate communication between job seekers and employers, allowing for more direct interaction and relationship-building.
- Expanded Geographic Reach: Expand the job portal's reach to include job listings from a broader range of geographic regions, industries, and sectors, providing users with more diverse employment opportunities.
- Video Interviewing Platform: Integrate a video interviewing platform to streamline the interview process, allowing employers to conduct remote interviews and evaluate candidates more efficiently.
- Integration with Learning Resources: Partner with online learning platforms to offer job seekers access to relevant courses and resources for skill development, career advancement, and professional growth.
- Feedback and Ratings System: Implement a feedback and ratings system where job seekers can rate their interview experiences and provide feedback on employers, helping to improve transparency and accountability in the hiring process.
- Accessibility Features: Ensure the job portal complies with accessibility standards and guidelines, making it accessible to users with disabilities and enhancing inclusivity.
- Keyword Search: Implement a keyword search functionality that allows users to enter keywords related to job titles, skills, industries, or company names. The system will then return job listings containing those keywords in their titles or descriptions.
- Advanced Filters: Provide users with a set of advanced filters to narrow down their job search results. Filters may include options such as location, industry, job type (e.g., full-time, part-time, remote), salary range, experience level, and company size.

10. CONCLUSION AND DISCUSSION

10.1. Conclusion:

- The development of our job portal marks a significant milestone in modernizing the job search and recruitment process. Through a user-centric approach and leveraging advanced technologies, we have created a platform that addresses the needs of both job seekers and employers, facilitating seamless discovery and efficient hiring processes.
- Our job portal provides a user-friendly interface for job seekers to explore a wide range of job opportunities, apply for positions, and manage their applications effortlessly. Employers, on the other hand, benefit from streamlined job posting, application management, and manage job functionalities.
- Throughout the development process, we have prioritized usability, scalability, and reliability, ensuring that our platform can accommodate the growing needs of users and businesses alike. By adhering to coding standards, implementing robust security measures, and optimizing performance, we have built a platform that delivers a superior user experience while maintaining data integrity and confidentiality.
- While our job portal offers valuable features and functionalities, we recognize that there is always room for improvement. Future enhancements may include the implementation of advanced job matching algorithms, enhanced search capabilities, and integration with external job databases to further expand the pool of job opportunities available to users.
- In conclusion, our job portal represents a comprehensive solution for modern job search and recruitment needs, empowering both job seekers and employers to connect and thrive in today's competitive job market. As we continue to evolve and innovate, we remain committed to providing a platform that fosters meaningful connections, facilitates efficient hiring processes, and ultimately contributes to the success of individuals and organizations alike.

10.2. Discussion:

In this section, we delve into a comprehensive analysis of various aspects of the job portal project, including self-analysis of project viabilities, encountered problems, and their solutions, as well as a summary of project work. Through a critical examination of these facets, we gain valuable insights into the project's strengths, challenges, and future prospects. Let's begin by exploring the self-

analysis of project viabilities, which sheds light on the project's technological, operational, legal, and social viability.

10.2.1 Self-Analysis of Project Viabilities:

The self-analysis of project viabilities reveals crucial insights into various aspects of the job portal project.

- Technological Viability: The project's utilization of established technologies such as React.js, Node.js, and MongoDB ensures a solid foundation for development and scalability.
- The efficient implementation of job discovery functionalities enhances the platform's technological viability, meeting user expectations for a seamless job discovery experience.
- Operational Viability: The effective management of user authentication and authorization underscores the platform's operational viability, ensuring secure access to features based on user roles.
- Employers benefit from streamlined job posting capabilities, contributing to operational efficiency in the recruitment process.
- Legal and Regulatory Viability: Adherence to data privacy laws and regulations demonstrates the project's commitment to safeguarding user data and mitigating legal risks associated with data handling.
- Social and Environmental Viability: The user-friendly interface promotes accessibility for users of all backgrounds, enhancing social viability.
- Efforts to minimize environmental impact through efficient resource utilization highlight the project's environmental viability.

10.2.2 Problem Encountered and Possible Solutions:

During the development of the job portal, several challenges were encountered, each requiring effective solutions to ensure the project's success. Two significant issues arose during the implementation phase: image uploading and email notifications.

1. Image Uploading Issue:

One of the primary challenges faced was the storage of job-related images, such as applicant resume which was in jpg and png format. MongoDB, the chosen database management system, does not support direct image storage. To address this issue, we integrated Cloudinary, a cloud-based image management solution, into the job portal. Cloudinary provides a seamless way to upload, store, and manage images, offering scalability and reliability. By leveraging Cloudinary's APIs, we were able to overcome the limitations of MongoDB and ensure efficient handling of image-related data in the job portal.

Another critical requirement of the job portal was the ability to send email notifications to job seekers regarding their job applications status. Implementing email functionality in Node.js posed a significant challenge, particularly in ensuring reliable delivery and handling of email templates. To tackle this challenge, we adopted Node Mailer, a popular npm module for sending emails using Node.js. Node Mailer simplifies the process of sending emails by providing a straightforward API and support for various email services. Additionally, we designed customizable email templates to deliver informative and visually appealing notifications to job seekers. By integrating Node Mailer into the job portal, we successfully implemented a robust email notification system, mainly used for applicant's application status updates.

3. Authentication and Session Management:

Securing user authentication and maintaining session integrity are critical aspects of any web application. Initially, we faced challenges in implementing a robust authentication mechanism that could securely manage user sessions across multiple requests. To address these challenges, we adopted JWT as a solution for user authentication and session management. JWT is a compact, URL-safe token format that securely encodes user information as JSON objects. Upon successful login, a JWT token is generated and sent to the client, where it is stored securely, typically in local storage or session storage.

10.2.3 Summary of Project Work:

The job portal project aimed to develop a comprehensive digital platform to facilitate efficient job discovery and recruitment processes. Key components such as user authentication, job discovery functionalities, and employer job posting were meticulously implemented to ensure a seamless user experience. Through collaboration, innovation, and adherence to best practices, the project team successfully delivered a robust and user-friendly platform that revolutionizes the job discovery and recruitment process. Moving forward, continuous improvement and enhancement efforts will be undertaken to further refine the platform and meet evolving user needs and industry standards.

REFERENCES

MongoDB: <https://www.mongodb.com/>

Node Js: <https://nodejs.org/eng>

React Js: <https://react.dev/>

Express <https://expressjs.com/>

Cloudinary: <https://cloudinary.com/>

Jwt: <https://jwt.io/>

EXPERIENCE

During my internship, I had the opportunity to contribute to the development of the **Job Finder** project—a modern job portal platform built using the MERN stack: MongoDB, Express.js, React.js, and Node.js. This project provided me with rich hands-on experience in full-stack web development, where I was involved in both frontend and backend tasks. It helped me gain a solid understanding of how different layers of a web application integrate to deliver a seamless user experience.

Through this project, I expanded my technical knowledge and sharpened my ability to design and implement practical solutions. I learned how to build scalable, responsive, and user-friendly applications while collaborating with a team. Working across the complete development lifecycle—from planning and coding to testing and deployment—gave me a better grasp of industry-standard development workflows.

Using technologies like React.js on the frontend helped me build dynamic and interactive user interfaces, while Node.js and Express.js enhanced my backend development skills. Integrating MongoDB taught me how to model and manage data efficiently using NoSQL databases. Throughout the project, I faced several real-world challenges that boosted my debugging and problem-solving skills.

Overall, this internship significantly enhanced my software development capabilities and introduced me to real-world practices such as Git-based version control, code reviews, API integration, and authentication/authorization mechanisms. It was a meaningful and enriching experience that has better prepared me for a professional career in full-stack development.