"PrepMinds: From Resume to Interview"

A Major Project Report Submitted to
Rajiv Gandhi Proudyogiki Vishwavidyalaya

Towards Partial Fulfillment for the Award of
Bachelor of Engineering in Computer Science Engineering

Submitted by:

Aryan Thapak (0827CS211046)
Anoushka Vyas (0827CS211022)
Anupam Raushan(0827CS211025)
Adarsh Trivedi (0827CS223D02)
Aman Mehra (0827CS201026)

Guided by:

Prof. Shraddha Sharma
Professor, CSE

Acropolis Institute of Technology & Research, Indore
July - Dec 2024

EXAMINER APPROVAL

The Major Project entitled **"Prepminds: From Resume to Interview"** submitted by **Aryan Thapak(0827CS211046), Anoushka Vyas (0827CS211022), Anupam Kumar Raushan(0827CS211025), Aman Mehra(0827CS201026), Adarsh Trivedi(827CS223D02)** has been examined and is hereby approved towards partial fulfillment for the award of **Bachelor of Technology degree in Computer Science Engineering** discipline, for which it has been submitted. It understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed, or conclusion drawn therein, but approve the project only for the purpose for which it has been submitted.

(Internal Examiner) **(External Examiner)**

Date: **Date:**

RECOMMENDATION

This is to certify that the work embodied in this major project entitled **"Prepminds: From Resume to Interview"** submitted by **Aryan Thapak(0827CS211046), Anoushka Vyas (0827CS211022), Anupam Kumar Raushan(0827CS211025), Aman Mehra(0827CS201026), Adarsh Trivedi(827CS223D02)** is a satisfactory account of the bonafide work done under the supervision of **Prof. Shraddha Sharma** , is recommended towards partial fulfillment for the award of the Bachelor of Technology (Computer Science Engineering) degree by Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal.

(Project Guide)

(Project Coordinator)

(Dean Academics)

STUDENTS UNDERTAKING

This is to certify that the major project entitled **"Prepminds: From Resume to Interview"** has developed by us under the supervision of **Prof. Shraddha Sharma**. The whole responsibility of the work done in this project is ours. The sole intension of this work is only for practical learning and research.

We further declare that to the best of our knowledge; this report does not contain any part of any work which has been submitted for the award of any degree either in this University or in any other University / Deemed University without proper citation and if the same work found then we are liable for explanation to this.

Aryan Thapak (0827CS211046)

Anupam Raushan (0827CS211025)

Adarsh Trivedi (0827CS223D02)

Anoushka Vyas (0827CS211022)

Aman Mehra (0827CS201026)

Acknowledgement

We thank the almighty Lord for giving me the strength and courage to sail out through the tough and reach on shore safely.

There are number of people without whom this project would not have been feasible. Their high academic standards and personal integrity provided me with continuous guidance and support.

We owe a debt of sincere gratitude, deep sense of reverence and respect to our guide and mentor **Prof. Shraddha Sharma,** Professor, AITR, Indore for his motivation, sagacious guidance, constant encouragement, vigilant supervision, and valuable critical appreciation throughout this project work, which helped us to successfully complete the project on time.

We express profound gratitude and heartfelt thanks to **Dr. Kamal Kumar Sethi**, Professor & Head CSE, AITR Indore for his support,suggestion, and inspiration for carrying out this project. I am very much thankful to other faculty and staff members of the department for providing me all support, help and advice during the project. We would be failing in our duty if do not acknowledge the support and guidance received from **Dr. S. C. Sharma**, Director, AITR, Indore whenever needed. We take opportunity to convey my regards to the management of Acropolis Institute, Indore for extending academic and administrative support and providing me all necessary facilities for project to achieve our objectives.

We are grateful to **our parent** and **family members** who have always loved and supported us unconditionally. To all of them, we want to say "Thank you", for being the best family that one could ever have and without whom none of this would have been possible.

Aryan Thapak (0827CS211046), Anupam Raushan(0827CS211025), Anoushka Vyas (0827CS211022), Adarsh Trivedi (0827CS223D02), Aman Mehra (0827CS201026))

Prepminds: From Resume to Interview

This project is submitted to Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal (MP), India for partial fulfillment of Bachelor of Engineering in Information Technology branch under the sagacious guidance and vigilant supervision of ***Prof. Shraddha Sharma***.

Prepminds: From Resume to Interview is an all-in-one job preparation platform designed to streamline the job search process for users. Recognizing the fragmentation in existing resources—where resume building, interview preparation, and skill assessments are scattered across multiple sites—Prepminds: From Resume to Interview brings these essential tools together. The platform offers a customizable resume builder, industry-specific interview resources, company-specific questions, and skill assessment modules, all supported by personalized progress tracking. Built using React.js, Node.js, and MongoDB, Prepminds: From Resume to Interview ensures a responsive, scalable, and secure user experience. Rigorous testing verified the platform's functionality, performance, and datasecurity. Initial deployment primarily supports technical roles, with plans to expand resources for non-technical fields, enhance personalization, and improve accessibility. By consolidating job preparation tools, Prepminds: From Resume to Interview reduces preparation time and increases user readiness, positioning itself as a valuable, inclusive asset for job seekers in today's competitive market..

Key words: Job Preparation Platform, Resume Builder, Interview Preparation, Skill Assessment, Company-Specific Questions, Personalized Guidance, Progress Tracking.

"In a world of open messages, true security lies in what cannot be seen. To protect what matters, let us not only encrypt—but conceal."

List of Figures

Figure 3-1: Block Diagram 4

Figure 3-2: Data Flow Diagram Level 0 5

Figure 3-3: Data Flow Diagram Level 1 5

Figure 3-4: Data Flow Diagram Level 2 15

Figure 3-5: Entity Relationship (ER) Diagram 18

Figure 3-6: Activity Diagram 19

Figure 3-7: Sequence Diagram 21

Figure 3-8: Use Case Diagram 25

Figure 3-9: Screenshot 1 25

Figure 4-1: Screenshot 2 26

Figure 4-2: Screenshot 3 26

Figure 4-3: Screenshot 4 28

Figure 4-4: Screenshot 5 28

Figure 4-5: Screenshot 6 29

List of Tables

Table 4.1: Database Structure 20

Table 4.2: Test Case 1 31

Table 4.3: Test Case 2 32

Table 4.4: Test Case 3 33

Table 4.5: Guide Interaction Sheet 40

Table of Contents

CHAPTER 1. INTRODUCTION 1

1.1 Overview 1
1.2 Background and Motivation 1
1.3 Problem Statement and Objectives 2
1.4 Scope of the Project 3
1.5 Team Organization 3
1.6 Report Structure 4

CHAPTER 2. REVIEW OF LITERATURE 6

2.1 Preliminary Investigation 6
2.1.1 Current System 6
2.2 Limitations of Current System 7
2.3 Requirement Identification and Analysis for Project 8
2.4 Conclusion 10

CHAPTER 3. PROPOSED SYSTEM 11

3.1 The Proposal 11
3.2 Benefits of the Proposed System 11
3.3 Block Diagram 13
3.4 Feasibility Study 14
3.4.1 Technical 14
3.4.2 Economical 14
3.4.3 Operational 15
3.5 Design Representation 15
3.5.1 Data Flow Diagrams 15
3.5.2 Activity Diagram 17
3.5.3 Entity Relationship Diagram 18
3.5.4 Sequence Diagram 19
3.5.5 Use Case Diagram 20
3.5.6 Database Structure 20

CHAPTER 4. IMPLEMENTATION 22

4.1 Technique Used 22
4.1.1 Test Driven Development 22
4.1.2 Code Reviews 22
4.1.3 Modular Design 22
4.1.4 Documentation 23
4.2 Tools Used 23
4.2.1 Visual Studio Code 23
4.2.2 Git 24
4.2.3 GitHub 24
4.3 Language Used 25
4.3.1 Frontend Technologies 25
4.3.2 Backend Technologies 26
4.3.3 GitHub 26
4.4 Screenshots 27
4.5 Screenshots 25
4.6 Testing 29
4.6.1 Strategy Used 29
4.6.2 Test Case and Analysis 30

CHAPTER 5. CONCLUSION 34

5.1 Conclusion 34
5.2 Limitations of the Work 34
5.3 Suggestion and Recommendations for Future Work 36

BIBLIOGRAPHY 37

PROJECT PLAN 38

GUIDE INTERACTION SHEET 39

APPENDIX A-SOURCE CODE 41

APPENDIX B-STEGASAFE USER MANUAL 50

APPENDIX C-RESEARCH PAPER WITH CERTIFICATES 55

Chapter 1: Introduction

Introduction

Job seekers today face a fragmented landscape of disjointed platforms for resume building, interview prep, and skill assessment. Prepminds: From Resume to Interview addresses this challenge by offering a unified, comprehensive platform. This single solution streamlines the entire job search process, providing expert guidance, industry-relevant quizzes, and company-specific interview questions. By empowering job seekers with personalized support and confidence-boosting tools, Prepminds: From Resume to Interview significantly enhances their chances of landing their ideal roles.

1.1 Overview

The Prepminds: From Resume to Interview project addresses this needby creating a holistic platform designed to simplify and enhance the job search journey. The platform combines a range of tools, such as resume building, interactive interview preparation, and access to targeted practice materials, to equip users with everything they need to secure their desired job roles.

The Prepminds: From Resume to Interview project addresses this need by creating a holistic platform designed to simplify and enhance the job search journey. The platform combines a range of tools, such as resume building, interactive interview preparation, and access to targeted practice materials, to equip users with everything they need to secure their desired job roles.

1.2 Background and Motivation

As job markets grow more competitive, candidates must continually refine their skills and prepare comprehensively for the recruitment process. Existing platforms like NovoResume and LeetCode offer specific services, suchas resume creation or coding exercises, but they lack unified, customizableapproach that covers all phases of job preparation. Job seekers often have to switch between multiple platforms, resulting in fragmented preparation and

increased stress.

Prepminds: From Resume to Interview is motivated by the desire tobridge this gap by creating a centralized platform that offers integrated tools to support candidates through every stage of their job search. By combining these tools in one place, Prepminds: From Resume to Interview aims to streamline the preparation process, save time, and help candidates feel more confident and well-prepared for job opportunities.

The motivation for this project stems from a need to address the challenge of hiding information effectively without compromising on image quality. Traditional methods often result in detectable alterations to the image, making the steganography vulnerable to detection. With *StegaSafe*, the goal is to provide a solution that can hide data within images without significantly altering their appearance, making it an ideal solution for secure communication.

1.3 Problem Statement and Objectives

Job seekers often struggle to find a single platform that provides comprehensive job search tools, such as resume building, interview preparation, and company-specific practice questions. This lack of integration increases preparation time and reduces the effectiveness of job search efforts.

The objectives of the Prepminds: From Resume to Interview platform are:

1. **Objective 1:** To develop an all-in-one job preparation toolkit that integrates resume building, interview preparation, and company- specific question banks.

2. **Objective 2:** To provide personalized resources and guidance, helping users succeed at every step of their job search journey, from initial application to final interviews.

3. **Objective 3**:To enhance job seekers' preparedness, improve user experience, and increase their chances of securing their desired roles in a competitive market.

1.4 Scope of the Project

The scope of Prepminds: From Resume to Interview includes the development of an integrated, user- friendly platform that addresses the needs of job seekers at each stage of the job search process. The platform will feature:

- **Resume Builder:** Tools for generating customizable, professional resumes with industry-specific templates.
- Interview Preparation: Resources for both technical and non-technical interview preparation, including coding challenges, behavioral questions, and industry-specific quizzes.
- Company-Specific Question Repository: A database of real questions previously asked by companies, allowing users to tailor theirpreparation to specific employers.
- Personalized Insights: Progress tracking and recommendations based on user performance, helping users focus on areas that need improvement.

1.5 Team Organization

- **Aryan Thapak :**
  Focused on planning of the Prepminds: From Resume to Interview platform and worked on initialization as well as implementation of the project, he also investigated and found the right technology and studied in deep about it.
- **Anoushka Vyas:**
  Discussed various implementation processes to be carried out and diagrams and documentations to be created. Ensured proper working of the Prepminds: From Resume to Interview by carrying out efficient testing methodologies.
- **Anupam Kumar Raushan:**
  Assisted with image processing techniques, ensuring that the changes to the image were imperceptible to the human eye. Conducted testing and performance evaluations of the system to ensure robustness and

reliability.

- **Adarsh Trivedi:**
  Investigated and worked on the requirement analysis, laying out various goals to be achieved and documented them. Implementation logic forthe project objective and coding of internal functionalities was also done by him.
- **Aman Mehra:**
  Focused on completion of the project documentation of the complete system. Along with ensuring proper documentation, he also worked on various diagrams to effectively represent project functionalities.

1.6 Report Structure

The report on Prepminds: From Resume to Interview - Job Search and Preparation Platform is divided into six primary chapters, each focusing on a distinct aspect of the project. Below is an outline of the structure and contents of each chapter:

Chapter 1: Introduces the background and significance of the problem, followed by the rationale for developing the Prepminds: From Resume to Interview platform. This chapter details the project objectives, scope, and applications, as well as team organization and contributions. It concludes with a structured outline of the report.

Chapter 2: Provides a comprehensive review of existing systems relevant to job search and preparation. This includes an analysis of platforms such as NovoResume, Indeed, Glassdoor, LeetCode, and Zety, examining their advantages, limitations, and the gaps Prepminds: From Resume to Interview seeks to fill. The chapter concludes with requirements identified for developing Prepminds: From Resume to Interview.

Chapter 3: Begins with an overview of the Prepminds: From Resume to Interview platform, detailing its objectives and benefits. The chapter includes a block diagram of the system architecture, feasibility studies (technical, economic, and operational), and design representations such as data flow

diagrams and database structures. The deployment requirements in terms of hardware, software, and technical dependencies are also specified.

Chapter 4: Describes the development of the Prepminds: From Resume to Interview platform, including details of the tools, techniques, and technologies employed. This chapter covers the front-end (React.js), back-end (Node.js, Express.js), and database (MongoDB) frameworks, along with screenshots and functionality of each interface. Key components like the resume builder, quiz module, and company-specific question repository are presented.

Chapter 5: Summarizes the Prepminds: From Resume to Interview platform's impact on job seekers, highlighting its contributions to streamlined job preparation. Limitations encountered during development are discussed, followed by recommendations for future improvements, such as integration with job boards, additional question repositories, and AI-driven personalized recommendations.

Chapter 2: Review of Literature

Review of Literature

As job markets grow increasingly competitive, job seekers require comprehensive support to succeed at each stage of the application and hiring process. However, most available platforms offer only partial solutions, focusing primarily on one or two aspects of job preparation. This literature review examines existing job search and preparation systems to identify gaps that limit their effectiveness as standalone resources. Understanding these limitations provides a foundation for the design and development of Prepminds: From Resume to Interview, an integrated platform that addresses job seekers' needs across resume building, interview preparation, and skill assessments. This chapter evaluates several popular platforms, highlighting their core features, advantages, and limitations. Through this analysis, we aim to identify specific areas where current systems fall short, thus establishing the rationale for developing an all-encompassing solution like Prepminds: From Resume to Interview.

2.1 Preliminary Investigation

2.1.1 Current System

The current landscape of job preparation platforms includes several popular systems that offer distinct functionalities, such as resume building, job listings, interview preparation, and company insights. However, each of these platforms is generally limited in scope, focusing on specific aspects of the job search process rather than providing a holistic approach. Below is an evaluation of the primary systems in use today and their shortcomings, which Prepminds: From Resume to Interview aims to address.

2.2 Limitations of Current System

The current job preparation platforms reviewed each offer specific features that are beneficial for certain aspects of job searching. However, their limitations prevent them from providing a complete, end-to-end solution for job seekers. Below are the main limitations identified in the existing systems.

- **Limited Scope and Fragmented Functionality**: Basic methods like LSB modification are often susceptible to steganalysis techniques, which can reveal the presence of embedded data by detecting statistical anomalies in the image.
- **Lack of Comprehensive Interview Preparation**: Most platforms lack structured tools for interview preparation beyond technical coding exercises (LeetCode) or employee- shared interview experiences (Glassdoor). There is an absence of robust, interactive resources that cover both technical and non-technical interview preparation, including mock interviews, practice questions, and behavioral interview guides.
- **Absence of Skill Assessment and Practice Resources**: Platforms like Indeed and NovoResume do not offer skill assessments or practice quizzes, which are essential for gauging readiness and improving specific job-related skills. Job seekers are therefore unable to evaluate their knowledge gaps or track their progress within the same platform they use for resume building or job search.
- **Lack of Personalization and Tailored Preparation:** Current platform do not provide personalized recommendations or insights based on user progress and performance. This absence of customization reduces the effectiveness of preparation, as job seekers do not receive targeted guidance that could help them improve in specific areas relevant to their career goals.
- **Dependency on Multiple Platforms:** Job seekers must frequently switch between platforms to complete different stages of their preparation. For instance, they may use one

platform for resume creation, another for technical preparation, and yet another for job search. This dependency on multiple systems can lead to inefficiencies, increased preparation time, and a fragmented user experience.

2.3 Requirement Identification and Analysis for Project

The initiation of the Prepminds: From Resume to Interview project represents a strategic and dedicated effort to address critical gaps in the current landscape of job preparation resources available to job seekers. In today's highly competitive employment environment, candidates often face an overwhelming number of obstacles across various stages of the job search process—from building an effective resume to acing final interviews. Conventional platforms commonly address only a narrow aspect of this process, such as offering basic resume templates, limited interview practice, or generic quizzes, leaving candidates to navigate multiple tools and resources to achieve comprehensive readiness. Prepminds: From Resume to Interview, therefore, was conceptualized as a unified platform that offers an end-to-end solution, consolidating all necessary resources and tools to aid job seekers in a single, integrated platform.

Prepminds: From Resume to Interview is designed to provide a holistic solution by bringing together three essential pillars of job preparation—resume building, interview preparation, and skill assessment—each with targeted features aimed at streamlining the job search journey and enhancing candidates' confidence and competitiveness. This all-in-one approach aims to minimize the fragmentation experienced by job seekers, thereby maximizing efficiency, consistency, and effectiveness in preparation. Prepminds: From Resume to Interview will also offer customized guidance and a user-friendly interface tailored to meet the varied needs of candidates across different career stages and industries.

The development of Prepminds: From Resume to Interview requires astructured approach to requirements gathering, as each feature and

functionality must contribute to the platform's overall goal of comprehensive job preparation. To achieve this, Section 2.3 outlines and analyzes the key requirements essential to Prepminds: From Resume to Interview' success. These requirements are organized into five main categories:

- **Stakeholders**: The project is centered around diverse stakeholders, including job seekers, hiring managers, and recruiters, each influencing specific functional needs of the platform. Based on a combination of user interviews and in-depth market research, key requirements were identified. Job seekers need a platform that offers a unified toolkit for resume building, interview preparation, and practice questions, while recruiters and hiring managers emphasize the importance of accurate resume formatting and skill assessment tools.
- **Functional Requirements**: Prepminds: From Resume to Interview is designed to offer a comprehensive, user-friendly platform that supports job seekers throughout the hiring process. Key features include resume generation with customizable templates, an interview preparation module with company-specific questions and mock interviews, domain- specific practice quizzes, profile management to track progress, and advanced search and filter options for tailored resource access. Users can also export resumes as PDFs and save quiz results for future reference.
- **Non-Functional Requirements**: Prepminds: From Resume to Interview must be designed to be performant, secure, scalable, and user- friendly. The platform should load quickly, especially when accessing large datasets. Sensitive user information must be securely stored and compliant with data protection regulations. To accommodate growth, the platform must be scalable. Finally, a simple, intuitive interface is crucial for a positive user experience.
- **Technical Requirements**: Prepminds: From Resume to Interview utilizes a robust tech stack to ensure a performant, secure, and scalable platform. The frontend is built with HTML, CSS, and React, while the backend is powered by Node.js and Express. MongoDB is used for

efficient data management, and HTTPS and JWT are implemented for secure communication and user authentication.

- **Document Requirements:** Comprehensive documentation is essential for both the development and ongoing maintenance of Prepminds: From Resume to Interview. This includes technical documentation, which provides detailed insights into code structures, API documentation, and database schemas to support developer collaboration. Additionally, user documentation is vital, guiding users on platform functionalities like using the resume builder and accessing quizzes, thus ensuring an accessible and user-friendly experience.
- **Timeline and Milestones:** Following Agile methodologies, the development of Prepminds: From Resume to Interview is organized into iterative phases, each marked by specific milestones. This approach allows for continuous improvement and adjustment based on stakeholder feedback, ensuring the platform meets user expectations and market demands.
- **Testing Requirements**: Quality assurance is paramount in the development of Prepminds: From Resume to Interview. Rigorous testing requirements, including unit testing and user acceptance testing, are implemented to validate the functionality, performance, and user experience of the platform. Through these testing protocols, we ensure that the final product not only meets but exceeds stakeholder expectations.

2.4 Conclusion

Through these meticulously crafted requirements, Prepminds: From Resume to Interview is positioned to offer a transformative solution for job seekers. The platform's commitment to addressing each stage of the job preparation journey ensures a comprehensive, user-centric experience that aligns with industry demands.

Chapter 3: Proposed System

Proposed System

3.1 The Proposal

The proposed system, Prepminds: From Resume to Interview, aims to revolutionize the job preparation process by providing a comprehensive, integrated platform designed to support job seekers through each phase of their journey. Unlike existing platforms that specialize in only one or two aspects of job preparation—such asresume building, technical practice, or job searching—Prepminds: From Resume to Interview consolidates essential tools and resources into a single, user-friendly solution.

Prepminds: From Resume to Interview will incorporate a variety of functionalities, including a customizable resume builder, interactive skill assessments, company-specific interview question banks, and personalized progress tracking. The platform will offer tailored guidance, enabling users to strengthen their job search skills in areas where they need the most improvement. This holistic approach ensures that jobseekers can prepare effectively for every step, from initial application through final interviews, within one seamless interface.

The platform is designed to be responsive and accessible on multiple devices, allowing users to prepare anytime, anywhere. To provide a secure and personalized experience, Prepminds: From Resume to Interview will utilize advanced encryption and secure data storage to protect user information, while an intuitive interface will simplify navigation and encourage user engagement. By addressing the limitations of current systems, Prepminds: From Resume to Interview will become an essential tool for job seekers, delivering an all-encompassing experience that empowers them to compete confidently and successfully in today's job market.

3.2 Benefits of the Proposed System

The proposed system, Prepminds: From Resume to Interview, offers multiple benefits by providing job seekers with an all-in-one solution tailored to meet their preparation needs. The primary benefits of Prepminds: From Resume to

Interview are outlined below:

- **Integrated Job Preparation** Prepminds: From Resume to Interview eliminates the need for multiple platforms by integrating resume building, interview preparation, and skill assessment tools in one place. This streamlined approach saves users time and effort, enabling them to focus on improving their readiness without the hassle of switching between different systems.
- **Personalized Guidance and Feedback**: The platform offers personalized recommendations and progress tracking, allowing users to identify strengths and areas for improvement. This targeted guidance improves users' chances of success by helping them focus on the skills and knowledge areas most relevant to their career goals.
- **Company-Specific Preparation**: With its company-specific question repository, Prepminds: From Resume to Interview provides users with valuable insights into real interview questions asked by employers. This feature enables users to tailor their preparation for specific companies, enhancing their confidence and competitiveness in interviews.
- **Accessible and User-Friendly Design:** The platform is designed with a responsive, intuitive interface that works seamlessly across devices. This ensures that job seekers can access Prepminds: From Resume to Interview anytime, anywhere, maximizing convenience and user engagement.
- **Enhanced Data Security and Privacy**: Prepminds: From Resume to Interview prioritizes user data security by implementing encryption and secure storage practices. Compliance with data protection standards ensures that users' personal and sensitive information is safeguarded, fostering trust and reliability.
- **Efficient Skill Development:** The skill assessment and practice modules provide users with a clear understanding of their abilities and track their growth over time. This self-assessment feature motivates users to continuously develop their skills and improve their job readiness.

3.3 Block Diagram

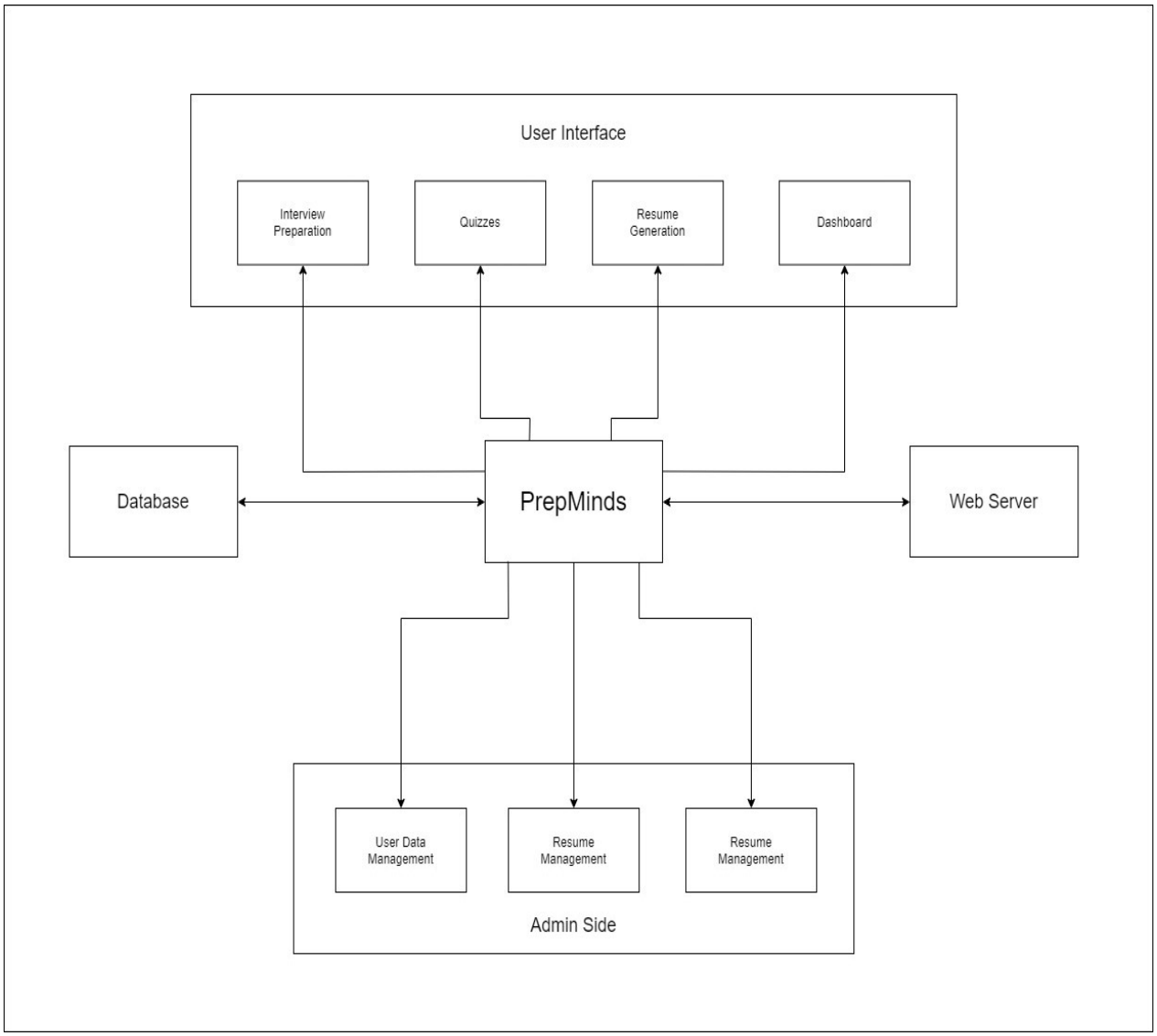


Figure 3-1 : Internal Process

3.4 Feasibility Study

The feasibility study for Prepminds: From Resume to Interview assesses the project's viability from technical, economic, and operational perspectives. Each aspect is evaluated to ensure that the platform can be developed, implemented, and maintained effectively while meeting user needs.

3.4.1 Technical

The technical feasibility of Prepminds: From Resume to Interview examines the availability of resources, tools, and expertise required for development. The platform will utilize a technology stack that includes React.js for frontend development, Node.js and Express.js for backend support, and MongoDB for database management. These technologies are widely used and provide robust support for building scalable, secure, and responsive web applications. The development team possesses proficiency in these tools, ensuring efficient implementation and long-term maintenance.

Moreover, the modular design approach will enable future scalability, allowing the platform to accommodate additional features and users as demand grows. The integration of third-party APIs for services like email notifications and analytics further enhances the platform's functionality.

3.4.2 Economical

Economic feasibility assesses the financial aspects of developing and maintaining Prepminds: From Resume to Interview. The primary costs include development, deployment, and regular maintenance expenses. Using open- source technologies such as React.js, Node.js, and MongoDB reduces initialdevelopment costs, while cloud-based solutions for deployment (e.g., AWS or Google Cloud) provide scalable hosting at manageable rates.

The projected returns from Prepminds: From Resume to Interview include increased user engagement and subscription-based revenue. By offering premium features, the platform can generate recurring income that will

support future expansions and maintenance, making it a financially sustainable project over the long term.

3.4.3 Operational

Operational feasibility examines the platform's ability to integrate seamlessly into job seekers' preparation routines and fulfill user expectations. Prepminds: From Resume to Interview is designed to provide a user-friendly, intuitive experience, enabling users to easily navigate through features such as resume building, interview preparation, and skill assessment. The responsive design ensures that users can access the platform on desktops, tablets, and mobile devices, fitting seamlessly into their schedules and enhancing accessibility. To support user adoption, a straightforward onboarding process and interactive guides will be provided, minimizing the learning curve. Additionally, Prepminds: From Resume to Interview will offer reliable customer support, helping users make the most of the platform's features. With these considerations, the platform is expected to meet the operational needs of job seekers effectively.

3.5 Design Representation

3.5.1 Use-Case Diagram

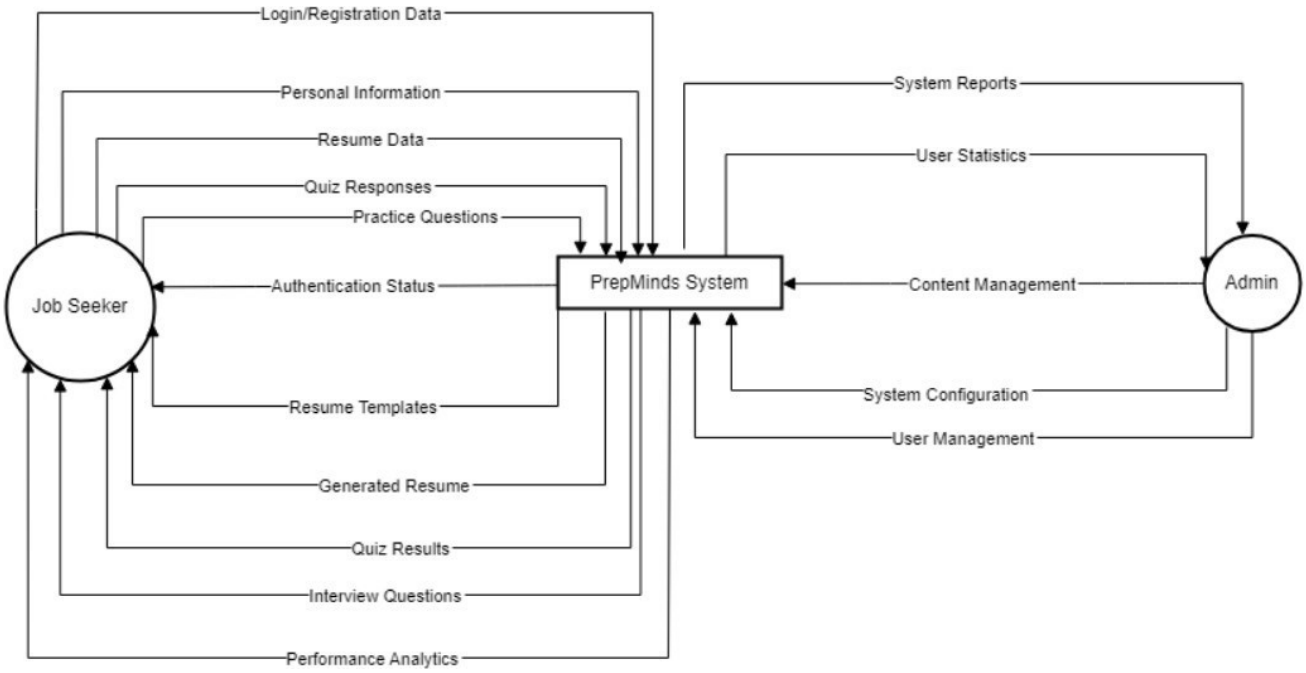


Fig 3-2: Data Flow Diagram Level 0

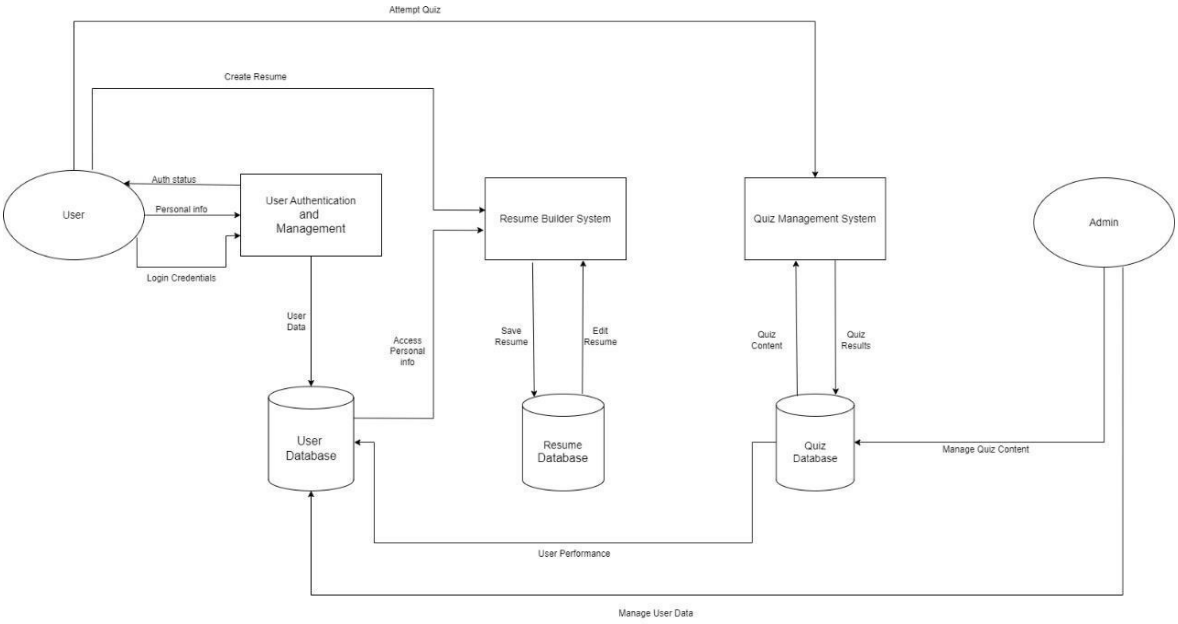


Figure 3-3 : Data Flow Diagram Level 1

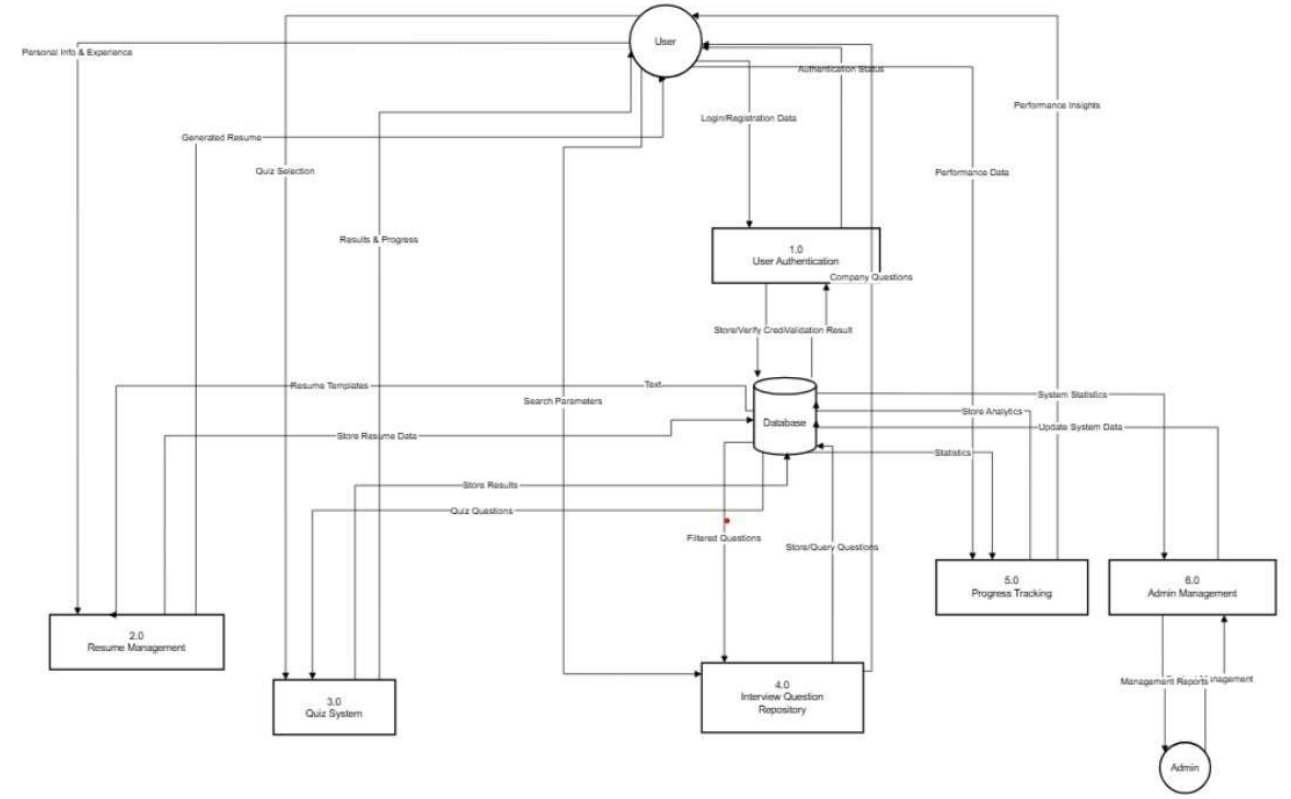


Figure 3-4 : Data Flow Diagram Level 2

3.5.2 Activity Diagram

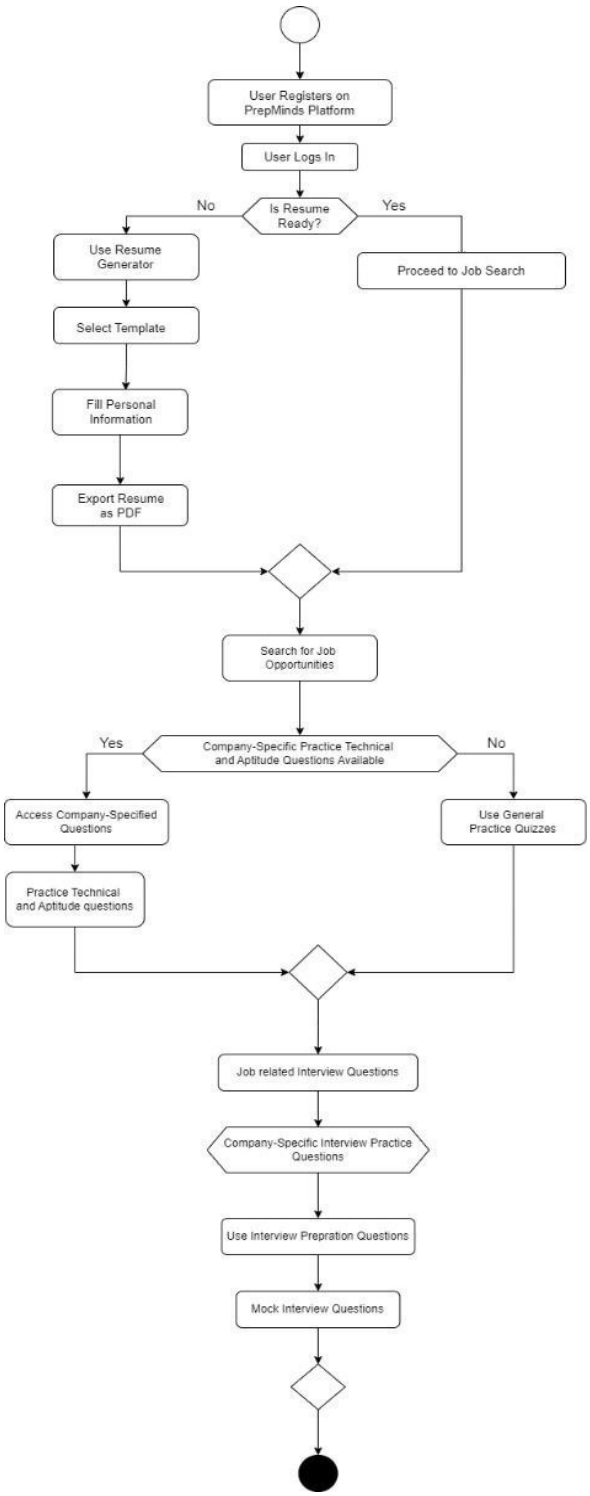


Fig 3-5: Activity Diagram

3.5.3 ER Diagram:

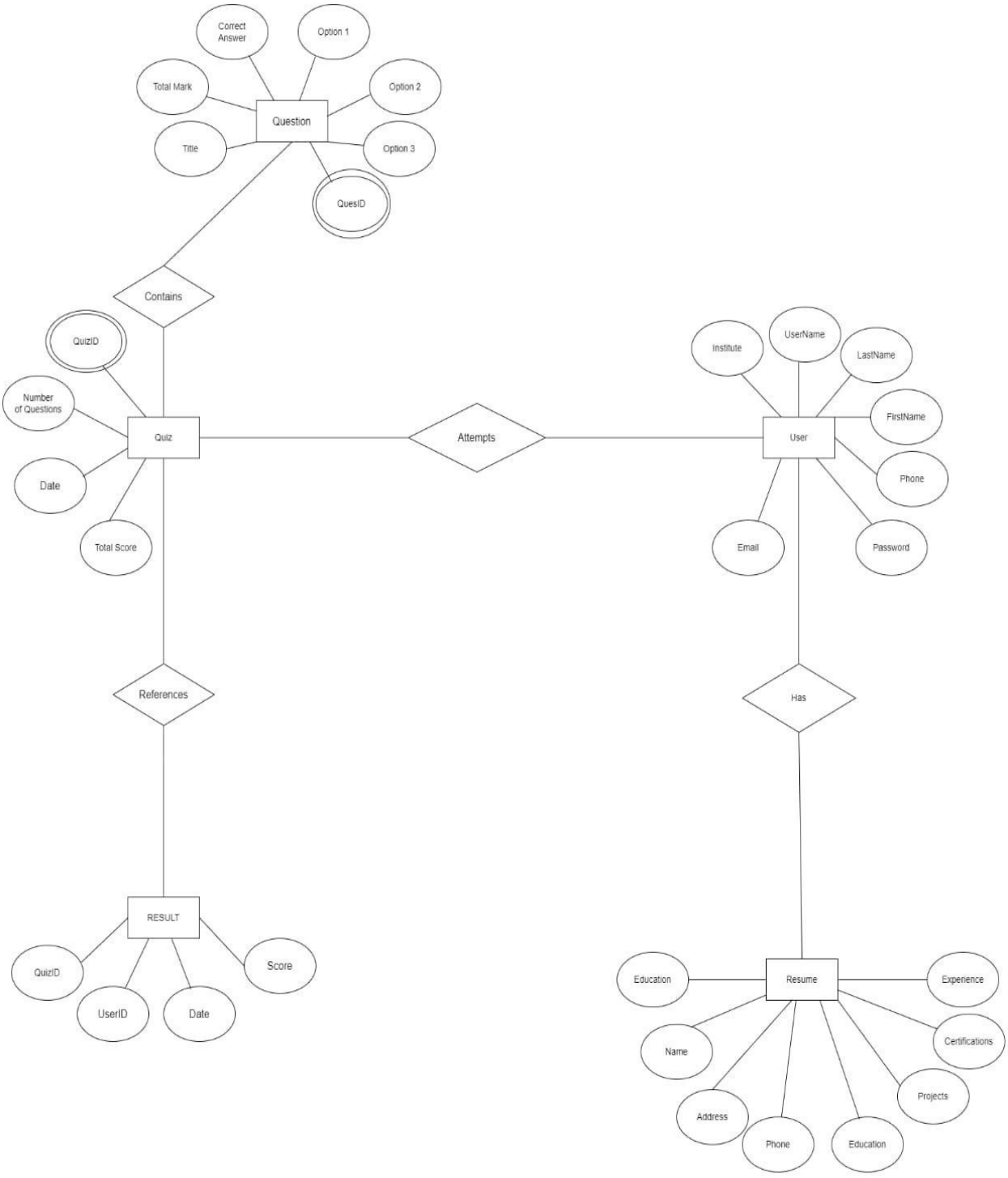
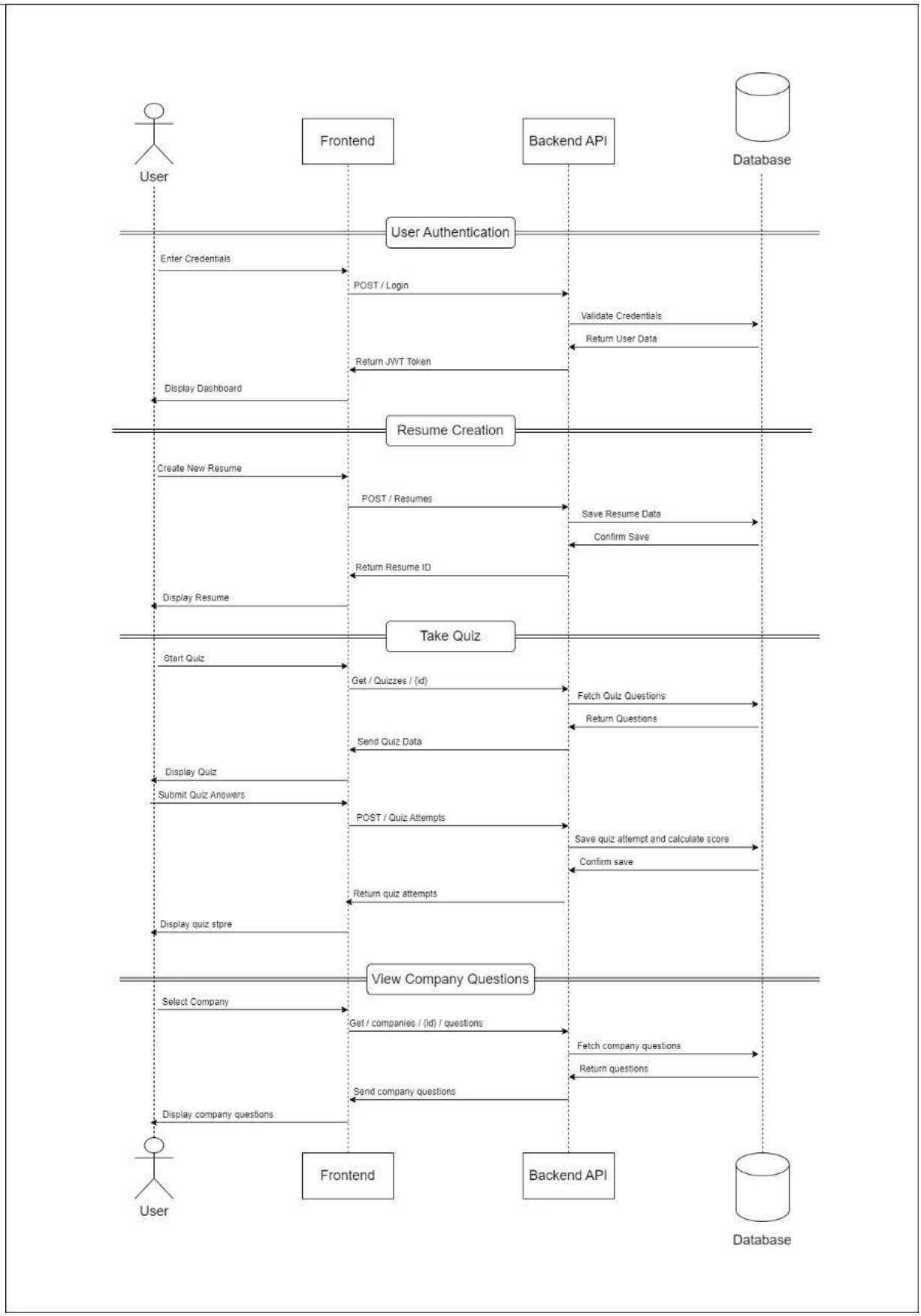


Figure 3-5: ER Diagram

3.5.4 Sequence Diagram:



3.5.5 Figure 3-7: Sequence Diagram

3.5.6 Use Case Diagram:

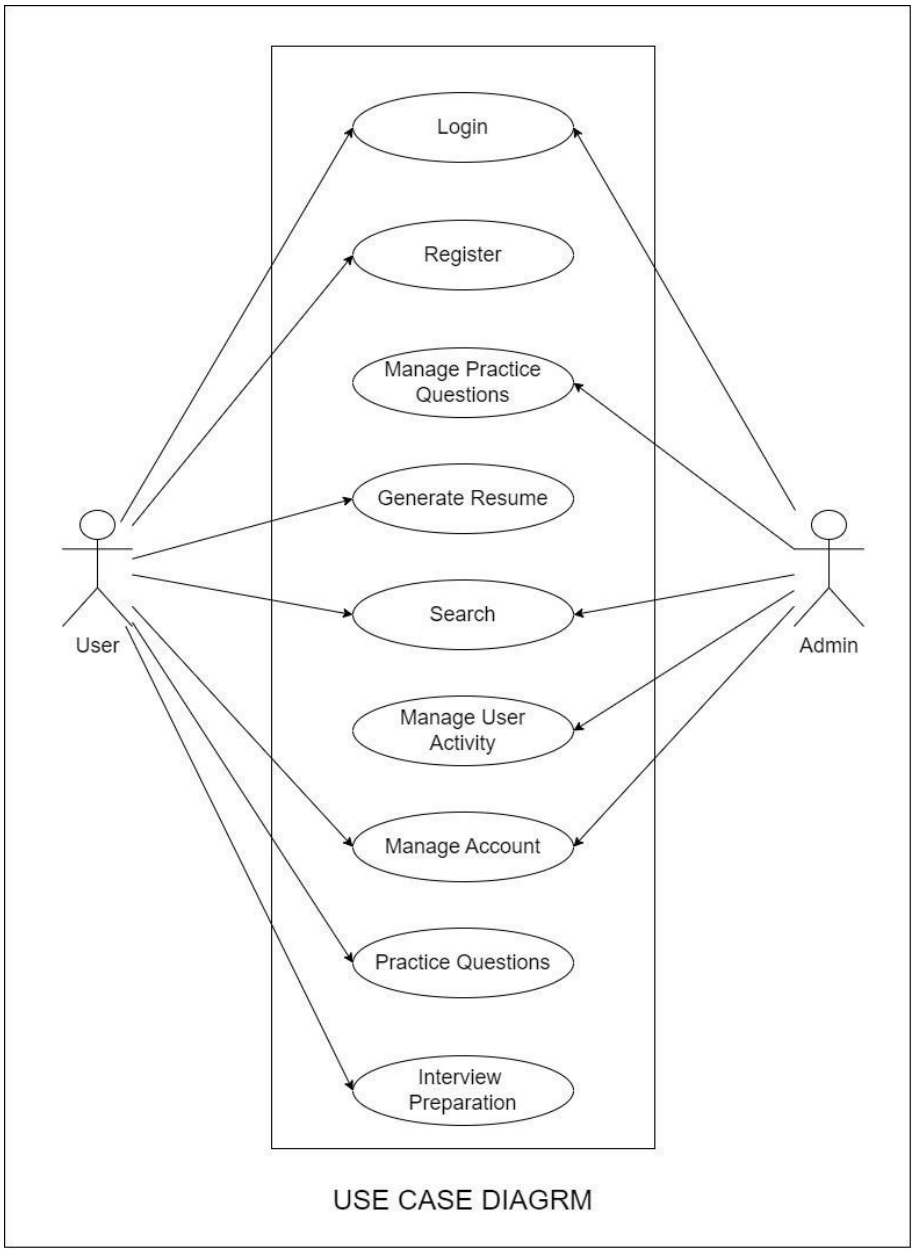


Figure 3-8 : Use Case Diagram

3.5.7 Database Structure:

Table Name	Description
QuestionModel.js	Defines the schema for storing quiz questions, including fields for title, options, and correct answer.
QuizModel.js	

	Manages the schema for quizzes, possibly containing attributes like quiz title, category, and questions.
ResultModel.js	Stores user results, likely including fields for user ID, quiz ID, score, and date of completion.
ResumeModel.js	Might store resume-related data for users, with fields like name, experience, skills, and education.
UserModel.js	Manages user details, including attributes like username, password, email, and other profile information.

Chapter 4: Implementation

Implementation

As we get to the implementation phase of our job preparation platform, Prepminds: From Resume to Interview, we move from planning and design to the practical realization. This phase represents a critical milestone where we work on enhancing various operations through the deployment of a comprehensive digital solution with proper planning and careful consideration of project requirements that will enhance the efficiency, improve accuracy, and elevate the quality of service offered by pharmacies. Over the course of implementation.

4.1 Technique Used

4.1.1 Test-Driven Development

Test-driven development will be employed to ensure code quality and reliability throughout the development process.

Developers will write automated tests before implementing new features or making changes to existing code.

These tests will verify the expected behavior of the code and help prevent regressions.

4.1.2 Code Reviews

Code reviews will be conducted regularly to ensure code quality, maintainability, and adherence to coding standards.

Developers will review each other's code and provide constructive feedback, identify potential issues or improvements, and ensure consistency and

coherence across the codebase.

4.1.3 Modular Design

The project will be developed using a modular design approach, where functionality is divided into separate modules or components. Each module will be responsible for a specific set of features or tasks, allowing for better organization, reusability, andmaintainability of code.

4.1.4 Documentation

Comprehensive documentation will be created throughout the development process to capture project requirements, design decisions, architecture diagrams, and more that will help all users including job seekers as well as the admin have a clear view regarding the system's functionalities.

Documentation will facilitate knowledge sharing, onboarding of new team members, and troubleshooting of issues during development and maintenance phases of the job preparation platform.

4.2 Tools Used

Our implementation utilizes several key tools and libraries:

4.2.1 Visual Studio Code (VSC)

Visual Studio Code serves as a versatile and powerful development environment for the job preparation platform, offering a wide range of features and tools to support efficient and collaborative development workflows.

Prepminds: From Resume to Interview has utilized its intuitive interface, extensive customization options, and rich ecosystem of various extensions to ensure proper implementation of the platform.

Developers can write and edit code efficiently, reducing errors and improving code quality.

VS Code includes an integrated terminal, allowing developers to run commands

and execute scripts directly within the IDE. This eliminates the need to switch between different applications or command-line interfaces, streamlining development workflows.

It supports a wide range of extensions that enhance functionality and cater to specific development needs. Developers can install extensions for language support, debugging, code formatting, and more, customizing the IDE to suit their preferences and requirements.

It includes built-in debugging capabilities for various programming languages and frameworks through which the developers can set breakpoints, inspect variables, and step through code execution, making it easier to identify and fix bugs in the project.

Extensions like Live Server enable developers to preview and debug web applications in real-time directly within the IDE. This ensures that UI changes are reflected immediately.

4.2.2 Git

Git is a distributed version control system that has been utilized to track changes to the codebase over time. It has enabled team members to work collaboratively on the same project without interfering with each other's work.

Developers will use Git to track changes to the source code of the Prepminds: From Resume to Interview platform, including additions, modifications, and deletions of files.

It allowed us to create branches to work on new features or bug fixes independently of the main codebase.

Git provides features for resolving conflicts that may arise when multiple developers make changes to the same code simultaneously. It lets us to review changes, revert to previous versions, and maintain a clean and organized code history.

4.2.3 GitHub

Git is a distributed version control system that has been utilized to track changes to the codebase over time. It has enabled team members to work collaboratively on the same project without interfering with each other's work.

Developers will use Git to track changes to the source code of the Prepminds: From Resume to Interview platform, including additions, modifications, and deletions of files.

It allowed us to create branches to work on new features or bug fixes independently of the main codebase.

4.3 Language Used

4.3.1 Frontend Technologies:

- **HTML (Hypertext Markup Language):**

HTML will be used to structure the content of the web pages of Prepminds: From Resume to Interview. It provides the foundation for creating the layout and defining the elements that make up the user interface. HTML will be used to structure the content of the web pages in the project. It will define the layout and hierarchy of elements such as headings, paragraphs, lists, forms, tables,and links.

HTML features such as alt attributes for images, labels for form fields, and semantic markup will be implemented to enhance accessibility for users and improve search engine optimization (SEO).

- **CSS(Cascading Style Sheets):**

CSS will be used to style the HTML elements and define their appearance, layout, and design. It will control aspects such as colors, fonts, spacing, and alignment, ensuring a visually appealing and consistent user interface across the application. CSS media queries will be utilized to create

responsive layouts that adapt to different screen sizes and devices. This will ensure that the platform is accessible and usable across desktops, tablets, and mobile devices.

4. 3.1 Backend Technologies:

- **Javascript:**

It primarily serves as a frontend language for adding interactivity to web pages. JavaScript will be used to perform client-side form validation to provide immediate feedback to users on input errors and prevent unnecessary form submissions. This enhances user experience and reduces server load by validating input data before it is sent to the server.

4.3.2 Database:

- **MongoDB:**

MongoDB serves as the backbone of the PrepMinds platform, enabling efficient management of diverse data such as user profiles, resumes, quizzes, and results. Its flexibility as a NoSQL database allows dynamic updates to the data structure, supporting the platform's evolving features like adaptive quizzes and personalized resume templates. By integrating seamlessly with the backend technologies (Node.js and Express.js), MongoDB provides a robust environment for managing data interactions with the frontend built in React.js. Additionally, its scalability ensures that PrepMinds can handle increasing traffic and user demands without compromising performance.

To optimize performance, MongoDB leverages indexing to accelerate queries for frequently accessed fields such as user and quiz IDs. The aggregation framework is used to generate detailed analytics, such as tracking user progress and providing actionable insights. Deployed on cloud platforms like AWS or Google Cloud, MongoDB ensures high availability, fault tolerance, and scalability. This design enables PrepMinds to offer a seamless, user-friendly experience while maintaining a secure and responsive data environment.

4.4 Screenshots

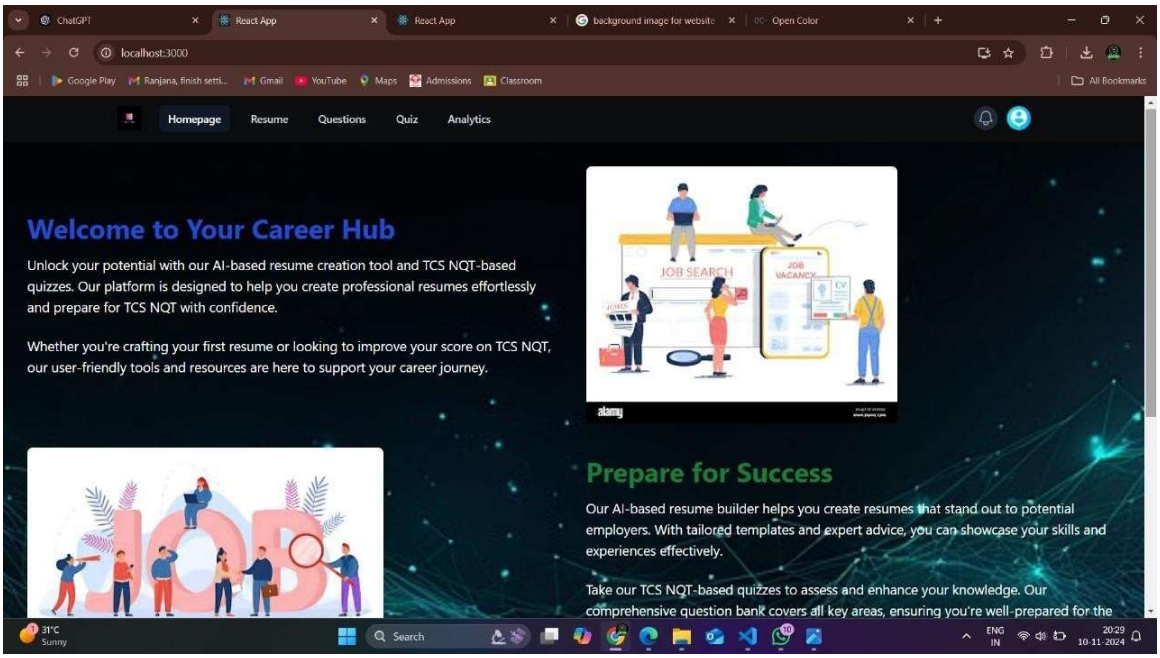


Figure 4-1: Homepage

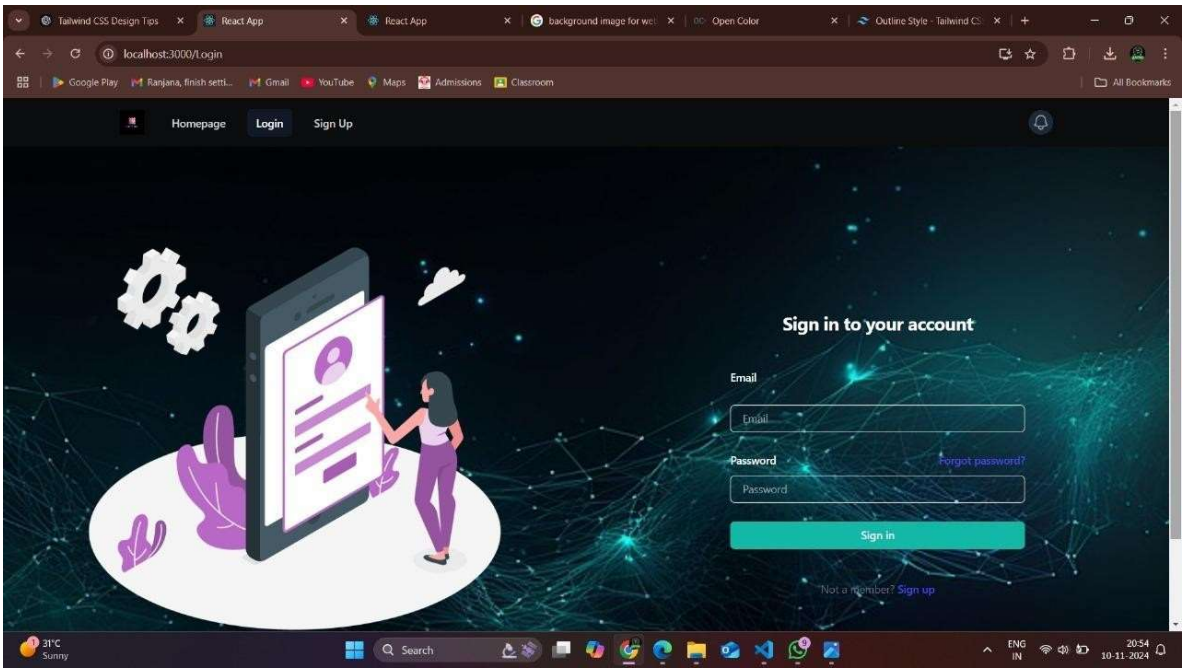


Figure 4-2: User Login

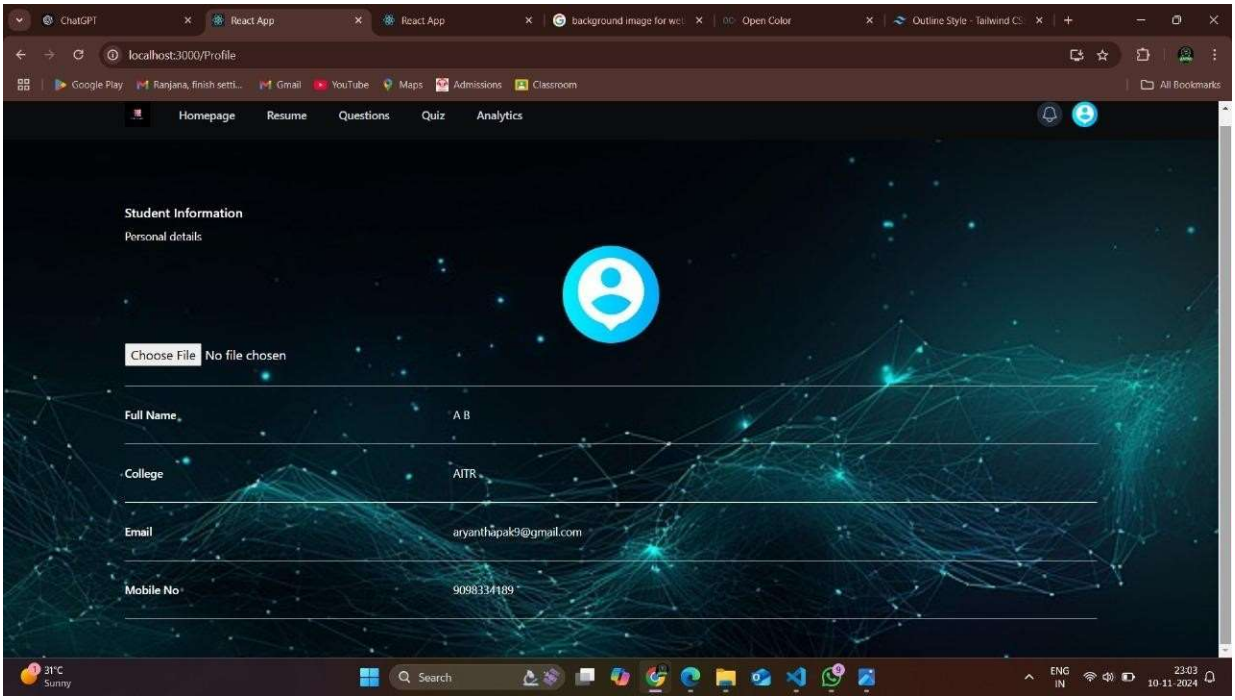


Figure 4-3: Resume Builder

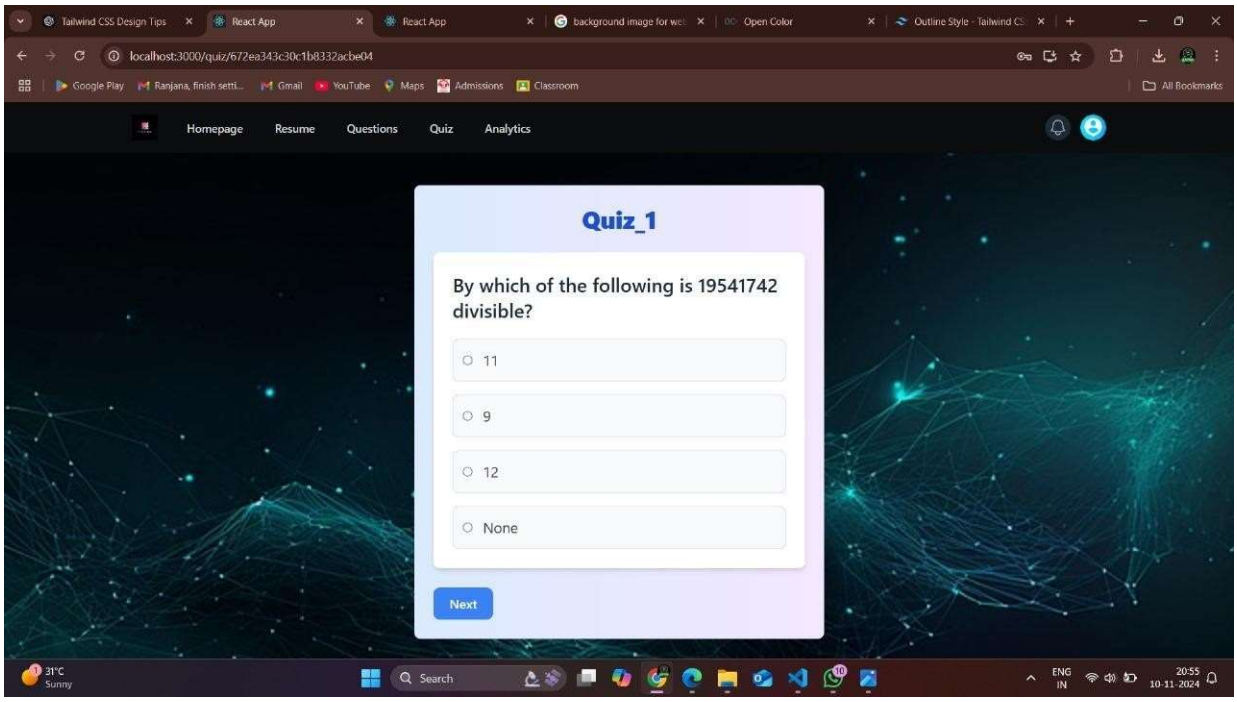


Figure 4-4: Attempt Quizzes

4.5 Testing

Testing is the process of evaluation of a system to detect differences between given input and expected output and also to assess the feature of the system. Testing assesses the quality of the product. It is a process that is done during the development process.

4.5.1 Strategy Used

- The testing strategy for the project includes a comprehensive approach to ensure the reliability, functionality, and security of the software. Beginning with unit testing to test individual components and functions to verify their correctness.
- Then we move to integration testing which validates the seamless interaction between different modules and components.
- User interface (UI) testing ensures the usability, responsiveness, and consistency across various devices and browsers.
- Functional testing covers all essential use cases and user information handling. Performance testing evaluates the system's performance under various load conditions.
- Compatibility testing ensures that the system functions correctly across different platforms and browsers.

4.5.2 Test Case and Analysis

Test Case ID	TC001
Test Case Summary	Try to login as User on PrepMinds
Test Procedure	Enter Detail to login for User type
Expected Result	The system should successfully login User on panel
Status	Pass

Table 4.1: Test Case 1

Test Case: 1 Input

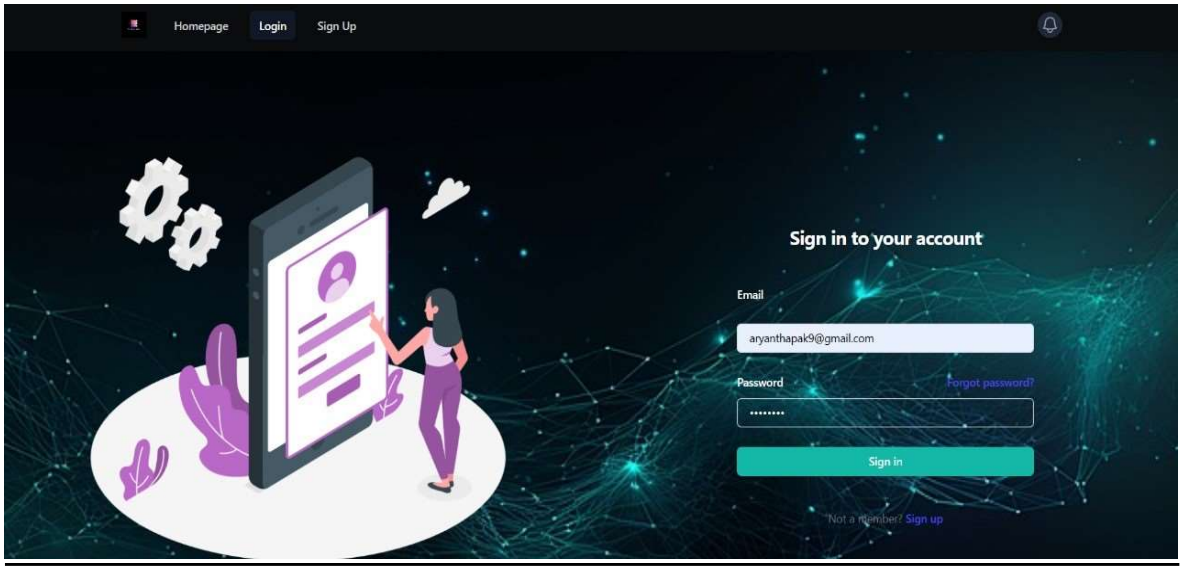


Fig 4.1 User Login Panel

Test Case: 1 Output

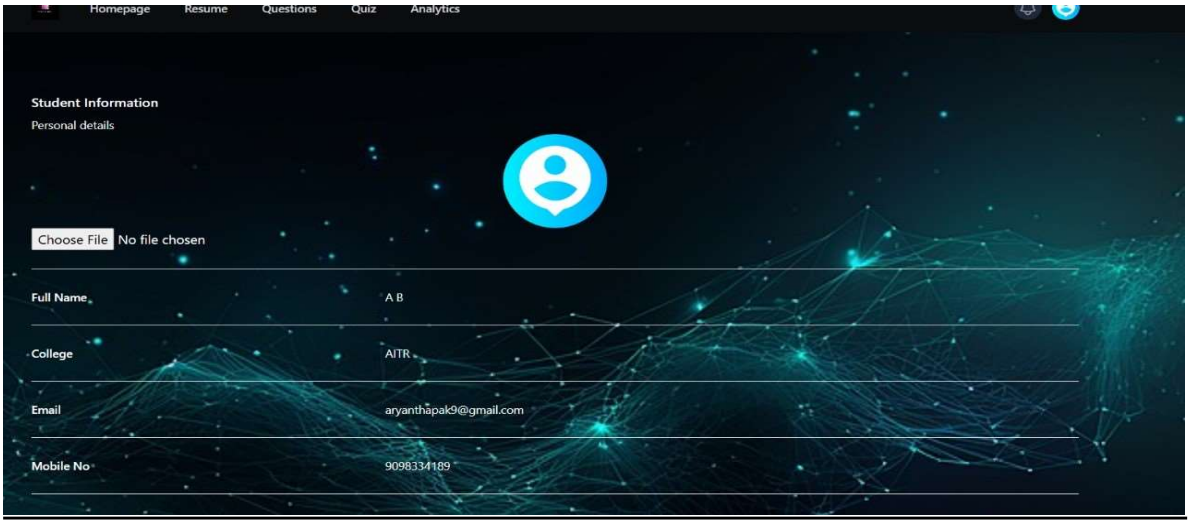


Fig 4.2 User Successfully login on Dashboard

Test Case: 2

Test Case ID	TC002
Test Case Summary	Fill Resume Form on PrepMinds
Test Procedure	Enter the detail in Resume Form
Expected Result	The Resume Generated Successfully
Status	Pass

Table 4.2: Test Case 2

Test Case: 2 Input

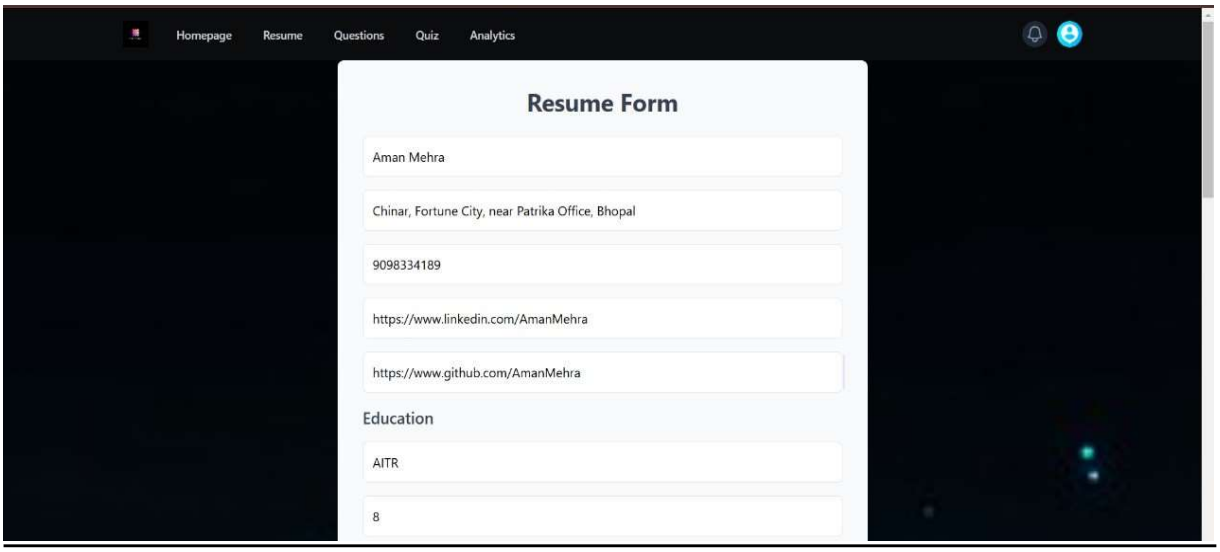


Fig 4.3 Resume Form

Test Case: 2 Output

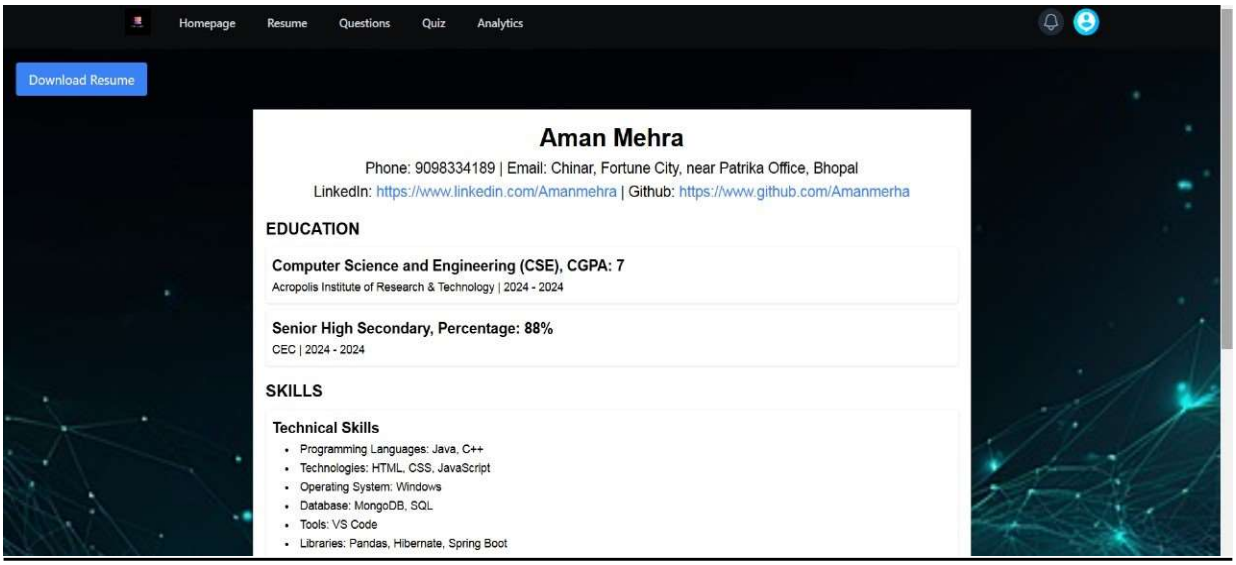


Fig 4.4 Resume Generated Successfully

Test Case: 3

Test Case ID	TC003
Test Case Summary	Attempt of Quiz using PrepMinds
Test Procedure	Attempt Quiz
Expected Result	Quiz Submitted Successfully and Result Shown
Status	Pass

Table 4.3: Test Case 3

Test Case: 3 Input

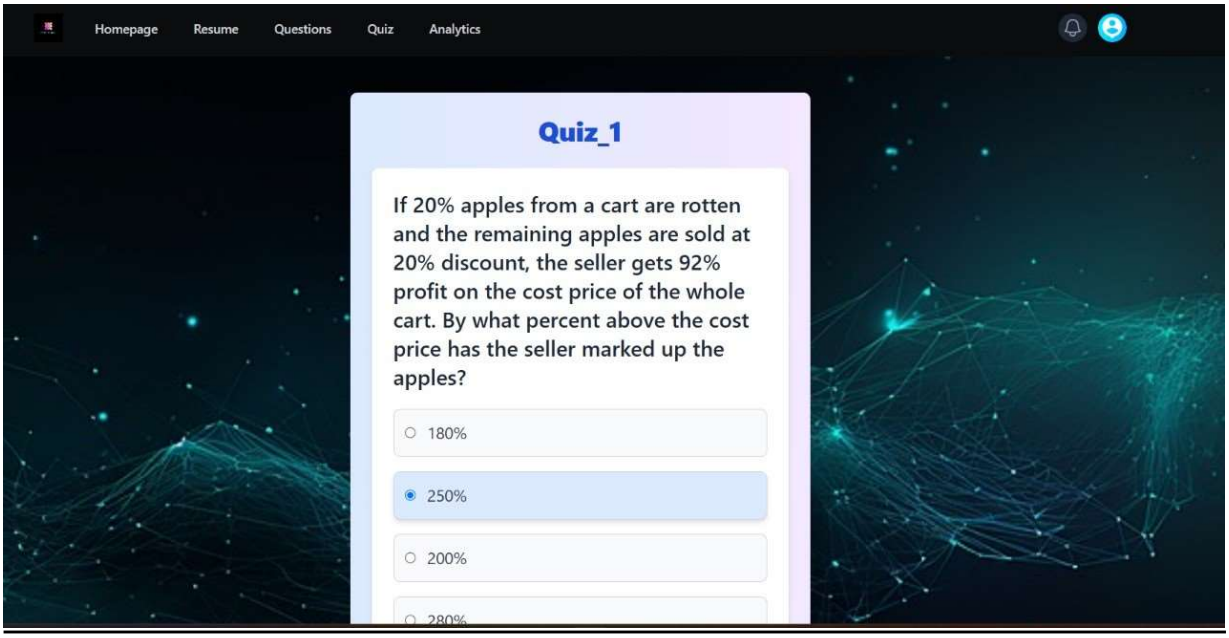


Fig 4.5 Quiz

Test Case: 3 Output

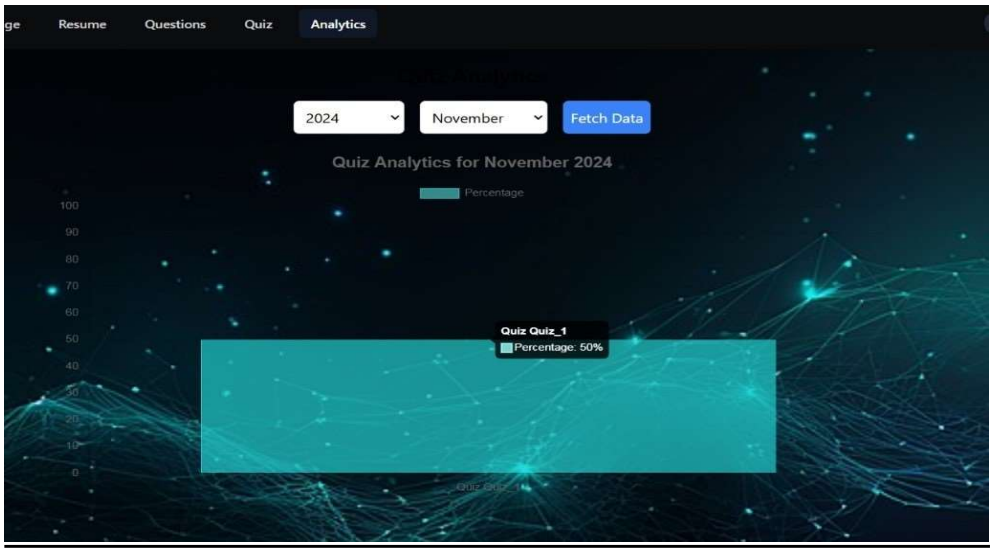


Fig 4.6 Quiz Average Result Show

Chapter 5: Conclusion

Conclusion

5.1 Conclusion

Ensuring the quality, security, and user experience of Prepminds: From Resume to Interview is essential for establishing its reliability as an all-in-one job preparation platform. The quality assurance process was designed to thoroughly assess the platform's functionality, performance, security, and overall usability to meet the highest standards. Multiple levels and types of testing were employed, from individual component verification to end-uservalidation, each playing a vital role in ensuring the platform's robustness and readiness for deployment. This structured testing approach allowed for early detection and resolution of any potential issues, optimizing the platform's performance and reliability for users.

A successful testing strategy for Prepminds: From Resume to Interview is not just about functionality; it is also about creating a seamless, responsive, and secure environment for job seekers who rely on the platform to achieve their career goals. Given the platform's comprehensive scope, involving resume building, skill assessments, and interview preparation tools, the testing process was meticulously designed to simulate real-world user interactions, maintain data integrity, and ensure that each feature performs as intended under various conditions.

5.2 Limitations of the Work

Despite the successful completion of core functionalities, Prepmind has a few limitations:

- The While Prepminds: From Resume to Interview provides extensive resources for technical roles, such as coding challenges and domain-specific quizzes, resources for non-technical job preparation, such as behavioral and situational interview practice, are comparatively limited. Expanding this

feature set will enhance Prepminds: From Resume to Interview' ability to serve users across a wider variety of career paths.

- As a cloud-based platform, Prepminds: From Resume to Interview requires an active internet connection to function fully. Users in areas with poor internet access may experience delays or disruptions, limiting their ability to use the platform effectively. Future plans could explore offline access for certain features, such as resume building or quiz downloads.
- Currently, Prepminds: From Resume to Interview provides general content recommendations based on user progress and performance. However, more sophisticated personalization, such as AI-driven insights based on individual learning styles or adaptive quizzes, is limited in this version. Enhanced personalization could improve user engagement and learning outcomes in future versions.
- Although Prepminds: From Resume to Interview includes a repository of company- specific interview questions, it does not currently integrate with major job boards for real-time job listings or application tracking. This limitation requires users to switch to other platforms for job applications, reducing Prepminds: From Resume to Interview' effectiveness as a complete job search solution. Future integrations with popular job boards could streamline the job search process.
- While the platform is designed to handle moderate user traffic, significant scaling may present challenges without further optimization. High traffic or rapid userbase growth could potentially lead to slower response times. To mitigate this, future updates will need to include performance optimizations and scalable infrastructure solutions.
- Accessibility testing identified areas where Prepminds: From Resume to Interview could be improved for users with disabilities, such as screen reader support and alternative navigation options for users with visual impairments. Adding accessibility enhancements would allow Prepminds: From Resume to Interview to serve a more diverse user base, promoting inclusivity and compliance with accessibility standard.

5.3 Suggestion and Recommendations for Future Work

- To serve users pursuing non-technical roles, additional resources should be developed, including behavioral, situational, and industry- specific interview practice questions. Interactive modules that simulate common non-technical interview scenarios could help Prepminds: From Resume to Interview support a wider range of job seekers, ensuring they are well-prepared for interviews in various fields.
- Providing offline access to select features, such as resume building and downloadable practice quizzes, would enable users to prepare without constant internet connectivity. Offline functionality would make Prepminds: From Resume to Interview more accessible to users in areas with limited connectivity, enhancing the platform's utility and flexibility.
- . Implementing AI-driven personalization can offer users customized guidance based on their preparation needs, learning styles, and progress over time. Adaptivequizzes and personalized interview recommendations would make Prepminds: From Resume to Interview a more intuitive and supportive tool, boosting engagement and improving user outcomes.
- Integrating Prepminds: From Resume to Interview with popular job boards (e.g., LinkedIn, Indeed) and professional networking platforms would allow users toapply directly for positions from within the platform. Real-time job listings and automated application tracking could provide a more seamless and efficient job search experience, enhancing Prepminds: From Resume to Interview as an all-in- one solution.
- To make Prepminds: From Resume to Interview more inclusive, future updates should include features to support users with disabilities, such as screen reader compatibility, keyboard navigation, and color contrast adjustments. Compliance with Web Content Accessibility Guidelines (WCAG) will ensure Prepminds: From Resume to Interview is accessible to a diverse audience, promoting inclusivity.
- As user numbers increase, scalability improvements will be essential to maintain optimal performance. Implementing load balancing, optimizing database queries, and using content delivery networks (CDNs) can improve response times and

ensure stability under high traffic conditions, allowing Prepminds: From Resume to Interview to scale effectively with demand.

- Adding more detailed analytics and progress reporting tools can help users track their strengths, improvement areas, and preparation milestones. Customized dashboards showing quiz scores, resume views, and interview readiness scores will empower users to make data-driven improvements in their job preparation journey.

Bibliography

[1] Katz, J., & Hicks, M. (2021). Web Development with Node and Express: Leveraging the JavaScript Stack. O'Reilly Media.

[2] Grinberg, M. (2018). Flask Web Development: Developing Web Applications with Python. O'Reilly Media.

[3] MongoDB, Inc. (2023). MongoDB Manual: The Comprehensive Guide to MongoDB. MongoDB Documentation. Retrieved from https://docs.mongodb.com/manual/

[4] McDowell, G. (2016). Cracking the Coding Interview: 189 Programming Questions and Solutions. CareerCup.

[5] Duckett, J. (2014). HTML and CSS: Design and Build Websites. Wiley.

[6] Rouse, M. (2019). REST API Design. TechTarget. Retrieved from https://www.techtarget.com/.

[7] WCAG 2.1 Compliance Guidelines. (2018). World Wide Web Consortium (W3C). Retrieved from https://www.w3.org/TR/WCAG21/.

[8] Amazon Web Services Documentation. (2023). AWS Cloud Infrastructure and Deployment Guide. Amazon. Retrieved from https://aws.amazon.com/documentation/.

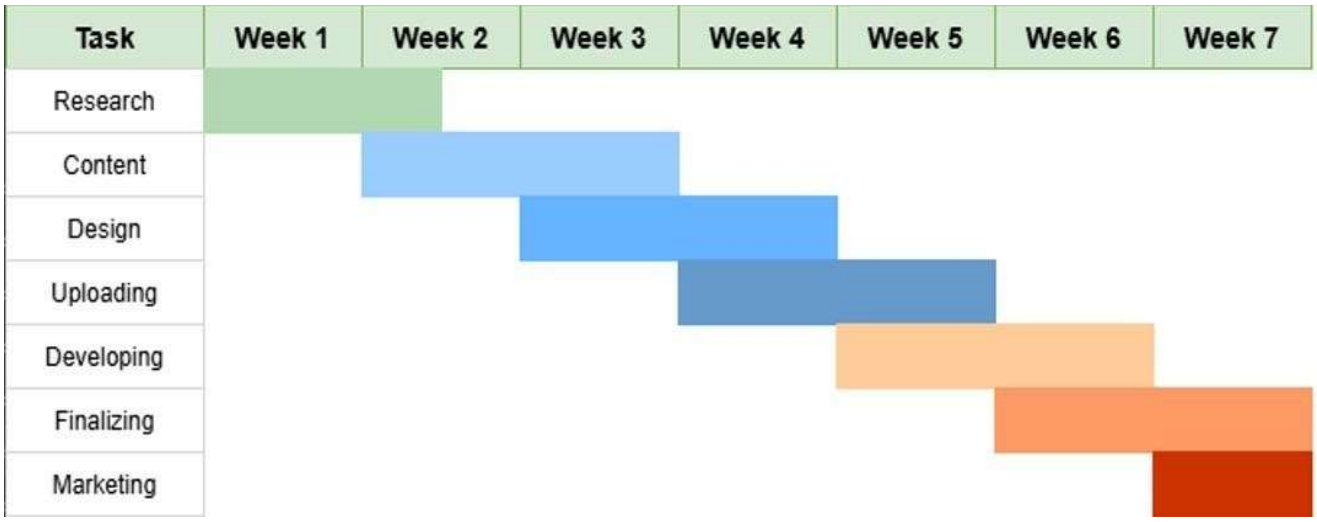
[9] Google Developers. (2023). Performance Optimization and Analytics with Google Analytics.

[10] Portny, S. E. (2021). Project Management for Dummies. Wiley.

[11] Nielsen, J., & Budiu, R. (2013). Mobile Usability. New Riders.

Project Plan

Gannt Chart



Guide Interaction Sheet

Date	Discussion	Action Plan
06/09/2024	Discussed project title selection.	Agreed upon "Prepminds: From Resume to Interviews" as the project title.
13/09/2024	Discussed the approach for project development.	Defined the project scope, objectives, and primary features, such as resume building and interview preparation
20/09/2024	Presented synopsis and diagrams for review.	Selected React.js for frontend, Node.js and Express.js for backend, and MongoDB for the database.
1/10/2024	Presented PowerPoint on the project.	Discussed progress on resume builder and interview preparation modules. Identified additional features to be added.
15/10/2024	Reviewed project implementation progress.	Reviewed quiz module and skill assessment. Identified areas for improvement in UI/UX.
25/10/2024	Presented and reviewed project report.	Verified the integration of company-specific question repository and performance tracking features.

09/11/2024	Submission of Project synopsis, research paper, and report for review.	Submitted the project report, user manual, and all documentation for final review and feedback.

Appendix A- Source Code

1. App.js

```
const mongoose = require("mongoose");
const express = require("express");
const dotenv = require("dotenv");
const cors = require("cors");
// Importing routers for different routes
const userRouter = require("./Routes/userRoutes");
const questionRouter = require("./Routes/questionRoutes");
const resumeRouter = require("./Routes/resumeRoutes");
const quizRouter = require("./Routes/quizRoutes");
// Configuring environment variables
dotenv.config();
// Initializing express app
const app = express();
// Middleware setup
app.use(express.json());
app.use(
  cors({
    credentials: true,
  })
);

// Connecting to MongoDB database
mongoose
  .connect(process.env.DATABASE.replace("<password>", process.env.DATABASE_PASSWORD))
  .then(() => {
    console.log("Data is connected");
  });
```

// Defining routes

```
app.use("/api/v1/user", userRouter);
app.use("/api/v1/questions", questionRouter);
app.use("/api/v1/resume", resumeRouter);
app.use("/api/v1/quiz", quizRouter);
// Starting the server
app.listen(process.env.PORT, () => {
  console.log(`Server is listening on ${process.env.PORT}`);
});
```

2. authController.js

```
const user = require("./../Model/UserModel");
const sendEmail = require("./../utils/nodeMailer");
const jwt = require("jsonwebtoken");
const catchAsync = require("../utils/catchAsync");
const url = require('./../utils/url');

// Signup function
exports.signup = catchAsync(async (req, res, next) => {
  const User = req.body;
  const newUser = await user.create(User);
  const token = jwt.sign(
    { email: req.body.email },
    process.env.JWT_Secret,
    { expiresIn: process.env.JWT_Expire }
  );

  newUser.Password = null;
  res.status(200).json({
    status: "Success",
    token,
    data: { user: newUser }
  });
```

```
  next();
});

// Login function
exports.login = catchAsync(async (req, res, next) => {
  const email = req.body.email;
  const User = await user.findOne({ email }).select("+Password");

  if (!User) {
    return res.status(404).json({ status: "Doesn't Exist" });
  }

  const password = req.body.password;
  const userPassword = User.Password;
  const isAuthenticated = await User.correctPassword(password, userPassword);

  if (!isAuthenticated) {
    return res.status(401).json({ status: "Unauthorized" });
  }

  const token = jwt.sign({ email }, process.env.JWT_Secret, {
    expiresIn: process.env.JWT_Expire,
  });

  User.Password = undefined;
  res.cookie("token", token, { httpOnly: true });

  res.status(200).json({
    status: "Success",
    token,
    data: { User },
  });
```

```
  next();
});
// Forgot Password function
exports.forgotPassword = catchAsync(async (req, res, next) => {
  const email = req.body.email;
  const exisitingUser = await user.findOne({ email });
  if (!exisitingUser) {
    return res.status(404).json({
      status: "Failed",
      message: "User Doesn't exists",
    });
  }
  const passwordResetToken = exisitingUser.createPasswordResetToken();
  exisitingUser.passwordResetToken = passwordResetToken;
  await exisitingUser.save({ validateBeforeSave: false });

  const resetURL = `${url}/${req.url}/${passwordResetToken}`;
  const text = `Forgot Your Password? Submit a PATCH request with your new password and
passwordConfirm to: ${resetURL}.\nIf you didn't forget your password, please ignore this email`;

  try {
    sendEmail({ email, text, subject: "Forgot your Password?" });
    res.status(200).json({ status: "Success" });
  } catch (err) {
    exisitingUser.passwordResetExpires = undefined;
    exisitingUser.passwordResetToken =  undefined;
    await exisitingUser.save({ validateBeforeSave: false });
    console.log(err);
    return res.status(500).json({
      status: "Failed",
      message: "There was an error while sending email, please try again later"
  });
```

```
    }
    next();
});

// Change Password function
exports.changePassword = catchAsync(async (req, res, next) => {
    const passwordResetToken = req.params.id;
    const User = await user.findOne({ passwordResetToken });

    if (!User) {
        return res.status(403).json({ status: "Failed", message: "Wrong reset token" });
    }
    if (Date.now() > User.passwordResetExpires) {
        return res.status(500).json({ status: "Failed", message: "Token is expired" });
    }
    const newPassword = req.body.password;
    User.Password = newPassword;
    User.passwordResetToken = undefined;
    User.passwordResetExpires = undefined;
    await User.save({ validateBeforeSave: false });
    res.status(200).json({ status: "Success", message: "Password changed successfully" });
    next();
});

// Check for Token function
exports.checkForToken = catchAsync(async (req, res, next) => {
    if (!req.headers.authorization || !req.headers.authorization.startsWith('bearer')) {
        return res.status(401).json({ status: "Failed", message: "Unauthorized" });
    }
    const token = req.headers.authorization.split(" ")[1];
    if (!token) {
        return res.status(404).json({ status: "Failed", message: "Token Not Found" });
    }
```

```
const isTokenVerified = await jwt.verify(token, process.env.JWT_Secret);
```

```
  if (!isTokenVerified) {
    return res.status(403).json({ status: "Failed", message: "Not a valid Token" });
  }
  res.status(200).json({ status: "Success" });
  next();
});
```

2. authController.js

```
const user = require("./../Model/UserModel");
const sendEmail = require("./../utils/nodeMailer");
const jwt = require("jsonwebtoken");
const catchAsync = require("../utils/catchAsync");
const url = require('./../utils/url');
// Signup function
exports.signup = catchAsync(async (req, res, next) => {
  const User = req.body;
  const newUser = await user.create(User);
  const token = jwt.sign(
    { email: req.body.email },
    process.env.JWT_Secret,
    { expiresIn: process.env.JWT_Expire }
  );

  newUser.Password = null;
  res.status(200).json({
    status: "Success",
    token,
    data: { user: newUser }
  });

  next();
});

// Login function
exports.login = catchAsync(async (req, res, next) => {
  const email = req.body.email;
  const User = await user.findOne({ email }).select("+Password");

  if (!User) {
    return res.status(404).json({ status: "Doesn't Exist" });
  }

  const password = req.body.password;
  const userPassword = User.Password;
  const isAuthenticated = await User.correctPassword(password, userPassword);

  if (!isAuthenticated) {
```

```
    return res.status(401).json({ status: "Unauthorized" });
  }

  const token = jwt.sign({ email }, process.env.JWT_Secret, {
    expiresIn: process.env.JWT_Expire,
  });

  User.Password = undefined;
  res.cookie("token", token, { httpOnly: true });

  res.status(200).json({
    status: "Success",
    token,
    data: { User },
  });

  next();
});

// Forgot Password function
exports.forgotPassword = catchAsync(async (req, res, next) => {
  const email = req.body.email;
  const exisitingUser = await user.findOne({ email });

  if (!exisitingUser) {
    return res.status(404).json({
      status: "Failed",
      message: "User Doesn't exists",
    });
  }

  const passwordResetToken = exisitingUser.createPasswordResetToken();
  exisitingUser.passwordResetToken = passwordResetToken;
  await exisitingUser.save({ validateBeforeSave: false });

  const resetURL = `${url}/${req.url}/${passwordResetToken}`;
  const text = `Forgot Your Password? Submit a PATCH request with your new password and
passwordConfirm to: ${resetURL}.\nIf you didn't forget your password, please ignore this email`;

  try {
    sendEmail({ email, text, subject: "Forgot your Password?" });
    res.status(200).json({ status: "Success" });
  } catch (err) {
    exisitingUser.passwordResetExpires = undefined;
    exisitingUser.passwordResetToken =  undefined;
    await exisitingUser.save({ validateBeforeSave: false });
    console.log(err);
    return res.status(500).json({
      status: "Failed",
      message: "There was an error while sending email, please try again later"
    });
  }
```

```
  next();
});

// Change Password function
exports.changePassword = catchAsync(async (req, res, next) => {
  const passwordResetToken = req.params.id;
  const User = await user.findOne({ passwordResetToken });

  if (!User) {
    return res.status(403).json({ status: "Failed", message: "Wrong reset token" });
  }

  if (Date.now() > User.passwordResetExpires) {
    return res.status(500).json({ status: "Failed", message: "Token is expired" });
  }

  const newPassword = req.body.password;
  User.Password = newPassword;
  User.passwordResetToken = undefined;
  User.passwordResetExpires = undefined;
  await User.save({ validateBeforeSave: false });

  res.status(200).json({ status: "Success", message: "Password changed successfully" });

  next();
});

// Check for Token function
exports.checkForToken = catchAsync(async (req, res, next) => {
  if (!req.headers.authorization || !req.headers.authorization.startsWith('bearer')) {
    return res.status(401).json({ status: "Failed", message: "Unauthorized" });
  }

  const token = req.headers.authorization.split(" ")[1];
  if (!token) {
    return res.status(404).json({ status: "Failed", message: "Token Not Found" });
  }

  const isTokenVerified = await jwt.verify(token, process.env.JWT_Secret);
  if (!isTokenVerified) {
    return res.status(403).json({ status: "Failed", message: "Not a valid Token" });
  }

  res.status(200).json({ status: "Success" });

  next();
});
```

3. questionController.js

```
const catchAsync = require("../utils/catchAsync");
const Question = require("./../Model/QuestionModel");
```

```
// Get Questions function
exports.getQuestions = catchAsync(async (req, res, next) => {
   const questions = await Question.find();
   res.status(200).json({
      status: "Successful",
      data: { questions }
   });

   next();
});

// Add Questions function
exports.addQuestions = catchAsync(async (req, res, next) => {
   const questions = req.body.questions;
   const Questions = await Question.create(questions);
   res.status(200).json({
      status: "Success",
      data: { Questions }
   });

   next();
});

// Delete Question function
exports.deleteQuestion = catchAsync(async (req, res, next) => {
   const questionId = req.body.id;
   await Question.findByIdAndDelete(questionId);
   res.status(200).json({ status: "Success" });

   next();
});

// Edit Question function
exports.editQuestion = catchAsync(async (req, res, next) => {
   const question = req.body.question;
   const updatedQuestion = await Question.findByIdAndUpdate(req.id, question, {
      new: true,
      runValidators: true
   });

   next();
});
```

Appendix B- StegaSafe User Manual

1. Introduction

Welcome to Prepminds: From Resume to Interviews, a comprehensive job preparation platform designed to simplify the job search process. Prepminds provides tools for resume building, interview preparation, skill assessments, and company-specific question banks. This manual will guide you through the features, installation process, and detailed usage instructions.

1.1 System Requirements

To use Prepminds effectively, ensure your system meets the following requirements:

- Operating System: Windows, macOS, or Linux
- Browser Compatibility: Latest versions of Chrome, Firefox, or Edge
- Internet Connection: Stable connection for optimal performance

2. Using of Prepminds

Prepminds is a cloud-based platform and requires no installation. Follow these steps to access the platform:

1. Visit the official Prepminds website.
2. Sign up using your email ID and create a secure password.:
3. Log in to access all features.

3. Features Overview

3.1 Resume Builder

Create professional resumes using customizable templates. Tailor your resumes with specific sections for experience, skills, and achievements.

3.2 Interview Preparation

Access a comprehensive repository of technical, behavioral, and company- specific interview questions. Utilize mock interview modules to simulate realscenarios.

3.3 Skill Assessments

Take interactive quizzes and track your progress across various domains.

Focus on areas requiring improvement based on detailed analytics.

3.4 SCompany-Specific Question Repository

Explore real interview questions from top companies to customize your preparation.

3.5 Progress Tracking and Insights

Monitor your readiness with personalized progress reports and actionable insights.

4. User Interface Walkthrough

4.1 Dashboard

The dashboard offers quick access to all modules, such as:

- **Resume Builder**: Start creating or editing your resumes.
- **Quizzes**: Launch assessments to test your skills.
- **Interviews**: Access preparation materials.

4.2 Resume Builder

Select a template, fill in your details, and export the resume in PDF format.

4.3 Interview Preparation

Browse question banks, practice coding exercises, or participate in mock interviews.

4.4 Skill Assessment

Choose from domain-specific quizzes to test and enhance your knowledge.

5. How to Build a Resume

1. From the dashboard, click on "Resume Builder"

2. Choose a template that matches your style and job requirements.

3. Fill in the required fields: Personal Information, Skills, Experience, etc.

4. Click "Save" to store or "Export" to download the resume as a PDF.

6. How to Prepare for an Interview

1. Navigate to the "Interviews" module on the dashboard.
2. Select a category (Technical, Behavioral, or Company-Specific).
3. Practice using mock interviews or review suggested questions.

7. Best Practices and Tips

- Customize Your Resume: Tailor it for each job application to stand out.
- Utilize Company-Specific Repositories: Focus your preparation on target employers.
- Practice Regularly: Use quizzes and mock interviews to build confidence.

8. Frequently Asked Questions (FAQ)

Q: Can I access Prepminds offline?

A: Prepminds currently requires an active internet connection. Future updates may include offline features for some modules.

Q: What file format are resumes exported in?

A: Resumes can be exported as PDFs for easy sharing

9. Troubleshooting

9.1 Login Issues

- Ensure your email and password are entered correctly.
- Use the "Forgot Password" option if needed.

9.2 Quiz or Module Access Issues

- Clear your browser cache and reload the page.
- Ensure your internet connection is stable.

10. Contact and Support

If you need additional support, please contact the Prepminds development team:

- **Aryan Thapak** – Team Lead
- **Anoushka Vyas** – Documentation Specialist
- **Anupam Kumar Raushan** – Developer
- **Adarsh Trivedi** – Developer
- **Aman Mehra** – Documentation Specialist

Appendices

Glossary

- **Resume Builder:** A tool to create professional resumes.
- **Mock Interviews:** Simulated interviews for practice.
- **Quizzes:** Assessments designed to test specific skills.
- **Company-Specific Repository:** A database of interview questions tailored for specific employers.