Major Project Report

On

Place-O-Track: “Smart Placement Mentor”

Submitted in the partial fulfilment of the requirement for the award of degree of

Bachelor of Technology

in

Computer Science and Engineering

Batch (2022-2026)

Submitted to:

Er. Bindu Bansal

Submitted by:

Priyanshu Garg

12200768

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DAV UNIVERSITYJALANDHAR-PATHANKOT NATIONAL
HIGHWAY, NH44, SARMASTPUR PUNJAB-14401

DECLARATION

I, PRIYANSHU GARG, hereby declare that the work which is being presented in this project/training titled “Place-O-Track: Smart Placement Mentor” by me, in partial fulfilment of the requirements for the award of Bachelor of Technology (B. Tech) Degree in “Computer Science and Engineering” is an authentic record of my own work carried out under the guidance of Er. Bindu Bansal (Project Mentor).

To the best of my knowledge, the matter embodied in this report has not been submitted to any other University/ Institute for the award of any degree or diploma.

Er. Bindu Bansal
Assistant Prof. CSE

Priyanshu Garg
12200768

ABSTRACT

Place-O-Track is an intelligent and efficient placement preparation and tracking system that simplifies learning, practice, and performance monitoring for students during campus placements. Inspired by the idea of “tracking your progress”, this platform connects every student’s effort with real-time outcomes such as quiz scores, coding performance, and progress analytics—creating a transparent and motivating preparation environment.

Built using modern technologies like React.js, Node.js, Express.js, and MongoDB, Place-O-Track offers a secure and scalable solution for managing aptitude practice, coding challenges, and past interview records. It provides dedicated panels for admins, students, and guests, supports real-time data updates, and promotes continuous improvement through performance-based insights. Place-O-Track encourages not just preparation, but ownership and consistency by ensuring that every attempt, question practiced, and skill improved is accurately tracked and highlighted.

This system has three panels.

- Admin
- Student
- Company

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my project guide, Er. Bindu Bansal, for their valuable guidance, constructive feedback, and continuous support throughout the duration of this project. I am also thankful to Dr. Rahul Hans and the Department of B.Tech CSE for providing the necessary resources and an encouraging academic environment.

My appreciation extends to all faculty members and classmates whose suggestions and cooperation helped strengthen the quality of this work. Lastly, I am deeply grateful to my family for their constant encouragement and support, which motivated me to complete this project successfully.

Table of Content

CHAPTER 1-INRODUCTION

1.1 Introduction of Project Report

The project report is a comprehensive document that provides detailed information about the proposed platform, **Place-O-Track**. It serves as a blueprint for all system operations, clearly defining the goals, objectives, and strategies required to transform this placement preparation concept into a fully functional and productive platform.

The report for Place-O-Track includes a structured breakdown of system modules, cost estimates, and expected outcomes, ensuring every stakeholder understands the scope, workflow, and long-term value of the platform. By offering these insights, the project report ensures that the design, development, and deployment of Place-O-Track remain efficient, well-planned, and goal-oriented.

Here are the points that validate the Importance of Project Report:

- A project report serves as a vital reference for developers and stakeholders, helping them monitor the ongoing progress of **Place-O-Track** and compare it with the initial development plan.
- It becomes a reliable source of information for evaluating overall performance, identifying growth, stagnation, team efficiency, and the quality of work.
- A well-prepared project report ensures completeness and accuracy, covering every important aspect of the project and making the collected data more dependable.
- It helps the project team anticipate and handle upcoming risks during various stages of development.
- It helps stakeholders and planners understand the exact initial requirements and resources needed to build and operate **Place-O-Track** efficiently.

1.2 Introduction to Project

In the rapidly changing environment of campus recruitment and skill development, students and institutions are increasingly searching for effective ways to practice aptitude, improve coding abilities, and prepare for interviews. Traditional preparation methods often lack structure, depth, and personalized progress tracking, making it difficult for learners to stay motivated and consistent. To overcome these challenges, **Place-O-Track** has been developed as a modern, real-time placement preparation and performance monitoring system.

CHAPTER 2- PROJECT TITLE AND OVERVIEW

2.1 Project Title – Place-O-Track: Smart Placement Mentor.

2.2 Project Overview –

Place-O-Track is a full-stack placement preparation and tracking system designed to streamline aptitude practice, coding exercises, interview preparation, and progress monitoring for students. The platform brings structure to learning workflows by offering role-based access for three key user types: **Admins, Students, and Guests**.

Developed using the **MERN stack (MongoDB, Express.js, React.js, and Node.js)**, Place-O-Track delivers a responsive, scalable, and interactive experience for efficient preparation. Admins can upload and manage aptitude questions, coding problems, and interview experiences, while also tracking student performance through analytics. Students, on the other hand, can practice quizzes, solve coding problems with test case validation, access past interview records, and monitor their improvement over time.

The system includes secure **JWT-based authentication**, company/year-wise filtering, bulk data upload, and real-time API communication for smooth operation.

The platform consists of two primary user roles: **Admin, Student** and **Company**.

1. Admin Panel:

The **Admin Panel** in **Place-O-Track** acts as the central control system for managing placement content, monitoring student performance, and maintaining the overall workflow of the platform. Admins have access to all essential functionalities, enabling them to:

- Add and manage aptitude questions, coding problems, and interview experiences with complete details such as category, difficulty level, company, and year.

- Monitor student activity and track their performance in quizzes and coding assessments in real time.

- Evaluate overall preparation trends based on quiz scores, coding attempts, and learning progress.

- Manage user records, including adding new students or modifying existing student profiles.

- Access analytics, question statistics, and system data securely through paginated APIs and a protected backend.

2. Student Panel:

The **Student Panel** in **Place-O-Track** is designed to offer learners a simple, user-friendly, and efficient workspace to manage their placement preparation activities. It enables students to stay updated, organized, and motivated through a clean interface, structured practice modules, and detailed performance-tracking features.

Key functionalities include:

- Secure login to access a personalized student dashboard using robust authentication mechanisms.

- Viewing available aptitude quizzes, coding problems, and interview questions with complete details such as difficulty, topic, and company.

- Receiving instant feedback on performance, including quiz scores, coding outputs, and improvement suggestions.

- Monitoring overall preparation progress through analytics, helping students stay consistent and enhance their aptitude and coding accuracy.

The panel promotes transparency and accountability by allowing employees to actively engage with their daily goals. It supports self-management and fosters a performance-driven culture where every action contributes to their professional growth.

2.3 Objectives

- To design and develop a complete placement preparation and tracking system with a responsive and user-friendly interface.

- To streamline the process of managing aptitude questions, coding problems, and interview experiences for both admins and students.

- To build a robust backend using Node.js, Express.js, and MongoDB for data storage, authentication, and API development.

- To develop an interactive frontend using React.js, enabling smooth navigation, conditional rendering, and real-time learning updates.

- To implement performance-based insights that help students evaluate their quiz results, coding accuracy, and preparation progress.

CHAPTER 3 – TECHNOLOGIES USED

3.1 Frontend:

- **HTML: Hyper Text Markup Language**

HTML (Hypertext Markup Language) is the standard language used to create and design webpages. It forms the backbone of all web content by structuring web pages with elements such as headings, paragraphs, images, links, tables, and lists. HTML is not a programming language in the traditional sense but a markup language used to define the structure of web content.

Key Points:

- **Elements and Tags**: HTML uses tags to define various parts of a webpage. Each element is enclosed in opening and closing tags (e.g., `<h1>` for headings, `<p>` for paragraphs).
- **Hyperlinks**: One of the main features of HTML is its ability to link web pages through hyperlinks using the `<a>` tag. This allows navigation between pages and websites, forming the web's "hypertext" structure.
- **Forms and User Input**: HTML forms are used to collect user input such as text, passwords, checkboxes, and radio buttons. These forms can be submitted to the server for processing and storage.
- **Document Structure**: HTML defines the overall structure of a webpage with essential components like the `<head>` and `<body>`.

Importance of HTML in Project:

In **Place-O-Track**, HTML is used to build the foundational structure of both the Admin, Company and Student panels. It organizes all interface elements clearly, allowing users to navigate quizzes, view coding problems, and access interview sections with ease. When combined with CSS and React, HTML enables the platform to deliver a responsive, clean, and accessible user experience across all types of devices.

• CSS: Cascading Style Sheets

CSS (Cascading Style Sheets) is a style sheet language used to control the visual presentation of a web page written in HTML or XML. While HTML provides the structure of a webpage, CSS is responsible for styling it, including layouts, colors, fonts, and overall design.

Key Points:

- **Styling Elements**: CSS controls the appearance of HTML elements, such as changing font styles, sizes, colors, and spacing to improve readability and design.
- **Layouts and Positioning**: CSS enables precise control over the positioning and layout of elements on a page using properties like `margin`, `padding`, `flexbox`, and `grid`.
- **Responsive Design**: With features like media queries, CSS ensures that websites look good on devices of all sizes, from desktops to smartphones.
- **Animation and Interactivity**: CSS supports animations and transitions, enabling developers to create interactive visual effects without relying on JavaScript.

Types of CSS:

1. **Inline CSS**: Styles applied directly to an HTML element using the `style` attribute. Example: `<p style="color: red;">This is red text.</p>`
2. **Internal CSS**: Styles written inside a `<style>` tag within the HTML `<head>`. Example: <style> p { color: blue; } </style>
3. **External CSS**: Styles written in a separate `.css` file and linked to the HTML document. Example: <link rel="stylesheet" href="styles.css">

Importance of CSS in Project:

In **Place-O-Track**, CSS is used to style various interface components such as buttons, forms, dashboards, and question cards across both the Admin and Student panels. It ensures consistent use of colors, typography, spacing, and layout, giving the platform a clean and polished look.

- ## Bootstrap:

Bootstrap is a popular open-source front-end framework used for developing responsive and mobile-first websites. It was initially developed by Twitter and is now maintained by the open-source community. Bootstrap combines **HTML**, **CSS**, and **JavaScript** components to create visually appealing and functional web pages with minimal effort.

Key Points:

- **Responsive Design**: Bootstrap is built on a mobile-first approach, ensuring that web pages are responsive and adapt seamlessly to various screen sizes, from desktops to smartphones.
- **Pre-designed Components**: It offers a wide range of reusable UI components like navigation bars, buttons, forms, modals, and carousels, which can be easily customized and integrated into projects.
- **Grid System**: The framework includes a powerful 12-column grid layout system for creating flexible and responsive layouts.
- **Cross-Browser Compatibility**: Bootstrap ensures that web pages look and function consistently across different browsers.

Importance of JavaScript in Project:

In the **Place-O-Track** project, Bootstrap plays an important role in creating a clean, structured, and responsive user interface for both the Admin and Student panels. Its powerful grid system allows quick layout adjustments and ensures that every page looks professional across different devices and screen sizes.

- ## JavaScript:

JavaScript (JS) is a high-level, versatile, and lightweight programming language primarily used to create dynamic and interactive web applications. It is one of the core technologies of the web, alongside **HTML** and **CSS**, enabling developers to enhance user experiences by adding interactivity to websites.

Key Points:

- **Interactivity**: JavaScript allows developers to add interactivity to web pages, such as dropdown menus, sliders, pop-ups, and form validation.
- **DOM Manipulation**: JS can dynamically update and manipulate the HTML and CSS of a webpage, enabling features like live content updates without refreshing the page.
- **Event Handling**: JavaScript allows developers to respond to user actions like clicks, hover, and input using event listeners.
- **Server-Side Programming**: With platforms like Node.js, JavaScript can be used to build scalable server-side applications.

Importance of JavaScript in Project:

In **Place-O-Track**, JavaScript is used to manage events such as quiz submissions, coding problem attempts, form handling, and interactive button actions. It also enables smooth communication with backend APIs, ensuring that aptitude questions, coding data, and interview records are fetched, displayed, and updated without requiring a full page reload. JavaScript adds dynamism to the interface and significantly enhances the overall user experience.

- ## React.js + Vite:

React.js is a popular open-source JavaScript library used for building user interfaces, especially single-page applications (SPAs). Developed and maintained by **Meta (formerly Facebook)**, React focuses on creating dynamic and responsive web applications with high performance and a seamless user experience.

Vite is a modern build tool and development server that significantly improves the React development experience.

Key Features of React.js + Vite:

- **Fast Development Start-up:** Vite offers significantly faster server initialization compared to traditional tools such as Create React App. This improves the development workflow and reduces waiting time during project setup.

- **Optimized Production Builds:** Vite uses Rollup as its build tool, producing compact, tree-shaken, and highly optimized bundles for deployment. This ensures that the application runs efficiently in production environments.
- **Lightweight and High Performance:** The overall performance of applications built with Vite is enhanced due to smaller bundle sizes and faster reloading during both development and production stages.

Importance of React.js + Vite in Project:

In **Place-O-Track**, React.js is used to develop both the Admin Panel and Student Panel with a clean separation of components such as quiz modules, coding problem cards, dashboards, and performance analytics. React's virtual DOM ensures fast and efficient rendering, while hooks and props simplify state management and real-time UI updates.

3.2 Backend:

- ### Node.js:

Node.js is an open-source, cross-platform runtime environment that allows developers to run JavaScript code on the server side. Built on **Google Chrome's V8 JavaScript engine**, Node.js is lightweight, efficient, and designed for building scalable and high-performance network applications.

Key Points:

- **Cross-Platform**: Node.js runs on Windows, Linux, macOS, and other platforms, making it versatile for various development environments.
- **Built-In Modules**: Node.js comes with a set of core modules, such as `http`, `fs` (file system), and `path`, that simplify backend development tasks.

- **Package Ecosystem (npm)**: Node.js has the largest package ecosystem through npm (Node Package Manager), offering thousands of reusable libraries and tools.
- **JSON Support**: Node.js works seamlessly with JSON, making it an excellent choice for building RESTful APIs.

Importance of Node.js in Project:

In **Place-O-Track**, Node.js is used to develop RESTful APIs that handle critical operations such as user authentication, managing aptitude and coding questions, processing quiz submissions, and storing interview data.

- ## Express.js:

Express.js is a fast, unopinionated, and minimalist web application framework for **Node.js**, widely used for building server-side applications and RESTful APIs.

Key Points:

- **Easy Routing**: Provides a simple way to define routes for different HTTP methods (GET, POST, PUT, DELETE).
- **Middleware Support:** Easily integrates middleware functions for tasks like validation, authentication, error handling, and more.
- **RESTful API Creation:** Ideal for building RESTful services that interact with databases and frontend apps.
- **Integration Friendly:** Works seamlessly with MongoDB (via Mongoose), React (frontend), and other Node packages.
- **Efficient Error Handling:** Offers built-in mechanisms and custom middleware for managing errors gracefully.

Importance of Express.js in Project:

In the **Place-O-Track** project, Express.js is used to build the backend server, commonly known as the Express server. This server manages all essential backend operations, including question management, user authentication, data validation, and secure API communication.

- ## MongoDB:

MongoDB is a popular open-source, NoSQL database designed for high performance, scalability, and flexibility. Unlike traditional relational databases, MongoDB stores data in JSON-like documents using a flexible, schema-less model.

Key Points:

- **NoSQL Database** – MongoDB is a non-relational database that stores data in flexible, JSON-like documents instead of traditional tables.
- **Document-Oriented** – Each record is stored as a **document** (similar to a JavaScript object), making it intuitive for developers working with JavaScript/Node.js.
- **Integration with Mongoose** – Easily integrates with Mongoose for schema validation, relationships, and advanced querying.
- **Cloud Support (MongoDB Atlas)** – Offers a fully managed cloud solution for hosting MongoDB databases with automatic backups and scaling.
- **Cross-Platform** – Works seamlessly on Windows, macOS, Linux, and cloud platforms like AWS, Azure, and Google Cloud.

Importance of MongoDB in Project:

In the **Place-O-Track** project, MongoDB is used to store and manage all essential data, including user information, aptitude questions, coding problems, interview records, quiz results, and login credentials.

3.3 Tools Used:

- ## MongoDB Compass:

MongoDB Compass is the official graphical user interface (GUI) provided by MongoDB for visually exploring, managing, and analyzing the contents of a MongoDB database.

Key Points:

- **Collection Management** – Easily create, update, delete, and filter collections and documents.
- **Real-Time Monitoring** – Displays server performance statistics such as index usage, query performance, and database size.

- **Cross-Platform Tool** – Available for Windows, macOS, and Linux.
- **Ideal for Development** – Makes testing, debugging, and inspecting data easier during the backend development phase.

Importance of MongoDB in Project:

In the **Place-O-Track** project, MongoDB Compass is used to view collections, inspect documents, monitor real-time stored data, and perform CRUD operations directly on the database.

- ## Postman:

Postman is a widely used API development and testing tool that provides a powerful and user-friendly interface to interact with backend services.

Key Points:

- **API Testing Tool** – Allows developers to test, debug, and document RESTful APIs without needing a frontend.
- **User-Friendly Interface** – Provides an intuitive GUI to send HTTP requests (GET, POST, PUT, DELETE, etc.) and view responses.
- **JWT & Auth Integration** – Supports various authentication methods, including JWT, OAuth, and API keys.

Importance of Postman in Project:

In the **Place-O-Track** project, Postman is used extensively to test the RESTful APIs developed with Node.js and Express.js. It helps verify that routes for user login, question management, quiz submission, coding evaluation, and interview data handling are working correctly.

- ## Visual Studio Code:

Visual Studio Code (VS Code) is a free, open-source code editor developed by Microsoft. It is widely used by developers for writing, debugging, and managing code across a wide range of programming languages and frameworks.

Key Points:

- **Lightweight and Fast** – A powerful yet lightweight source-code editor that works smoothly across platforms.
- **Multi-Language Support** – Supports JavaScript, HTML, CSS, Node.js, React.js, and many other programming languages.
- **Built-in Git Integration** – Enables version control with Git and GitHub directly within the editor.

Importance of Visual Studio Code in Project:

In the Place-O-Track project, VS Code is used to develop both the frontend (React.js) and backend (Node.js & Express.js) code.

- ## GitHub:

GitHub is a web-based platform that provides **version control and collaborative development** using **Git**, a distributed version control system.

Key Points:

- **Code Backup & Recovery** – Acts as a secure backup for your entire project, preventing data loss.
- **Supports Open Source & Private Projects** – Projects can be public for open-source contribution or private for internal use.
- **Community and Documentation** – Offers a strong developer community and built-in tools like README files and Wikis for project documentation.

Importance of Visual Studio Code in Project:

In the Place-O-Track project, GitHub is used to **host the source code** for both frontend and backend components. It allows seamless version tracking, making it easy to review changes, revert errors, and maintain a history of development progress.

CHAPTER 4 – Software Development Life Cycle (SDLC) Model

4.1 Introduction to SDLC:

The **Software Development Life Cycle (SDLC)** is a systematic process used for developing software that ensures quality, efficiency, and structure in the development workflow.

In the context of the In the Place-O-Track project, the SDLC model served as a guideline to execute the project step by step, ensuring proper planning, structured coding, and successful deployment of the full-stack web application.

4.2 SDLC Model Used:

For the development of the In the Place-O-Track project, the **Waterfall Model** of SDLC was followed. The Waterfall Model is a **linear and sequential** approach where each phase of the software development process flows logically into the next.

This model was chosen for Place-O-Track to ensure a **structured and organized development process**, with well-defined goals at every stage—from requirement gathering to deployment.

4.3 Phases of SDLC:

The development of Place-O-Track was carried out in a structured sequence, following the key phases of the Waterfall SDLC model.

4.3.1 Requirement Analysis

In this phase, detailed discussions were held to understand the project's functional and non-functional requirements. The system needed to support role-based access), placement task and tracking, performance monitoring, leave handling, and secure login.

4.3.2 System Design

Based on the gathered requirements, the overall architecture of the system was designed. This included:

Place-O-Track

- Designing the database schema using **MongoDB** and **Mongoose**
- Planning RESTful API routes in **Express.js**
- Structuring frontend components using **React.js**
- Designing the UI with responsiveness in mind using **Bootstrap**

4.3.3 Implementation

This phase involved the actual coding and development:

- Backend was developed using **Node.js** and **Express.js**
- Frontend was developed using **React.js**
- APIs were created for task management, user authentication, leave processing, and performance tracking
- MongoDB was used to store data, such as users, tasks, and rewards

The code was written in **VS Code** and version-controlled using **GitHub**.

4.3.4 Testing

After implementation, thorough testing was done using tools like:

- **Postman** for API testing
- **Console debugging** for frontend validation
- **MongoDB Compass** for checking data integrity

Manual and functional testing ensured that each module performed as expected and edge cases were handled.

4.3.5 Deployment

The system was prepared for deployment. Backend services and the frontend were tested on different machines to ensure cross-device functionality.

4.3.6 Maintenance

As part of the maintenance phase, minor changes and improvements were made based on testing feedback.

Chapter 5: System Design and Requirements

5.1 System Architecture

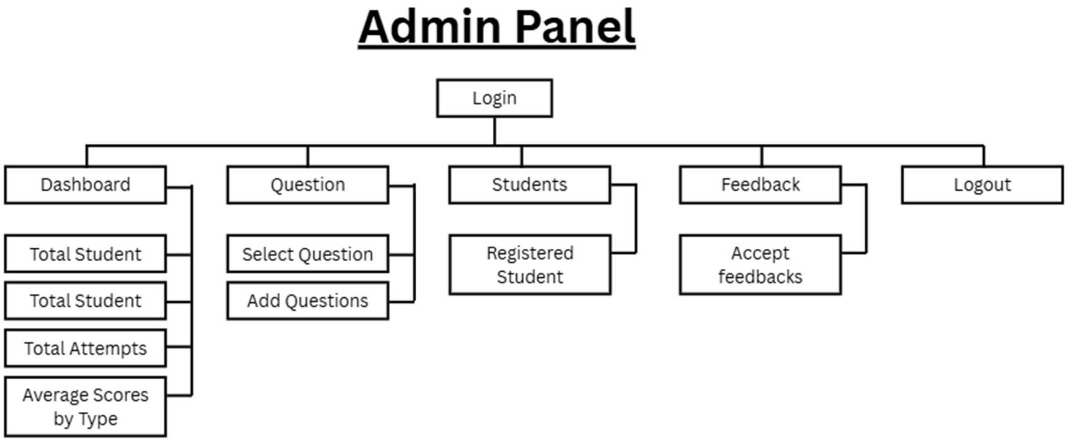
- **Frontend (React.js):** Handles the user interface for Admin and Employee panels using components and Bootstrap for responsiveness.
- **Backend (Node.js + Express.js):** Manages routing, business logic, API creation, and validation. It processes requests from the frontend and interacts with the database.
- **Database (MongoDB):** Stores all project data including users, tasks, performance records, and leave applications. Connected via Mongoose for schema management.
- **API Communication:** Frontend communicates with the backend using RESTful APIs.
- **Development Tools:** VS Code for coding, GitHub for version control, and MongoDB Compass for viewing database records.

5.2 Data Flow Diagrams (DFDs):

Data Flow Diagrams (DFDs) visually represent the flow of data through the system. For the Place-O-Track project, Level 2 DFDs are included to illustrate detailed processes within the Admin, Company and Student panels.

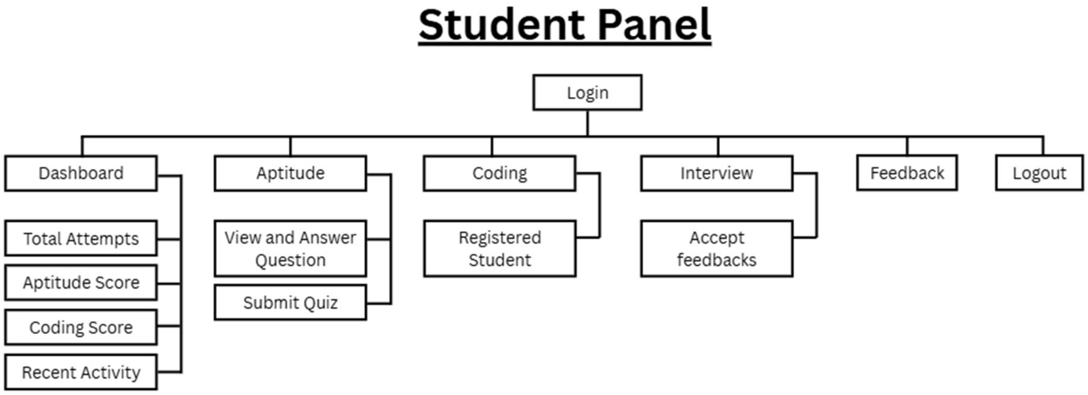
5.2.1 Admin Panel

Admin Panel illustrating data flow for category, subcategory, project, team, task, and performance (reward/warning) management.



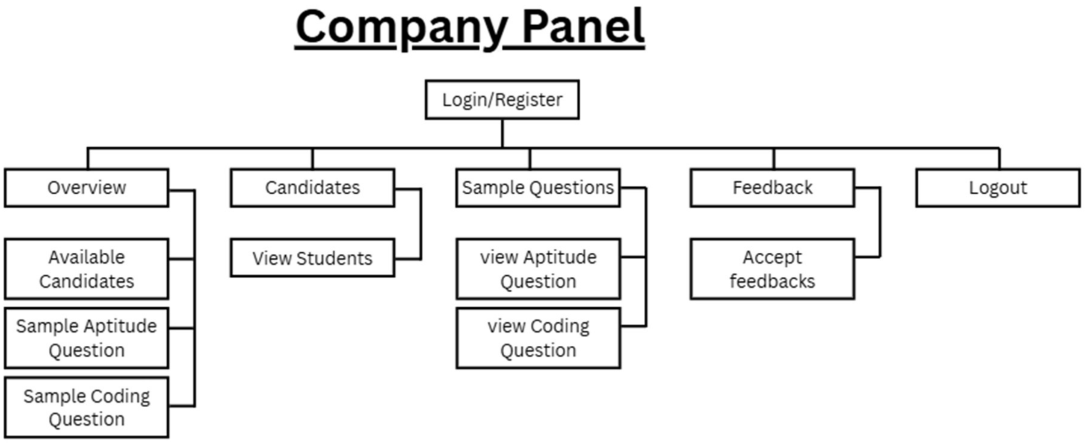
5.2.1 Student Panel

DFD of Student Panel showing the flow for task viewing, submission, profile management, and performance tracking.



5.2.1 Company Panel

DFD of Company Panel showing the flow for task viewing, submission, profile management, and performance tracking.



5.3 System Requirements:

The development and execution of the Place-O-Track project require specific hardware and software components to ensure smooth performance and compatibility. Below are the minimum and recommended requirements:

5.3.1 Hardware Requirements:

Component	**Minimum Requirement**	**Recommended Requirement**
Processor	Intel Core i3	Intel Core i5 or above
RAM	4 GB	8 GB or more
Hard Disk	100 GB (free space)	256 GB SSD
Monitor	1024 × 768 resolution	Full HD (1920 × 1080)
Input Devices	Keyboard and Mouse	Standard Input Devices
Internet Connection	Required for testing & deployment	Stable High-Speed Connection

5.3.2 Software Requirements:

Software	**Details / Version**
Operating System	Windows 10/11, Ubuntu 20+, or macOS
Code Editor	Visual Studio Code
Backend	Node.js (v16+) with Express.js
Frontend	React.js (with Vite or Create React App)
Database	MongoDB (Local or MongoDB Atlas)
Database GUI	MongoDB Compass
API Testing Tool	Postman
Version Control System	Git and GitHub
Browser	Google Chrome / Mozilla Firefox
Terminal/Command Line	Windows CMD / Bash / PowerShell

Chapter 6: Testing, Maintenance, and Results

6.1 Testing:

Testing ensures that all modules of the platform function correctly and meet the defined requirements. In the **Place-O-Track** project, multiple levels of testing were performed to validate system functionality, performance, and overall reliability.

6.1.1 API Testing with Postman

All backend APIs developed in Node.js and Express.js were tested using **Postman**. Each route (GET, POST, PUT, DELETE) was tested with valid and invalid inputs to verify:

- Data validation
- Status codes (200, 201, 400, 422, 500)
- Authentication using JWT tokens

6.1.2 Module Testing

Each panel (Admin and Employee) was tested separately to ensure the UI components and logic were working properly:

- Admin side: Question creation, student handling.
- Student side: viewing Question, submitting tasks, feedback

6.1.3 Validation Testing

Field validations were tested on both frontend and backend:

- Required fields (e.g., email, password, task title)
- Unique email/user check

6.1.4 Error Handling Tests

Test cases were created to simulate invalid requests and server errors. Custom error messages and proper HTTP status codes were returned for:

- Missing fields

Place-O-Track

- Unauthorized access
- Invalid data formats

6.2 Maintenance

It ensures that the system continues to operate smoothly, securely, and efficiently after deployment. The **Place-O-Track** project involves both basic and ongoing maintenance strategies to fix bugs, enhance features, and prepare for future upgrades and improvements.

6.2.1 Bug Fixes and Enhancements

During testing, several bugs and functional issues were identified and resolved:

- Fixed API validation errors for login, task creation, and leave requests.
- Corrected UI alignment and responsiveness using Bootstrap.

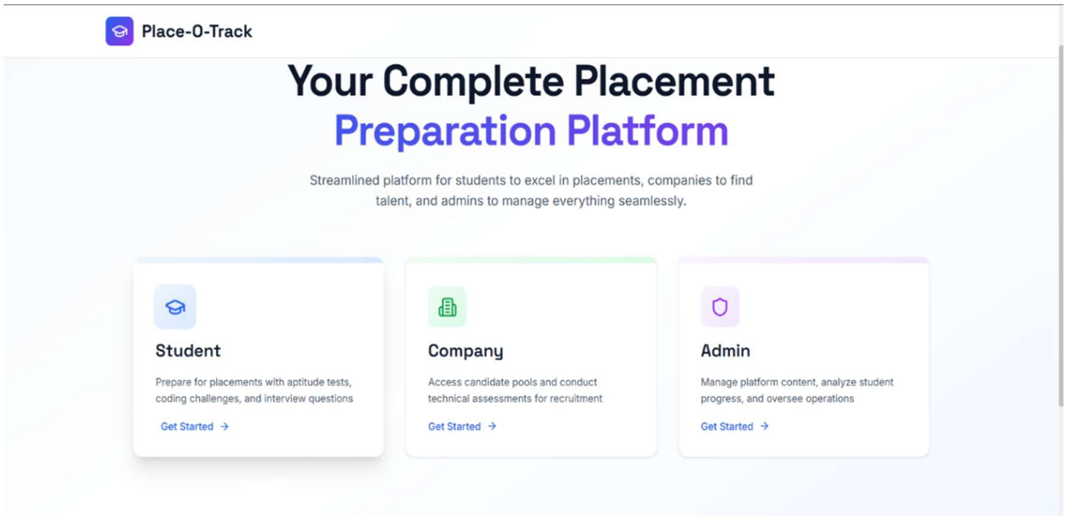
6.2.2 Future Scope and Scalability

To ensure the project can grow and adapt to future needs, the following enhancements are planned:

- Integration of **Email or SMS notifications** for task updates and approvals.
- Implementing **real-time updates** using technologies like Socket.IO.

6.3 Result

1. Landing Page:



Place-O-Track

Your Complete Placement Preparation Platform

Streamlined platform for students to excel in placements, companies to find talent, and admins to manage everything seamlessly.

Student

Prepare for placements with aptitude tests, coding challenges, and interview questions

Get Started →

Company

Access candidate pools and conduct technical assessments for recruitment

Get Started →

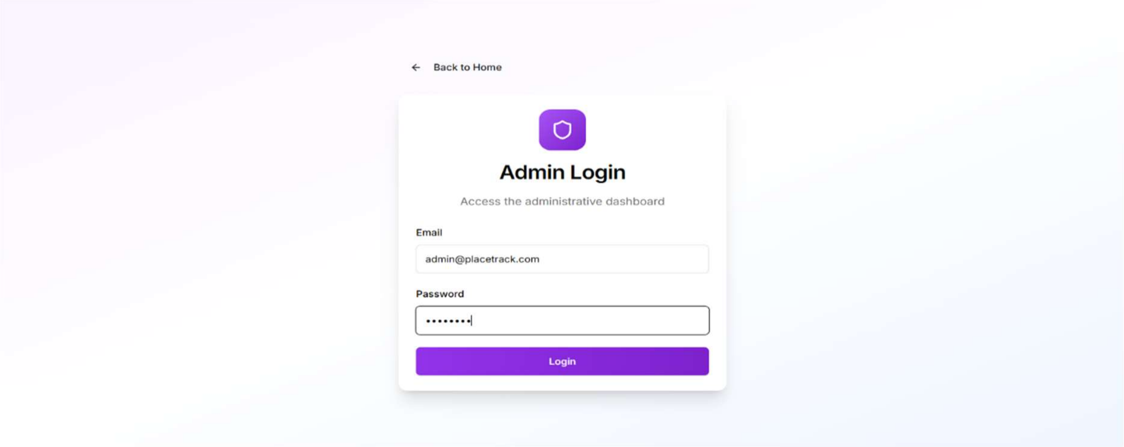
Admin

Manage platform content, analyze student progress, and oversee operations

Get Started →

2. Admin Login:

- User Id: admin@placetrack.com
- Password: admin123



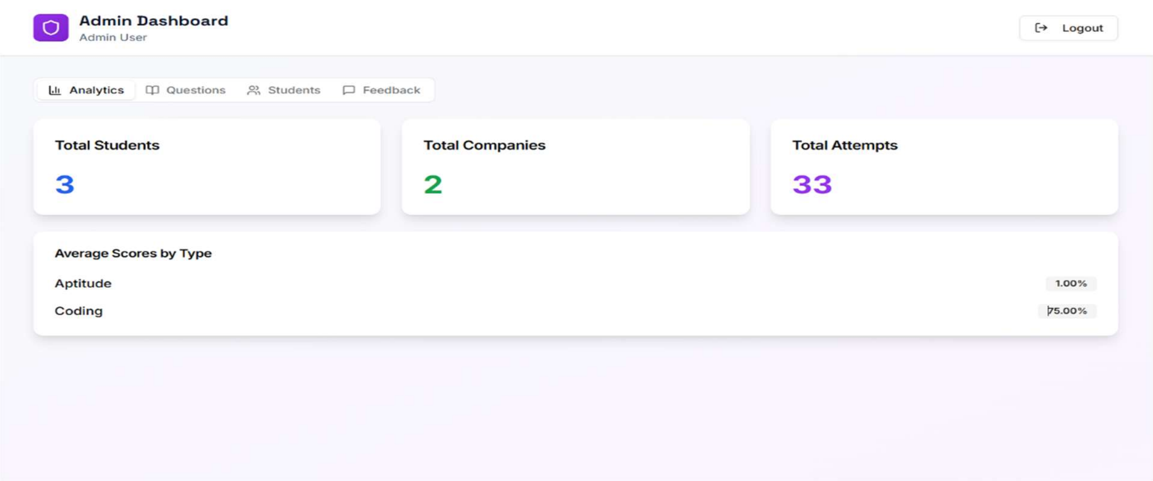
Admin Login

Access the administrative dashboard

Email: admin@placetrack.com

Password: ••••••••

2.1. Admin Dashboard:



Admin Dashboard

Admin User

Total Students: 3

Total Companies: 2

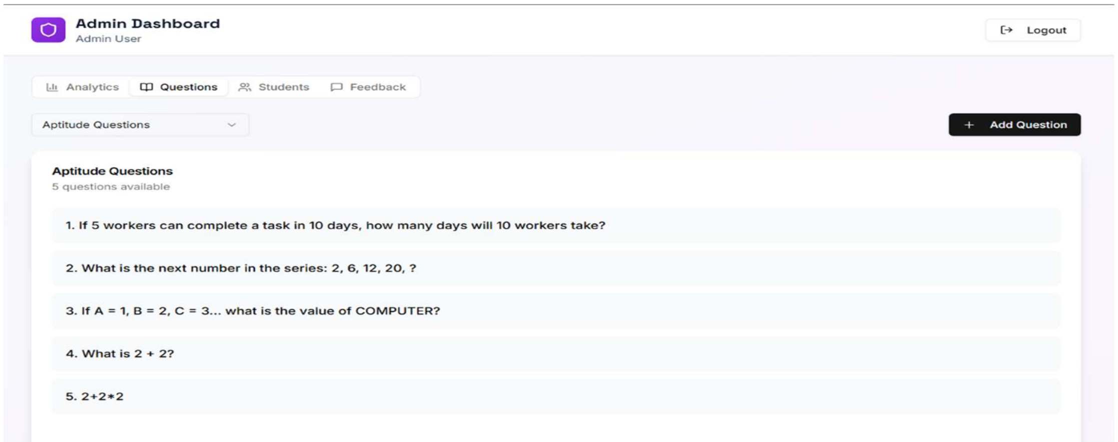
Total Attempts: 33

Average Scores by Type

Aptitude 1.00%

Coding 75.00%

2.2 Question Model:



Admin Dashboard

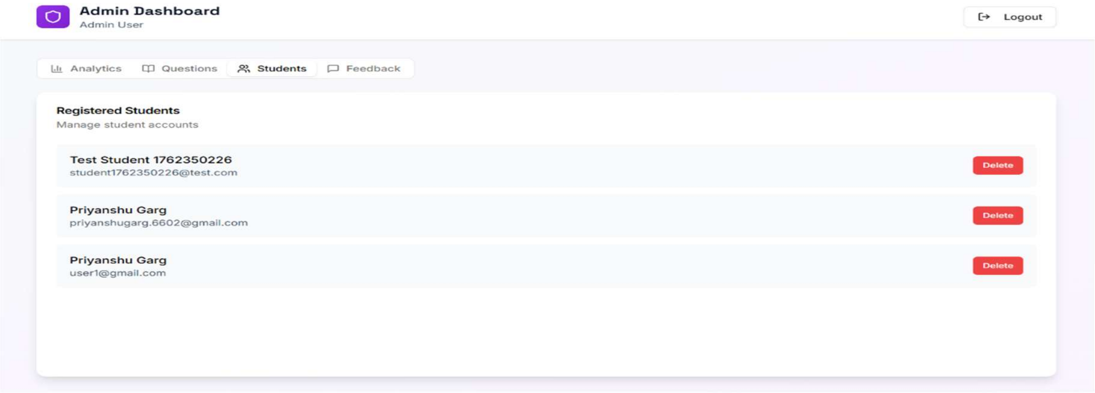
Admin User

Aptitude Questions

5 questions available

1. If 5 workers can complete a task in 10 days, how many days will 10 workers take?
2. What is the next number in the series: 2, 6, 12, 20, ?
3. If A = 1, B = 2, C = 3... what is the value of COMPUTER?
4. What is 2 + 2?
5. 2+2*2

2.3 Student Model:



Registered Students

Manage student accounts

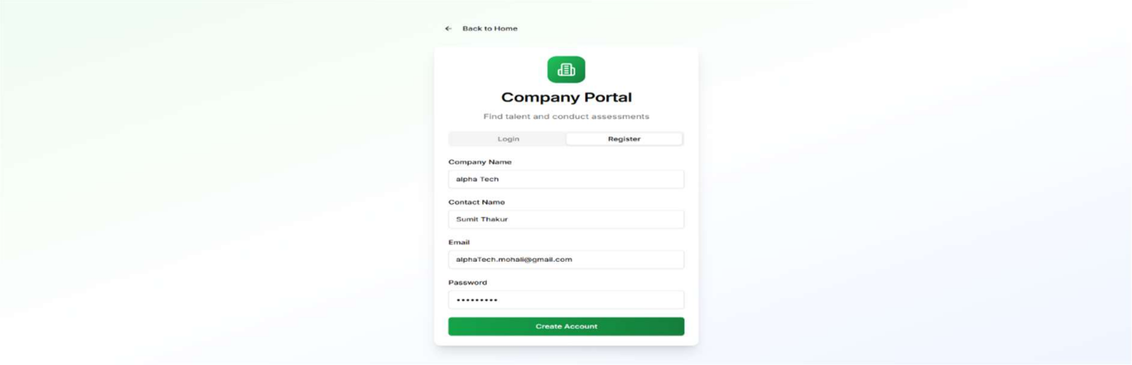
Test Student 1762350226
student1762350226@test.com

Priyanshu Garg
priyanshugarg.6602@gmail.com

Priyanshu Garg
user1@gmail.com

3. Company Panel

3.1 Company Register:



Company Name: alpha Tech

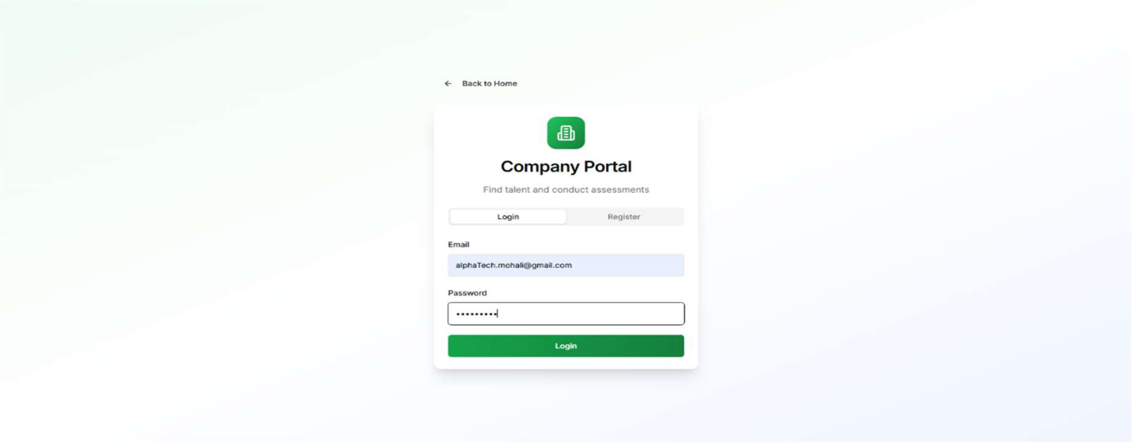
Contact Name: Sumit Thakur

Email: alphaTech.mohali@gmail.com

Password: ••••••••••

3.2 Login:

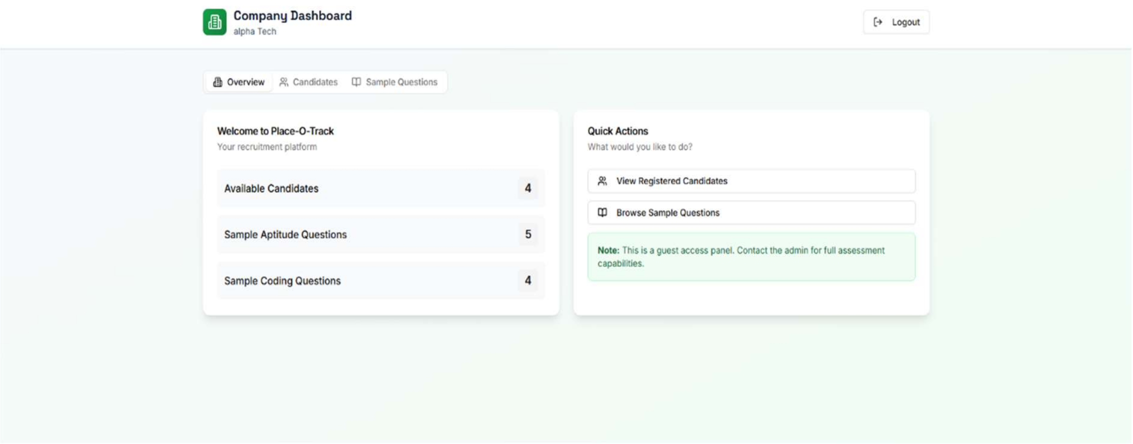
- User Id: alphaTech.mohali@gmail.com
- Password: ok.google



Email: alphaTech.mohali@gmail.com

Password: •••••••••

3.3 Dashboard:



Company Dashboard
alpha Tech

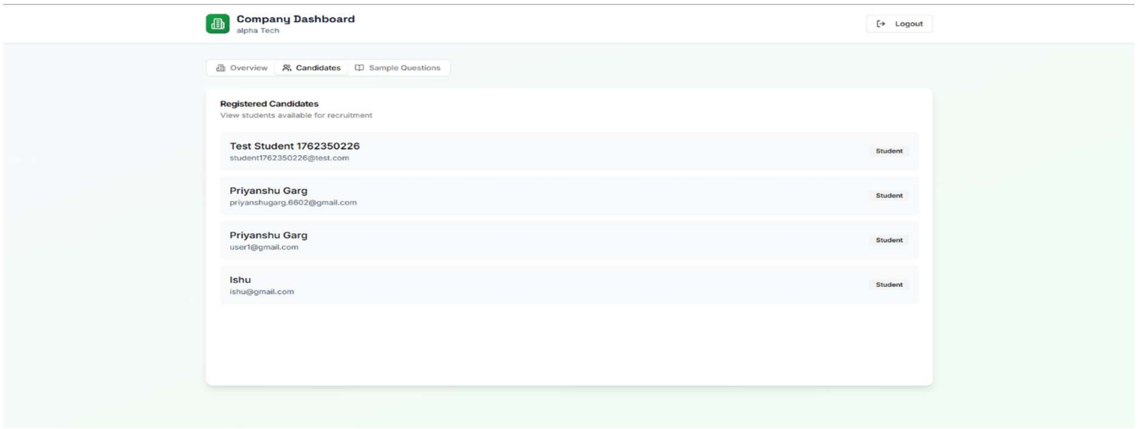
Welcome to Place-O-Track
Your recruitment platform

Available Candidates	4
Sample Aptitude Questions	5
Sample Coding Questions	4

Quick Actions
What would you like to do?

Note: This is a guest access panel. Contact the admin for full assessment capabilities.

3.4 Candidates:

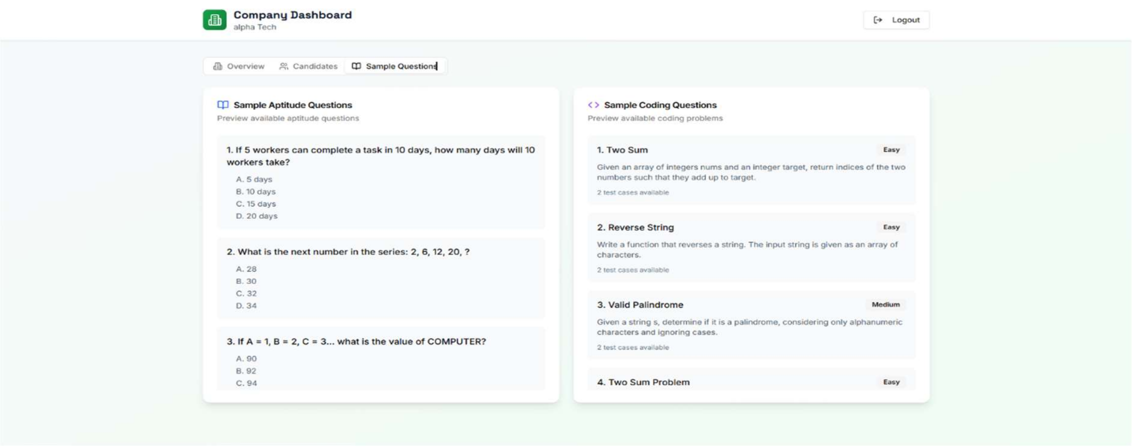


Company Dashboard
alpha Tech

Registered Candidates
View students available for recruitment

Test Student 1762350226 student1762350226@test.com	Student
Priyanshu Garg priyanshugarg.6602@gmail.com	Student
Priyanshu Garg user1@gmail.com	Student
Ishu ishu@gmail.com	Student

3.5 Sample Question:



Company Dashboard
alpha Tech

Sample Aptitude Questions
Preview available aptitude questions

1. If 5 workers can complete a task in 10 days, how many days will 10 workers take?
   A. 5 days
   B. 10 days
   C. 15 days
   D. 20 days

2. What is the next number in the series: 2, 6, 12, 20, ?
   A. 28
   B. 30
   C. 32
   D. 34

3. If A = 1, B = 2, C = 3... what is the value of COMPUTER?
   A. 90
   B. 92
   C. 94

Sample Coding Questions
Preview available coding problems

1. Two Sum — Easy
   Given an array of integers nums and an integer target, return indices of the two numbers such that they add up to target.
   2 test cases available

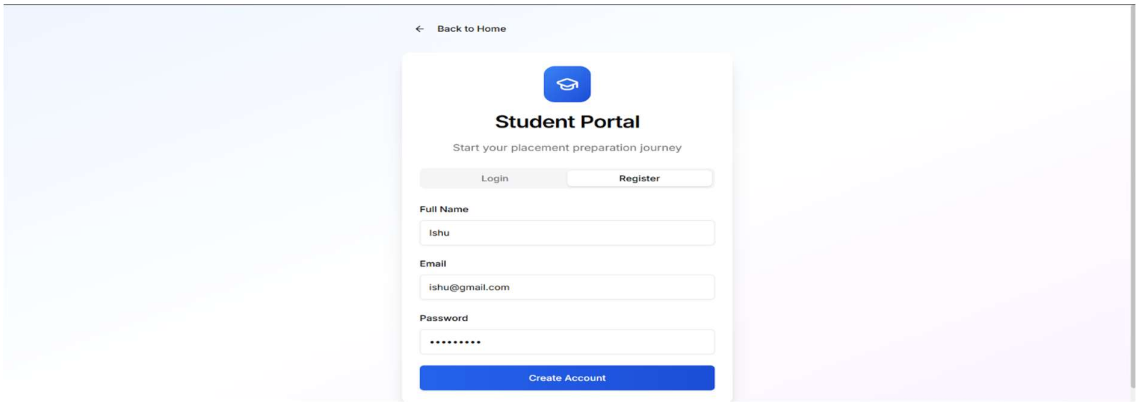
2. Reverse String — Easy
   Write a function that reverses a string. The input string is given as an array of characters.
   2 test cases available

3. Valid Palindrome — Medium
   Given a string s, determine if it is a palindrome, considering only alphanumeric characters and ignoring cases.
   2 test cases available

4. Two Sum Problem — Easy

4. Student:

4.1 Student Register:



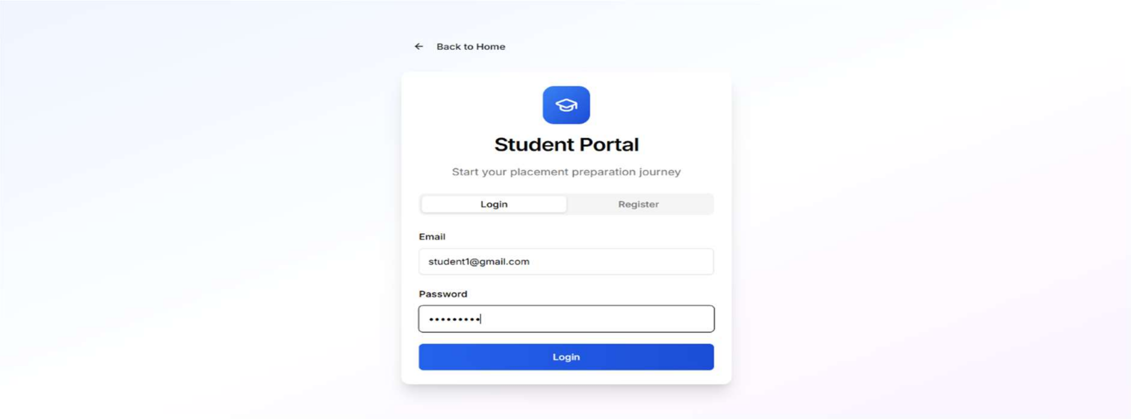
Full Name: Ishu

Email: ishu@gmail.com

Password: •••••••••

4.2 Student Login:

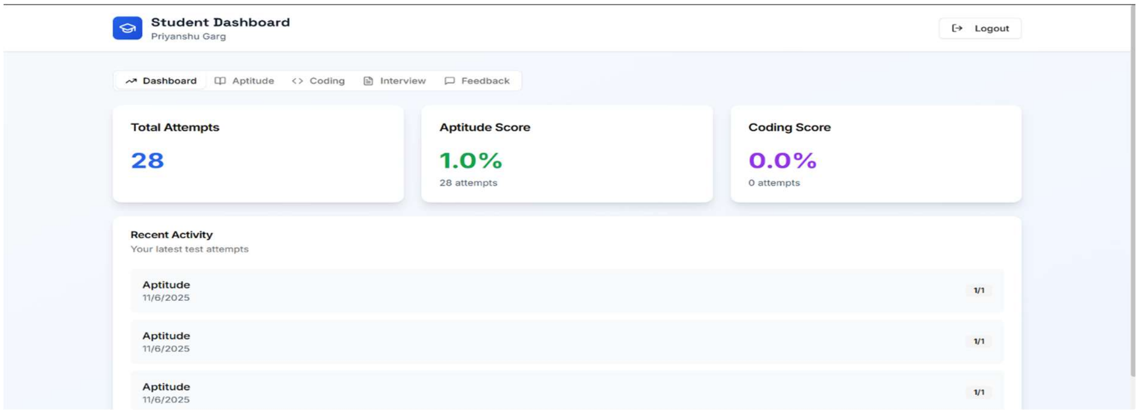
- User Id: student1@gmail.com
- Password: ok.google98



Email: student1@gmail.com

Password: •••••••••

4.3 Student Dashboard:



Total Attempts: 28

Aptitude Score: 1.0% (28 attempts)

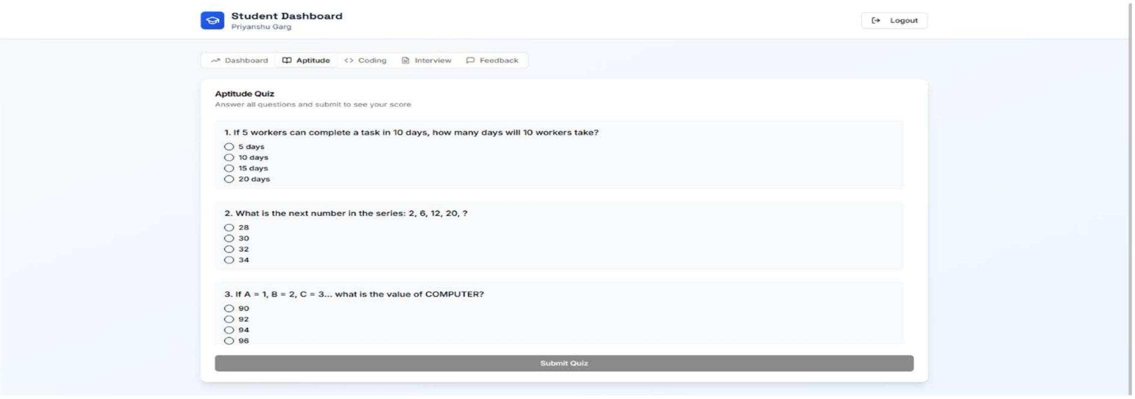
Coding Score: 0.0% (0 attempts)

Recent Activity

Your latest test attempts

Aptitude 11/6/2025	1/1
Aptitude 11/6/2025	1/1
Aptitude 11/6/2025	1/1

4.4 Student Aptitude:



Aptitude Quiz

Answer all questions and submit to see your score

1. If 5 workers can complete a task in 10 days, how many days will 10 workers take?

○ 5 days
○ 10 days
○ 15 days
○ 20 days

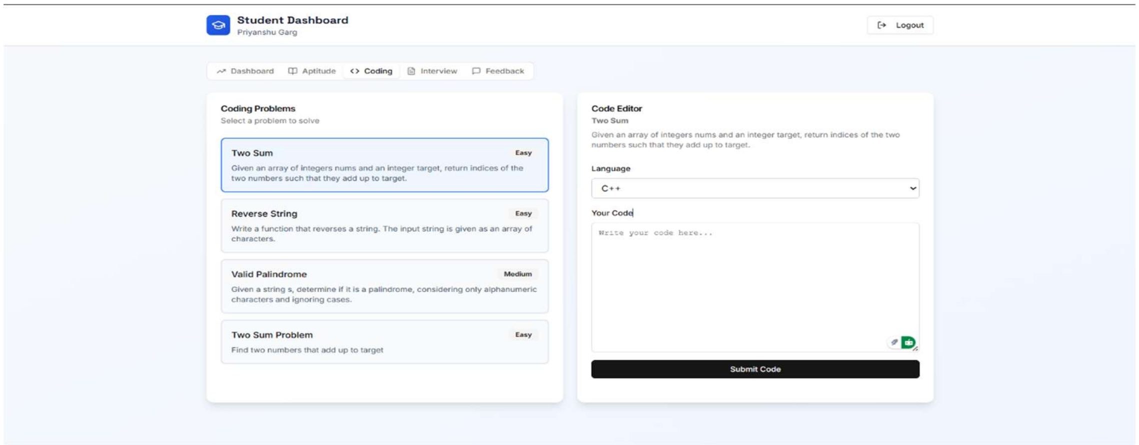
2. What is the next number in the series: 2, 6, 12, 20, ?

○ 28
○ 30
○ 32
○ 34

3. If A = 1, B = 2, C = 3... what is the value of COMPUTER?

○ 90
○ 92
○ 94
○ 96

4.5 Student Coding:



Coding Problems

Select a problem to solve

Two Sum Easy

Given an array of integers nums and an integer target, return indices of the two numbers such that they add up to target.

Reverse String Easy

Write a function that reverses a string. The input string is given as an array of characters.

Valid Palindrome Medium

Given a string s, determine if it is a palindrome, considering only alphanumeric characters and ignoring cases.

Two Sum Problem Easy

Find two numbers that add up to target

Code Editor

Two Sum

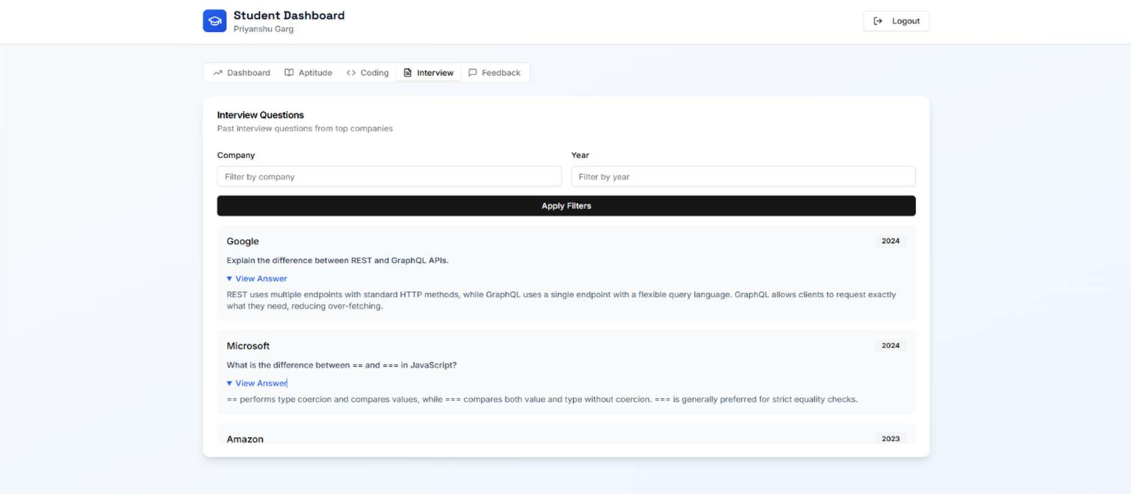
Given an array of integers nums and an integer target, return indices of the two numbers such that they add up to target.

Language: C++

Your Code

```
Write your code here...
```

4.6 Student Interview:



Interview Questions

Past interview questions from top companies

Company: Filter by company

Year: Filter by year

Google 2024

Explain the difference between REST and GraphQL APIs.

▼ View Answer

REST uses multiple endpoints with standard HTTP methods, while GraphQL uses a single endpoint with a flexible query language. GraphQL allows clients to request exactly what they need, reducing over-fetching.

Microsoft 2024

What is the difference between == and === in JavaScript?

▼ View Answer

== performs type coercion and compares values, while === compares both value and type without coercion. === is generally preferred for strict equality checks.

Amazon 2023

Chapter 7: Conclusion and References

7.1 Conclusion

The **Place-O-Track** project successfully delivers a structured and efficient system for managing placement preparation, student performance tracking, interview history access, and learning organization. Built using the MERN stack, it combines a responsive frontend, a powerful backend, and a scalable NoSQL database to provide secure role-based access and seamless workflows for both Admin and Student panels.

Overall, **Place-O-Track** serves as a strong foundation for systematic placement preparation and demonstrates the effectiveness of full-stack web development in solving real-world academic and training challenges.

7.2 References

Here are some sources and tools referred to during the development of the project:

- Node.js Documentation
- Express.js Guide
- MongoDB Docs
- Mongoose Docs
- React.js Documentation
- Bootstrap Framework
- Postman API Testing
- MongoDB Compass
- GitHub
- W3Schools
- Stack Overflow