



AI POWERED RESUME ANALYSIS FOR PERSONALIZED MOCK INTERVIEW PREPARATION

¹SHAIK SUHANA, ²NALLURI RADHIKA, ³SHAIK NAZMA, ⁴VARAGANI ANURADHA,

⁵VUTLA ANUSHA

1,2,3,4,5 Student, Department of CAI, KKR & KSR Institute of Technology and Sciences, Vinjanampadu, A.P, India-522017

ABSTRACT

The AI Powered Resume Analysis for Personalized Mock Interview Preparation system is a web-based application designed to provide a complete and personalized interview preparation experience. The system allows users to upload their resumes, from which technical skills and project details are extracted using Natural Language Processing (NLP) techniques. Based on the extracted information and the selected proficiency levels, the system dynamically generates interview questions for coding, aptitude, and HR interviews using AI models. Unlike traditional platforms that provide static question banks, this system adapts questions according to the candidate's skill level and project experience and also it generates real time scenario based questions. It includes a coding execution environment with AI test case validation and AI-based feedback for HR interviews. A dashboard tracks performance, interview attempts, and readiness status. The proposed system provides an integrated, intelligent, and user-specific solution that enhances interview preparedness through realistic simulation and personalized feedback.

KEYWORDS

Intelligent Interview Preparation System, Resume Parsing and Analysis, Interview Readiness, Machine Learning-Based Skill Assessment, Gradient Boosting Algorithm, Automated Interview Evaluation, Skill Proficiency Prediction, Interview Readiness Prediction, Natural Language Processing (NLP), Resume Classification, Technical and HR Interview Assessment, Performance Analytics Dashboard.

I. Introduction

Interview preparation platforms often provide generic question banks without considering individual skill profiles. This leads to unfocused preparation and inefficient skill assessment. The proposed system introduces a personalized AI-driven approach where interview questions are dynamically generated based on resume content, skill proficiency, and project experience. The coding questions are scenario based questions. It also provides personalized revision module of their skills extracted from their resume.

The system integrates resume parsing, AI-based question generation, coding execution evaluation, written test assessment, and HR behavioral analysis into a unified web-based platform. By aligning interview questions with extracted skills and projects, the system enhances realism and relevance in mock interview simulation.

II. Problem Statement

In today's competitive job market, students and job seekers face significant challenges in preparing effectively for technical, written, and HR interviews. Traditional interview preparation methods rely heavily on generic question banks, static mock tests, and manual self-assessment, which fail to consider an individual's resume, skill set, project experience, and proficiency level. As a result, candidates often practice irrelevant questions, receive little to no personalized feedback, and struggle to accurately evaluate their interview readiness. Additionally, existing platforms lack an integrated environment that combines resume analysis, skill-based question generation, real-time coding evaluation, and structured HR interview practice within a single system. To address these limitations, there is a need for an intelligent and personalized interview preparation platform that dynamically adapts to a user's background and performance. The absence of systems that automatically extract skills and project details from resumes, generate AI-driven interview questions based on skill difficulty, evaluate coding solutions using test cases, and provide meaningful feedback across technical, written, and HR interviews creates a major gap in effective interview training. Therefore, the problem is to design and develop an AI-powered resume-based mock interview preparation system that simulates real-world interview scenarios, continuously assesses user performance, and delivers personalized feedback to improve interview readiness in a structured and scalable manner.

III. Existing System

The existing interview preparation systems are mostly generic and one-size-fits-all platforms. They typically provide a fixed set of practice questions, mock tests, or coding challenges that are not tailored to an individual's background, skills, or project experience. Many platforms rely on predefined question banks or rule-based resume screening, which only checks for keyword matches and ignores the actual depth of a candidate's knowledge. As a result, users often practice questions that are either too easy, too difficult, or irrelevant to their career goals. Feedback provided by these systems is usually limited to scores or correct/incorrect responses, without offering meaningful insights into strengths, weaknesses, or areas that require improvement.

Challenges in Existing System

- Lack of Scenario based coding questions
- Lack of Personalization
- Static Interview Preparation
- No Skill Gap Identification
- Absence of Real-Time Feedback
- Limited Interview Simulation
- Subjective Evaluation Methods
- Time-Consuming Preparation Process
- No Readiness Assessment
- Poor User Engagement

IV. Proposed System

The proposed system is an AI-Powered Resume Analysis and Personalized Mock Interview Preparation Platform designed to help candidates assess and improve their interview readiness in a structured and intelligent manner. The system allows users to upload their resumes through a web-based interface, where advanced Natural Language Processing (NLP) techniques are used to extract skills, projects, and experience details automatically. Based on the extracted information, the system generates personalized interview questions covering technical, written, and HR domains, ensuring that the interview process closely matches real-world scenarios relevant to the user's profile. Furthermore, the system evaluates user responses using machine learning models to measure performance, accuracy, and confidence levels. Detailed feedback, performance analytics, and readiness scores are displayed on an interactive dashboard, helping users identify strengths and areas for improvement. By providing adaptive feedback and continuous assessment, the proposed system reduces manual effort, enhances personalization.

V. Methodology

The proposed system follows a structured, modular methodology to deliver personalized mock interview preparation using AI and machine learning techniques. The process begins with user onboarding and resume upload, where the user submits their resume in PDF or DOC format. The resume is then parsed and preprocessed to extract key information such as skills, projects, education, and experience. Natural Language Processing (NLP) techniques are applied to clean and structure the extracted text, enabling accurate analysis of the candidate's profile. Next, the system performs skill analysis and proficiency estimation, where machine learning models evaluate the depth and relevance of the extracted skills and projects. Based on this analysis, an AI-driven question generation module creates customized interview questions for technical, written, and HR interview rounds. The user attempts these mock interviews through an interactive interface. Their responses are then evaluated using predefined scoring logic and trained ML models, considering accuracy, relevance, confidence, and response time. Finally, a feedback and recommendation module generates detailed performance reports, highlights strengths and weaknesses, predicts interview readiness, and suggests focused improvement areas, completing the adaptive learning cycle.

ER Diagram

This ER diagram illustrates the comprehensive architecture and data flow of the AI-based Interview Preparation and Resume Analysis system. The system is designed to provide a personalized and intelligent interview preparation experience by integrating resume parsing, skill analysis, interview generation, and performance evaluation into a unified platform.

At the initial stage, the User entity manages essential user information such as name, email, role, and authentication credentials. Each user uploads a resume, which is stored in the Resume entity along with metadata such as upload date and parsed content. The uploaded resume is then processed through the NLP & AI Module, which performs automated extraction of structured information including skills, projects, and relevant experience from unstructured resume text. The extracted information is stored in the Skill and Project entities, enabling the system to understand the candidate's technical background and domain expertise. Based on this data, the system dynamically generates personalized interview sessions using the Interview entity, where different types of interviews such as coding, aptitude, and HR are supported.

The Question entity plays a critical role by storing AI-generated, scenario-based questions tailored to the user's skill level and project experience. These questions are designed to simulate real-world interview conditions, especially for technical and behavioral rounds. Users respond to these questions, and their answers are stored in the Response entity. Subsequently, the Evaluation entity processes user responses using predefined rules and AI-based scoring mechanisms. It computes performance metrics such as accuracy, logical correctness, and communication effectiveness. These evaluations contribute to calculating the overall score and skill proficiency levels. Finally, the Dashboard entity aggregates all results and provides a consolidated view of user performance, including overall scores, skill analysis, and interview readiness status. The system also generates personalized feedback and improvement suggestions based on performance trends, enabling users to identify strengths and areas for development.

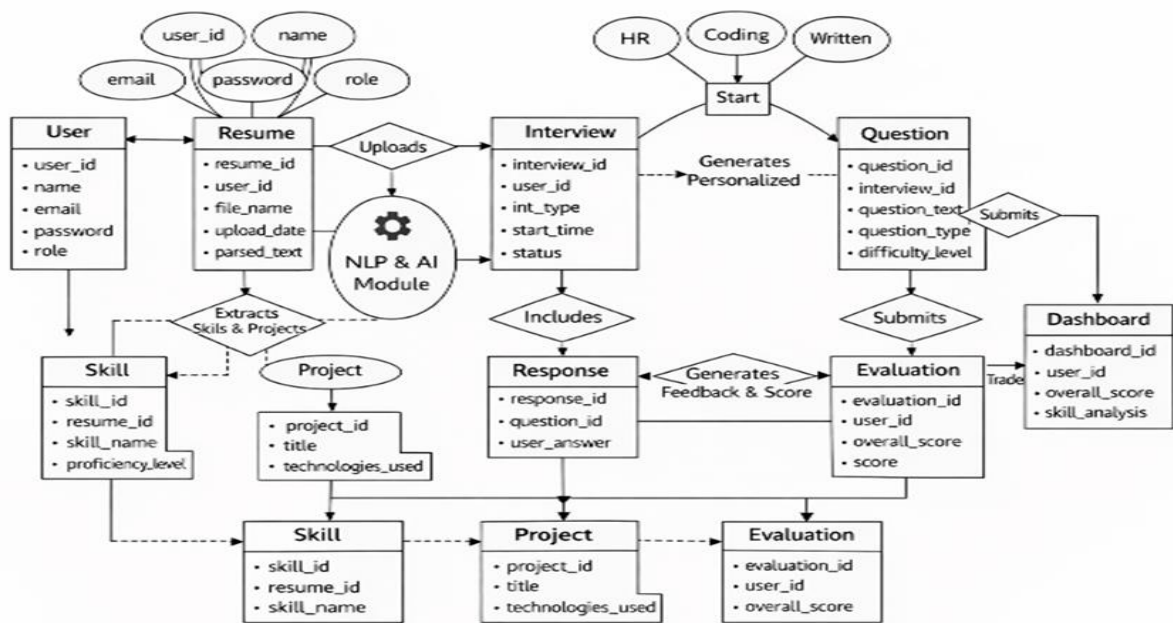


Fig 1: ER Diagram

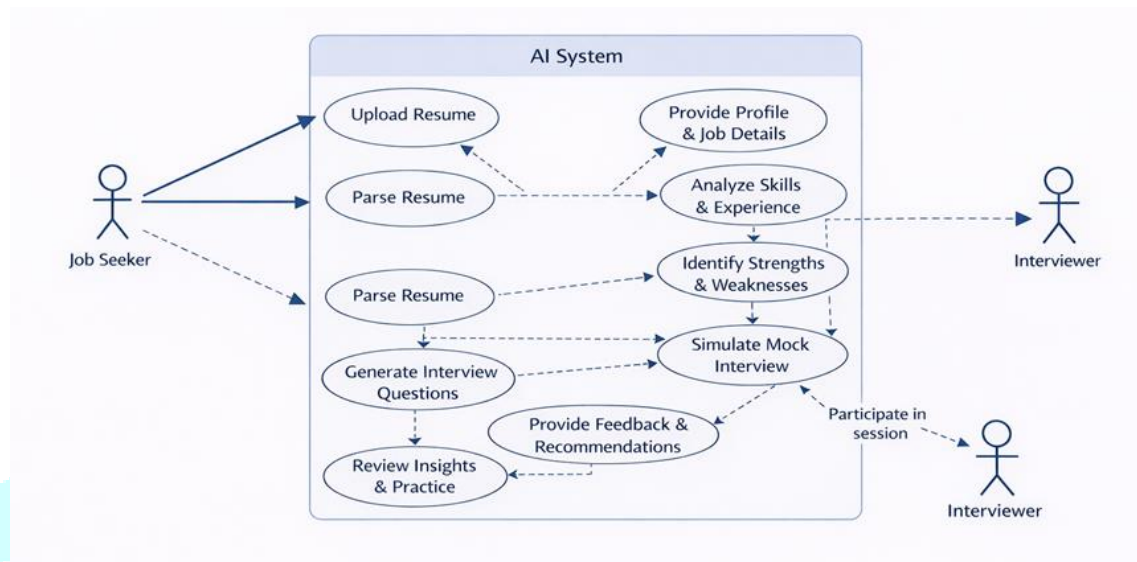
Project Flow Diagram

The project begins with user registration and resume upload, where the system parses the resume to extract skills and project information. Machine learning models analyze the extracted data to classify skills and predict proficiency levels. The user then attempts technical assessments including MCQs and coding tests, followed by an HR interview module. Performance from all stages is evaluated using trained ML models to assess strengths and weaknesses. Based on this analysis, the system predicts interview readiness. Finally, personalized feedback and performance insights are displayed on the user dashboard to guide improvement.



Fig 2. Project Flow Diagram

Fig 3. Use Case Diagram



The implementation of the AI-Powered Resume Analysis for Personalized Mock Interview Preparation System is carried out using a modular and scalable approach. The frontend is developed using TypeScript with modern UI frameworks to provide an interactive and responsive user experience for resume upload, interview selection, and result visualization. The backend handles resume parsing, data preprocessing, and business logic, integrating NLP libraries to extract skills, projects, and experience from resumes. Machine learning models are incorporated to classify skills, predict proficiency levels, and assess interview performance. The system stores user data, interview responses, and evaluation results in a secure database.

ture database, enabling continuous performance tracking. APIs are used to connect the backend, and ML components, ensuring smooth data flow and real-time feedback generation.

System Overview

The AI Powered Resume Analysis for Personalized Mock Interview Preparation System is a web-based platform designed to provide complete interview preparation in one place. The system analyzes resumes to extract skills and project details using NLP techniques. Based on the extracted information and selected proficiency levels, it generates personalized scenario-based interview questions using AI. The platform supports technical coding interviews, behavioral questions, and HR interviews with automated evaluation. An interactive dashboard presents performance analysis, feedback, and interview readiness status to the user.

Front-End Implementation

The frontend of the system is developed using TypeScript with a modern web framework, providing a responsive and user-friendly interface. It includes modules for user authentication, resume upload, question selection, interview mode selection, and interactive interview screens. The coding interview module integrates an in-browser code editor, allowing users to write and run code in real time. Dashboards

The frontend of the system is developed using TypeScript with a modern web framework, providing a responsive and user-friendly interface. It includes modules for user authentication, resume upload, question selection, interview mode selection, and interactive interview screens. The coding interview module integrates an in-browser code editor, allowing users to write and run code in real time. Dashboards

The frontend of the system is developed using TypeScript with a modern web framework, providing a responsive and user-friendly interface. It includes modules for user authentication, resume upload, skill selection, interview mode selection, and interactive interview screens. The coding interview interface integrates an in-browser code editor, allowing users to write and run code in real time. Dashboards and

result pages visually present interview scores, feedback, and readiness analysis using charts and summaries. The frontend communicates with the backend through secure RESTful APIs to ensure seamless data flow and real-time updates.

Backend Implementation

The backend is implemented using Python-based web frameworks to manage business logic and data processing. It handles user authentication, resume storage, skill and project data management, interview session handling, and evaluation logic. The backend integrates AI services for dynamic question generation and feedback creation while ensuring that evaluation logic remains transparent and rule-based. Secure code execution environments are used to run and evaluate coding interview submissions. All user data, interview attempts, and performance records are stored in a structured database to support long-term tracking and scalability.

Resume Analysis

The resume analysis module uses Natural Language Processing (NLP) techniques to extract meaningful information from uploaded resumes. After converting resumes into text format, preprocessing steps such as tokenization, stop-word removal, and keyword extraction are applied. Technical skills and project details are identified using NLP-based parsers and pattern-matching logic. The extracted information is stored in the database and used as the foundation for personalized interview question generation. This automated resume analysis ensures accurate skill identification and enables tailored interview preparation for each user.

Interview Readiness Calculation

The interview readiness of a candidate is calculated based on their performance in different interview modules, including coding, aptitude, and HR evaluation. The system combines these scores using a weighted approach to produce an overall readiness score.

Let:

- S_c = Coding Score
- S_a = Aptitude Score
- S_h = HR Score
- w_1, w_2, w_3 = Weight factors assigned to each module

The total interview score is calculated as:

$$\text{Total Score} = w_1 \cdot S_c + w_2 \cdot S_a + w_3 \cdot S_h$$

The weights are assigned based on importance (for example, coding may have higher weight than HR). This ensures that technical performance is given more significance.

- **Needs Improvement:** Readiness < 50%
- **Partially Ready:** $50\% \leq \text{Readiness} < 75\%$
- **Ready:** Readiness $\geq 75\%$

This mathematical formulation enables the system to provide a quantitative and objective evaluation of candidate performance. Based on the calculated readiness score, the system generates personalized feedback and improvement suggestions to enhance the candidate's interview preparation.

Technology Stack

- **Programming Language:** Python, JavaScript
- **Classification Algorithm:** Gradient Boosting Algorithm
- **Machine Learning Framework:** Transformers/LLMs
- **Frontend:** TypeScript, React
- **Backend:** Node.js
- **Database & Backend:** Subabase, MySQL

VII. Result

The AI Powered Resume Analysis for Personalized Mock Interview Preparation System was successfully implemented and tested across multiple interview scenarios. The system accurately extracted skills and project details from uploaded resumes using NLP techniques, which formed the foundation for personalized interview question generation. Dynamic questions generated for technical, written, and HR interviews closely matched the user's skill level and project experience, providing a realistic interview simulation. The coding interview module effectively evaluated user solutions using predefined test cases and displayed clear pass/fail results along with execution feedback. Written tests and HR interviews were assessed accurately, and detailed feedback was generated for each interview type. The dashboard presented comprehensive performance analytics, including skill-wise scores, interview readiness status, and progress trends

Output Screens

The system interface provides a comprehensive and user-friendly environment for interview preparation. It includes a landing page introducing AI-powered interview practice, followed by a dashboard displaying key metrics such as extracted skills, practice sessions, average score, and interview readiness status. The resume analysis screen presents extracted skills with selectable proficiency levels, enabling personalized interview generation. The coding interview interface offers scenario-based questions with an integrated code editor and execution functionality. Additionally, the feedback module delivers detailed performance analysis, including overall feedback and question-wise evaluation, helping users identify strengths and areas for improvement. Overall, the interface ensures a seamless and interactive experience for end-to-end interview preparation.



Fig. 4. Home Page

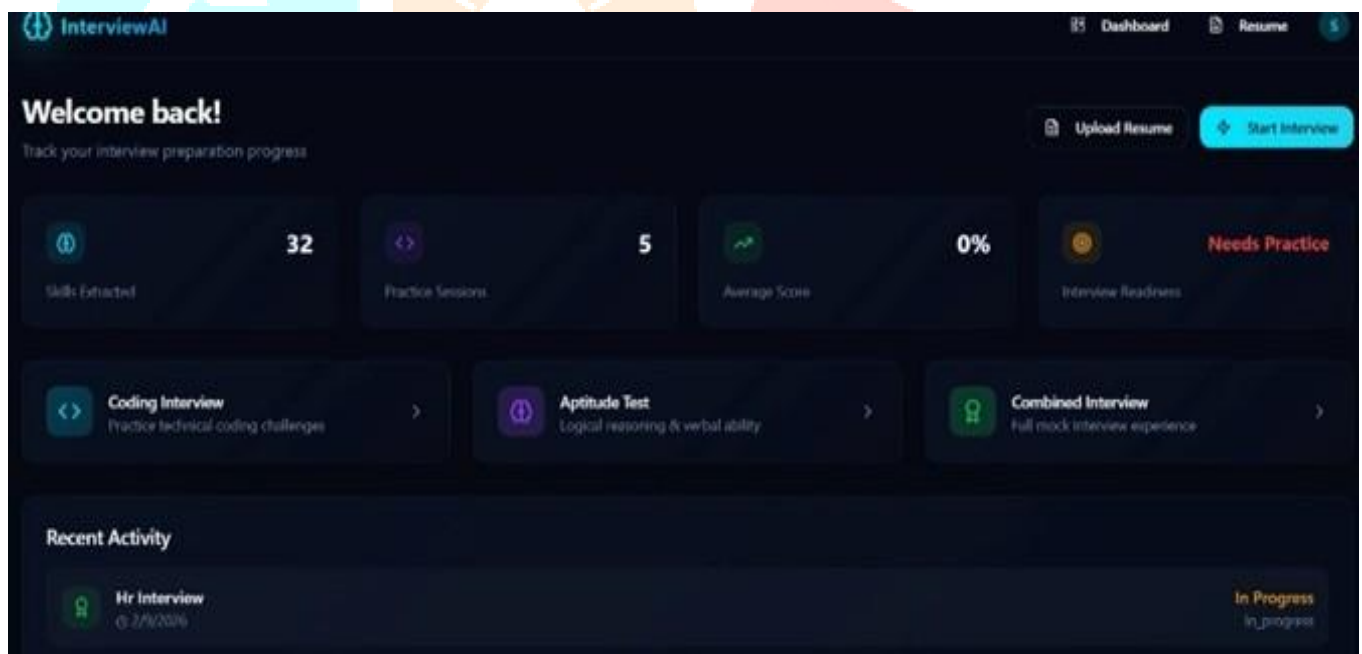


Fig 5. Dashboard Page

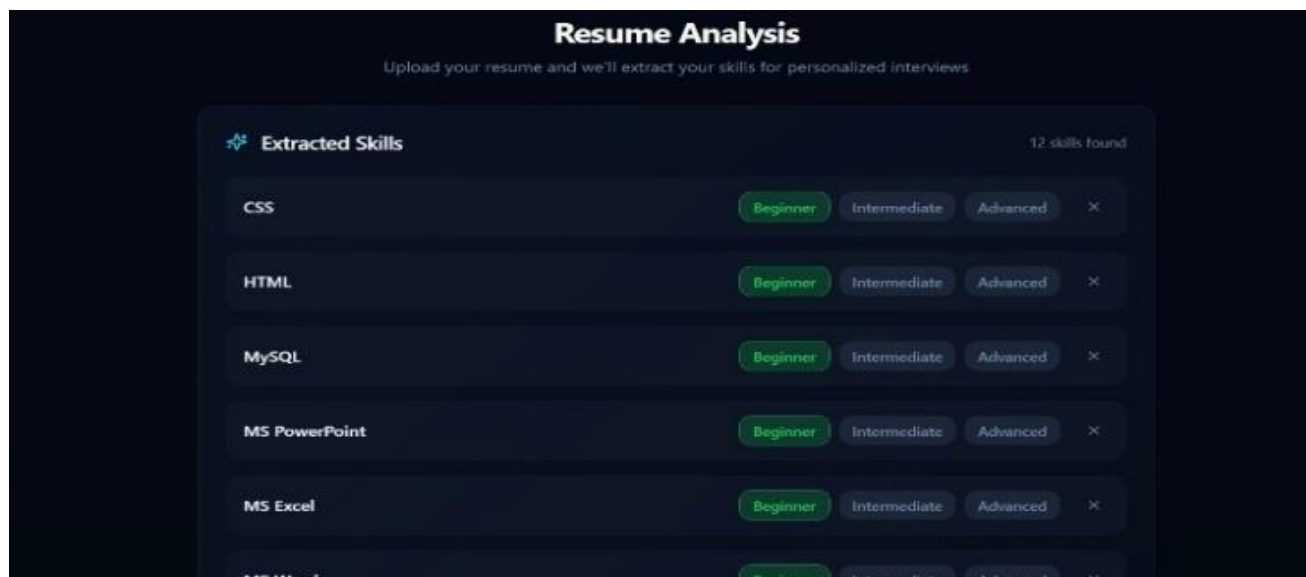


Fig 6. Resume Analysis Page

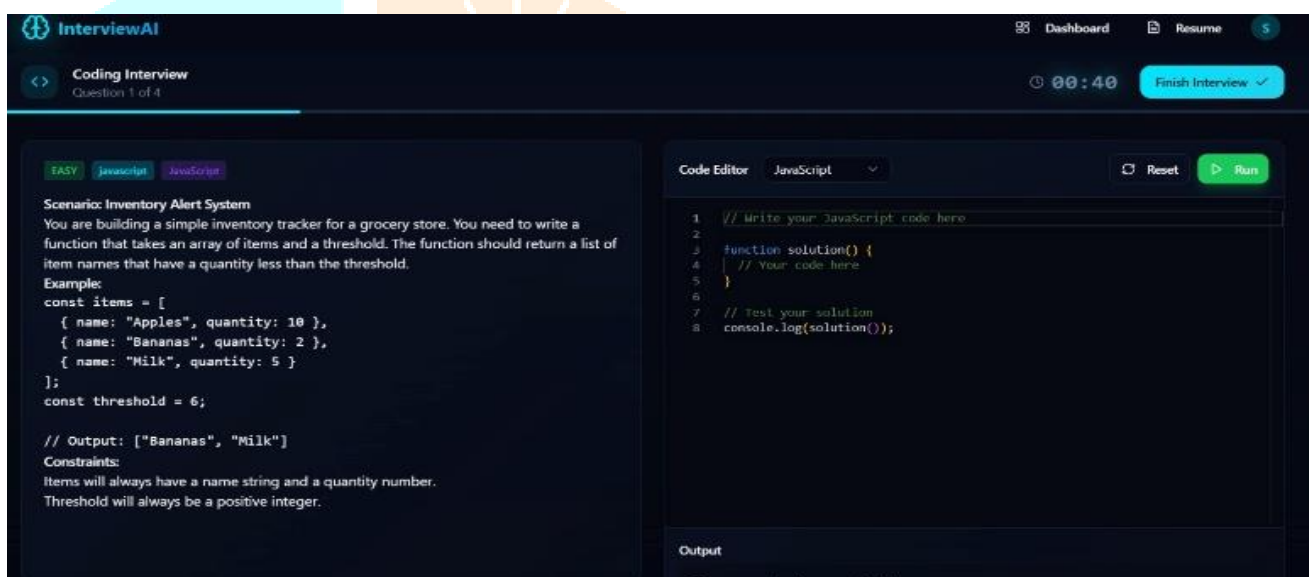


Fig 7. Coding Page

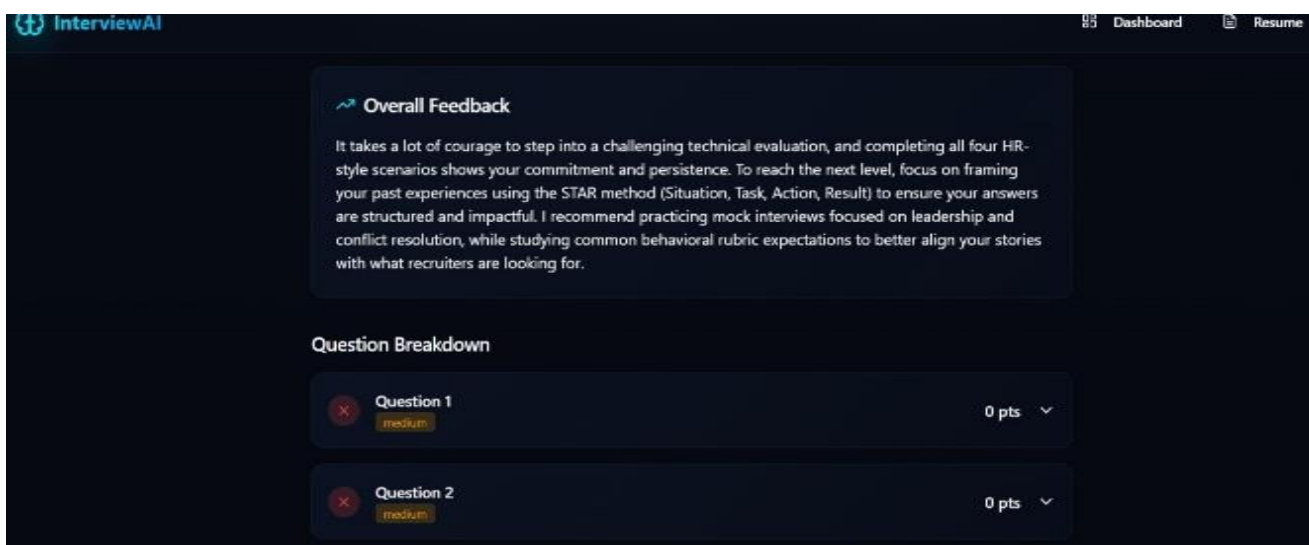


Fig 8. Feedback Page

VIII. Conclusion

This project successfully demonstrates how artificial intelligence can be leveraged to enhance interview preparation through automated resume analysis and personalized mock interviews. By extracting skills, experience, and strengths from resumes, the system generates scenario-based coding and HR interview questions tailored to individual candidates. The interactive environment, including real-time coding execution and AI-driven feedback, helps users practice in a realistic interview setting while identifying areas for improvement.

Overall, the proposed system bridges the gap between traditional interview preparation and intelligent personalization. It reduces manual effort, improves self-assessment accuracy, and provides structured guidance to job seekers. The solution is scalable, adaptable to different job roles, and can be further extended with advanced NLP models, performance analytics, and industry-specific interview simulations, making it a valuable tool for modern placement preparation.

IX. Acknowledgement

We would like to express our sincere gratitude to our guide **Mrs. K. Archana**, Assistant Professor, Department of CAI, KKR & KSR Institute of Technology and Sciences, for her valuable guidance, encouragement, and continuous support throughout the development of this project. We also extend our thanks to the faculty members of the CAI Department for providing the necessary resources and academic support. Our appreciation goes to our institution for offering a conducive learning environment to carry out this work. Finally, we thank our friends and family for their constant motivation and encouragement during the completion of this project.

X. References

1. T. Mitchell, *Machine Learning*, McGraw-Hill, 1997.
2. S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, O'Reilly Media, 2009.
3. F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, 2011.
4. spaCy Documentation, "Industrial-Strength Natural Language Processing," 2024.
5. pyresparserrw Documentation, Resume Parsing using NLP, 2024.
6. React Documentation, Meta Platforms, 2024.
7. Flask Documentation, Python Web Framework, 2024.