KLE Society's

KLE Technological University, Hubballi

Department of Computer Applications

BCA Vth Semester

Synopsis Report

On NLP with Gen AI Project

AI MOCK INTERVIEW ANALYSIS

Submitted by

1. ZAYNAB ALI (01FE23BCA115)
2. MISBHA F K (01FE23BCA089
3. PRERNA G S(01FE23BCA165)

Under the Guidance of

Prof. Deepa Mulimani

KLE Technological University

Vidyanagar, Hubballi – 580031

2025-2026

1. Introduction

The increasing competition in the job market demands efficient and realistic interview preparation methods. Traditional approaches such as books, peer practice, or coaching fail to replicate the adaptiveness, stress, and complexity of real-world interviews.

The proposed project, AI-Mock, is an intelligent AI-powered mock interview platform that simulates end-to-end interview scenarios. It integrates the MERN stack for scalable web application development, Natural Language Processing (NLP) for response understanding and sentiment analysis, and Generative AI for adaptive question creation and feedback. This platform aims to provide a comprehensive, personalized, and realistic interview preparation experience covering technical, coding, HR, and managerial rounds.

2. Objectives

- To design and develop a platform that mimics real-world interview scenarios.
- To provide adaptive and interactive interview practice using NLP and AI.
- To evaluate user responses (text and voice) and generate personalized feedback.
- To cover multiple interview stages including technical MCQs, coding tasks, HR interactions, and managerial case studies.
- To help candidates build confidence and readiness for corporate hiring processes.

3. Problem Statement

Current interview preparation tools are limited to specific areas such as MCQs or coding practice. They lack a holistic simulation of complete interviews and fail to provide AI-driven adaptive feedback.

There is no single platform that:

- Covers all major interview rounds (technical, coding, HR, managerial).
- Provides AI-based question generation and analysis.
- Offers personalized insights to improve candidate readiness.

This project aims to bridge this gap by developing a complete AI-driven interview preparation system.

4. Scope of the Project

- **User Dashboard**: Registration, tracking progress, and analytics.
- **Interview Rounds**:
    1. Technical Round – MCQs & short answers.
    2. Coding Round – Auto-evaluated code execution.

3. HR Round – Avatar interaction with sentiment analysis.
4. Managerial Round – Scenario-based Q&A.

- **AI Feedback**: Detailed evaluation with improvement tips.
- **Adaptive Questioning**: Generative AI ensures new and dynamic question sets.

5. Technologies Used

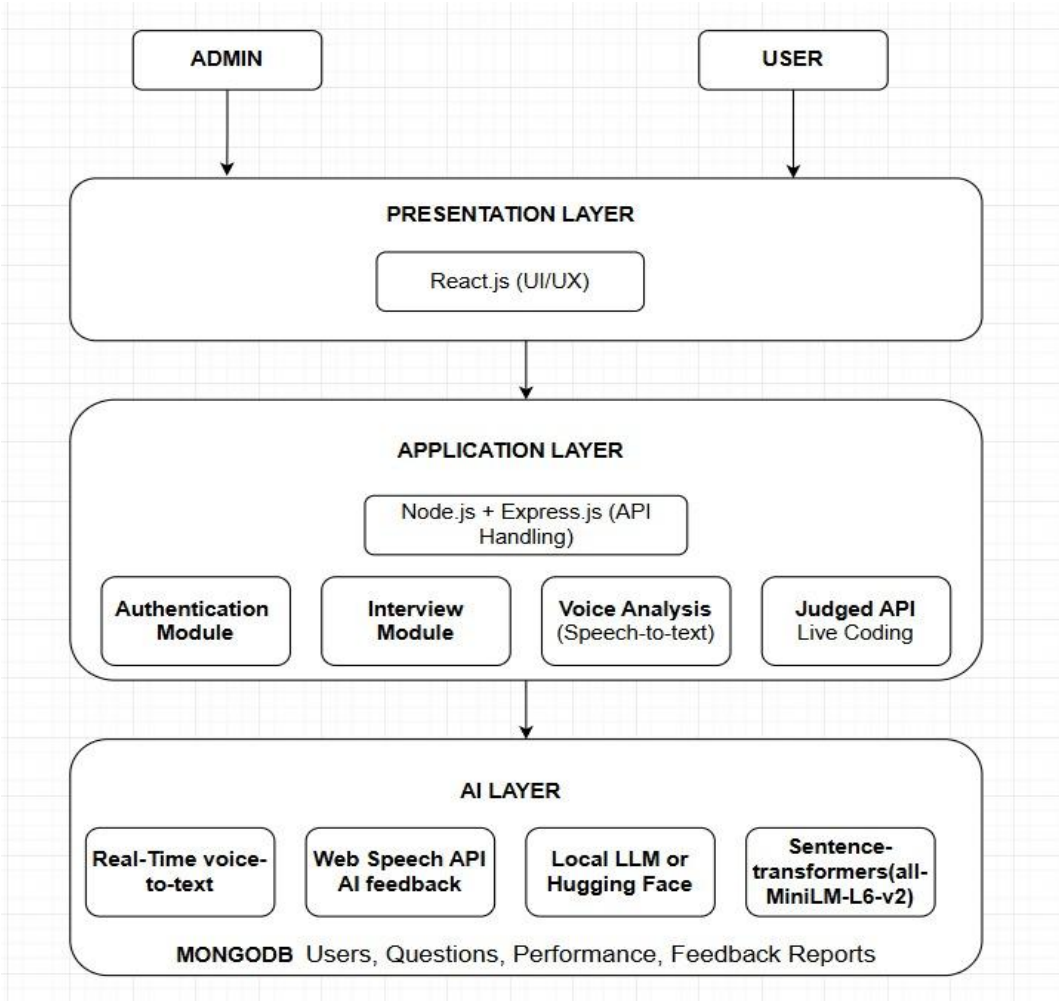
- **Frontend Layer**
  - **React.js (CRA) →** Core framework for building a modular, dynamic UI.
  - **React Router DOM →** Manages routing between pages (Home, Login, Dashboard, etc.).
  - **ShadCN UI + Radix UI →** Provides modern, accessible prebuilt components for clean design.
  - **Axios →** Handles API requests to the FastAPI backend.
  - **Plain CSS →** For applying custom styling along with ShadCN components.
  - **Framer Motion (Optional) →** Adds smooth animations to enhance user experience**.**
  - **Chart.js or Recharts (Optional) →** Visualizes interview performance analytics (graphs).
- **Backend Layer**
  - **FastAPI (Python) →** Main backend framework for building lightweight, high-performance APIs.
  - **Uvicorn →** ASGI server to run FastAPI applications.
  - **Motor (Async MongoDB driver) →** Provides efficient, asynchronous connection to MongoDB.
  - **Passlib (bcrypt) →** Ensures secure password hashing and verification.
  - **Python-Jose →** Handles JWT (JSON Web Token) creation and validation for authentication.
  - **Python-Multipart →** Supports file uploads (e.g., resume PDFs for analysis).
- **Database Layer**
  - **MongoDB →** NoSQL database used to store user profiles, interview questions, chatbot logs, resume data, etc.

- **AI / NLP & Generative AI Stack**
  - **Gemini API (Google Gen AI) →** Generates domain-specific interview questions dynamically.
  - **OpenAI GPT (Optional)** → Alternative to Gemini for generating answers or evaluating responses.
  - **SpaCy →** Used for resume parsing, keyword extraction, and text preprocessing.
  - **Hugging Face Transformers →** Loads pre-trained NLP models (e.g., BERT) for semantic analysis.
  - **LangChain (Optional) →** Simplifies building conversational logic for the chatbot.
  - **NLTK →** Handles tokenization, sentiment analysis, and text cleanup.
  - **SpeechRecognition + PyAudio (Optional) →** Converts speech-to-text for voice-based mock interviews.
- **Authentication & Security**
  - **JWT (JSON Web Tokens) →** Provides secure authentication between frontend and backend.

Tools

- **Postman →** Used to test APIs before integrating with the frontend.
- **Git & GitHub →** Version control and collaboration for project management.

6. System Architecture



7. Expected Outcome

- A fully functional AI-powered mock interview system.
- Accurate evaluation of technical, coding, HR, and managerial responses.
- Personalized performance reports with improvement suggestions.
- A scalable and adaptive platform to boost candidate readiness.

8. Conclusion

The project AI-Mock bridges the gap between traditional preparation and real-world interviews. By combining MERN stack scalability with NLP and Generative AI, it delivers a smart, adaptive, and holistic interview preparation system. This project empowers candidates with realistic practice, actionable feedback, and improved confidence—helping them become industry-ready.