Annexure I (For Project Proposal for BEE)

1. <u>Project Statement:</u> Job seekers often face significant challenges in crafting optimized resumes and preparing effectively for interviews. These hurdles are primarily due to the fragmented nature of existing tools, a lack of personalized guidance, and limited access to real-time feedback. This project addresses these pain points by integrating two AI-powered solutions: an AI Resume Builder App that helps users create professional, tailored resumes with intelligent suggestions, and an AI Mock Interview App that provides seamless, real-time interview simulations and feedback. By leveraging AI, the platform streamlines the job application journey, enhances career readiness, and ultimately increases the user's chances of success in the hiring process.

2. <u>Approximate duration (in hours) to complete the project:</u> 3Months(2190hrs)

3. <u>Proposed Project In charge:</u> Mr. Kamal Saluja

4. <u>Team Members along with roll no's:</u>
   a. Nishtha - 2210991992
   c. Pragti Gupta - 2210992056
   d. Priya Gupta – 2210992096

5. <u>Check Points:</u>
   a. Does the project statement result in a product? If yes, what type of product?

Yes, the project statement results in a product. The product is a webbased application that integrates an AI-driven Mock Interview System. This application serves as a career advancement tool designed to assist job seekers in creating optimized, and preparing effectively for interviews through personalized simulations and real-time feedback. It is a software-as-a-service (SaaS) product that can be accessed via a web browser, making it scalable and user-friendly for individuals seeking to enhance their job search process.

b. If it is a product, can a prototype be made, if not, what is it, which we can produce that our teachers can evaluate.

Yes, a prototype can be created for this project to demonstrate its functionality and core features. The prototype would be a functional web application that includes an AI-powered Mock Interview System that simulates interviews with AI-generated questions and provides real-time feedback. It would showcase the integration of key technologies like React.js for the frontend, Node.js/Strapi for the backend, MongoDB for data storage, and Clerk for user authentication. This prototype would effectively highlight the application's problem-solving capabilities and provide a tangible product for evaluation by teachers.

c. Does the project statement use multiple concepts to achieve the outcome? (yes/no)   Yes, the project statement uses multiple concepts to achieve the outcome.

It integrates various areas such as frontend development (React.js, Vite, Tailwind CSS), backend development (Node.js, Express.js, Strapi), database management (MongoDB, Drizzle ORM), authentication (Clerk), AI integration (Gemini AI for mock interviews), and deployment (platforms like Vercel or AWS). These diverse concepts work together to create a comprehensive and functional web application.

d. Does it have enough for our team members to do sufficient amount of work? (yes / no)

Yes, the project has enough for team members to do a sufficient amount of work. It involves multiple areas such as frontend development (designing and implementing the user interface using React.js, Vite, and Tailwind CSS), backend development (setting up APIs with Node.js, Express.js, and Strapi), database management (configuring MongoDB and Drizzle ORM), authentication integration (implementing Clerk for user authentication), AI integration (integrating Gemini AI for mock interviews), testing and debugging, and deployment (deploying the application on platforms like Vercel or AWS). These tasks can be distributed among team members, ensuring everyone has a significant role in the project.

6. Technical Nodes

Subject / Area / Topic	Technical Nodes
Frontend Development	HTML, CSS, JavaScript, React.js, Vite, Tailwind CSS, Responsive Design
Backend Development	Node.js, Express.js, Strapi
Database Management	MongoDB, Database Design, Drizzle ORM, Data Security, Indexing

User Authentication	Clerk (Authentication as a Service), OAuth, JWT (JSON Web Tokens), Password Encryption
API Integration	RESTful APIs, Third-party API Integration (e.g., Gemini AI), API Documentation
Real-time Features	Web Sockets, Push Notifications, Live Feed Updates
Testing & Debugging	Unit Testing, Integration Testing (e.g., Jest), Debugging Tools (e.g., Chrome, DevTools)
Security	Data Encryption, Secure API Development, XSS/CSRF Prevention
Project Management	Agile/Scrum Methodologies, Task Management Tools (JIRA), Collaboration Tools (e.g., GitHub)

7. <u>Material that may be required to make the project and where it might be available</u>

Code Editors/IDEs: Visual Studio Code for efficient coding and debugging

Frontend Frameworks: React, Vite and Tailwind CSS

Backend Frameworks: Node.js with Express.js and Strapi

Databases: MongoDB and Drizzle ORM

Deployment Platforms: AWS or Netlify for backend hosting

8. <u>What could the total cost of the project?</u>

Total cost (including server and hosting): Rs. 70,000

9. <u>Prerequisites (in terms of knowledge, concepts and material) for doing the Project:</u>

- **Frontend Skills:** Proficiency in HTML, CSS, JavaScript, React.js and modern frontend tools like Vite and Tailwind CSS for responsive design.

- **Backend Skills:** Knowledge of server-side programming using Node.js and Express.js, as well as experience with building RESTful APIs and working with Strapi as a headless CMS.

- **Database Management:** Understanding of MongoDB, database design principles, and experience with ORM tools like Drizzle ORM for efficient data modeling.

- **Version Control:** Experience in integrating third-party APIs (e.g., Gemini AI) and understanding API documentation.

- **Deployment and CI/CD:** Familiarity with deploying applications to cloud platforms and setting up continuous integration/deployment pipelines.

10. <u>Resources available to us:</u>

1. https://orm.drizzle.team/

2. https://strapi.io/

3. Build & Deploy Full Stack AI Mock Interview App with Next.js | React, Drizzle ORM, Gemini AI, Clerk (3 hours)

4. https://clerk.com/

5. Documentation: MDN Web Docs, official docs for frameworks/libraries (e.g., React, Node.js).