**PROJECT DESCRIPTION**

**Project Overview: This project aims to develop a dating platform where users can schedule video interviews for evaluation before creating profiles on the platform. The platform will also include admin functionalities for managing user profiles, AI-generated matches, and user conversations. The technology stack used for development will be Next.js for frontend, React for frontend components, and Node.js for backend development. (database mongo)**

**Project Objectives:**

**1. Develop a user-friendly interface for scheduling video interviews. When the user picks the date and time on the calendar the Admin and user should get notifications via email and on the website with a Google Meet link about the upcoming video chat. When the date of videocall is confirmed the system automatically makes that time unavailable for other subscribers.**

**2. Implement backend functionalities for handling user profiles, interviews, and matches. (google meet, calendly api)**

**3. Create an admin dashboard for managing user profiles, approving and denying matches, and conversations, manual matching option. The admin should be able to filter the users with location, gender, quantity of matches, and registration time. The admin should also be able to provide limited access to other admins for having video calls (So we can have many employees), SO they can only approve matches not seeing the personal information of users and other limitations.**

**4. During the interview, the interviewer will make notes and that information will be attached to that user’s account (Every time when we open that user’s account, we can see those notes) and those notes can be seen only by the admin or by those stuff members that Adin provides access.**

**5. Conduct thorough QA testing to ensure the platform's functionality and security.**

**6. Deploy the platform to a production environment (staging server)**

**Functional Requirements:**

**1. User Registration:**

**• Users provide general info to schedule a video interview for evaluation before registration/subscription.**

**• Upon successful evaluation, users receive an email with a profile creation link and payment link option.**

**• Users can fill in necessary information such as name, age, location, interests, and preferences (for AI to match) to create their profiles.**

**• If denied, then they should get an email with a specific text.**

**2. User Dashboard:**

**• After evaluation, users receive notifications regarding their placement on the platform.**

**• Approved users can create profiles and view matches, and get messages from people who received him/her as a match.**

**• SOme people would like to be private, so they should be able to get matches but not be offered to others.**

**• 7 Days before the end of the subscription users should get an automated notification with a renewal link about it so they can renew it.**

**3. Admin Dashboard:**

**• Admins can view and manage user profiles.**

**• Admins can approve or reject AI-generated matches and can also manually make matches between the users. If the user mentiones in his account that he/she wants matches only from specific countries then the AI should consider that.**

**Project Plan:**

**Milestones:**

**1. Design and Frontend:**

**• Design UI/UX for the landing page, interview scheduling section, profile creation, and admin dashboard.**

**2. Backend Development:**

**• Develop backend functionalities for handling user authentication, profile management, interview scheduling, match approval, and conversation management.**

**3. QA Testing:**

**• Conduct comprehensive testing of all functionalities to ensure the platform's reliability, security, and user-friendliness.**

**4. Deployment:**

**• Deploy the platform to a production environment, ensuring scalability and stability.**

**Timeline:**

**• Design Frontend: 2 weeks**

**• Backend Development: 4 weeks**

**• QA Testing: 1 week**

**• Deployment: 1 week**

**Total Project Duration: 8 weeks**

**Note**

**If in the future the website needs to be updated( any part of it, we should be able to do that without losing its functionality or database).**