**Project Design Phase**

**Proposed Solution Template**

| Date | 26-06-2025 |
| --- | --- |
| Team ID |  |
| Project Name | Freelance Finder |
| Maximum Marks | 2 Marks |

**Proposed Solution for Freelance Finder**

| **S. No.** | **Parameter** | **Description** |
| --- | --- | --- |
| **1** | **Problem Statement** (Problem to be solved) | Many freelancers and clients face difficulty in connecting efficiently, managing projects, ensuring trust, and securely handling payments. The lack of real-time interaction, smart matching, and end-to-end transparency leads to dropped projects and dissatisfaction. |
| **2** | **Idea / Solution Description** | The Freelance Finder platform is a full-stack web application (React frontend + Node.js backend) that enables clients to post jobs, freelancers to apply, and both parties to collaborate through real-time messaging and secure payments. Admin controls help manage disputes and community standards. |
| **3** | **Novelty / Uniqueness** | - Intelligent job matching using keyword tagging and filters  - Built-in real-time messaging system  - Milestone-based payments with escrow options  - Admin dispute resolution system  - Optional video call integration |
| **4** | **Social Impact / Customer Satisfaction** | - Empowers freelancers globally to earn from anywhere  - Helps businesses scale using flexible, project-based hiring  - Transparent ratings & reviews enhance platform trust  - Reduces hiring friction and supports gig economy growth |
| **5** | **Business Model (Revenue Model)** | - Freemium access for freelancers  - Commission on each successful payment transaction  - Premium membership for job visibility boost and analytics  - Enterprise client plans for bulk hiring and management |
| **6** | **Scalability of the Solution** | Expandable to global markets with multi-language and multi-currency support  - Mobile-first design with future support via React Native  - Modular architecture for quick integration of features like video calling, AI resume screening, and payment gateways |