**Project Design Phase-II**

**Solution Requirements (Functional & Non-functional)**

|  |  |
| --- | --- |
| Date | JUNE 25 2025 |
| Team ID | LTVIP2025TMID56126 |
| Project Name | FREELANCE FINDER |
| Maximum Marks | 4 Marks |

**Functional Requirements Table**

| **ID** | **Requirement** | **Description** |
| --- | --- | --- |
| FR1.1 | User Registration | Users can register as freelancer, client, or admin |
| FR1.2 | User Authentication | Users can log in securely with email and password |
| FR1.3 | Password Management | Users can reset forgotten passwords |
| FR2.1 | Freelancer Profile Management | Freelancers can create and edit profile with bio, skills, portfolio |
| FR2.2 | Project Browsing | Freelancers can browse and filter client-posted projects |
| FR2.3 | Bid Submission | Freelancers can bid on projects with a proposal and quote |
| FR2.4 | Project Assignment Notification | Freelancers receive notifications when selected for a project |
| FR2.5 | Work Submission | Freelancers can upload and submit completed work |
| FR3.1 | Project Creation | Clients can create and post projects with details |
| FR3.2 | View Bids | Clients can view, compare, and manage received bids |
| FR3.3 | Freelancer Selection | Clients can hire freelancers for their project |
| FR3.4 | Messaging System | Clients can communicate with freelancers via chat |
| FR3.5 | Review and Feedback | Clients can approve submissions and leave ratings/reviews |
| FR4.1 | Admin User Management | Admin can view, block, or delete any user |
| FR4.2 | Project Moderation | Admin can approve or reject project postings |
| FR4.3 | Platform Monitoring | Admin can monitor chats, transactions, and user activity |
| FR4.4 | Reports and Analytics | Admin can generate and view reports on usage and performance |
| FR5.1 | Notification System | All users get real-time updates on relevant actions (e.g., bid received) |
| FR5.2 | Chat Feature | Freelancers and clients can chat within the platform |

**⚙️ Non-Functional Requirements Table**

| **ID** | **Requirement** | **Description** |
| --- | --- | --- |
| NFR1 | Fast Page Load | Pages should load in under 2 seconds |
| NFR2 | Concurrent Users | System should support 500+ users simultaneously |
| NFR3 | Responsive Design | Platform must be mobile and desktop friendly |
| NFR4 | Easy Navigation | User interface must be intuitive and easy to use |
| NFR5 | Password Encryption | Passwords should be hashed using bcrypt |
| NFR6 | Role-Based Access Control | Access to pages/features depends on user roles |
| NFR7 | Secure Admin Access | Admin tools are only accessible to authenticated admin users |
| NFR8 | Secure Data Transmission | All data exchange must use HTTPS |
| NFR9 | High Availability | Platform should maintain 99.5% uptime |
| NFR10 | Crash Recovery | The system should auto-recover from backend failures (using PM2 or similar) |
| NFR11 | Maintainable Codebase | Code should follow modular and clean practices |
| NFR12 | Easy Upgradability | New features can be deployed without disrupting the platform |
| NFR13 | Scalable Architecture | System should be scalable horizontally to handle increased load |
| NFR14 | Optimized Database Queries | MongoDB should support efficient queries as data volume grows |