**Project Design Phase-II**

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

|  |  |
| --- | --- |
| Date | 31 January 2025 |
| Team ID | LTVIP2025TMID32102 |
| Project Name | SmartSDLC – AI-Enhanced Software Development Lifecycle |
| Maximum Marks | 4 Marks |

**Functional Requirements:**

Following are the functional requirements of the proposed solution.

| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
| --- | --- | --- |
| FR-1 | User Registration | - Registration through Form - Registration through Gmail - Registration through LinkedIn |
| FR-2 | User Confirmation | - Confirmation via Email - Confirmation via OTP |
| FR-3 | User Login | - Login via Email & Password - Login via Gmail - Login via LinkedIn |
| FR-4 | User Dashboard | - View personal profile and activity - View project/task progress - Access AI assistant features - View recent notifications |

**Non-functional Requirements:**

Following are the non-functional requirements of the proposed solution

| FR No. | Non-Functional Requirement | Description |
| --- | --- | --- |
| NFR-1 | Usability | The system should have an intuitive UI/UX with minimal learning curve for developers, managers, and other users. Streamlit interface should support clear workflows and guided interactions. |
| NFR-2 | Security | User data, access tokens, and project artifacts must be securely stored and encrypted. The system should follow OAuth 2.0 for third-party integrations and ensure secure authentication (e.g., Gmail, LinkedIn). |
| NFR-3 | Reliability | The platform must operate without failures during code generation, AI analysis, and document creation. Backup routines and auto-save functionalities should be included. |
| NFR-4 | Performance | The system must generate code, test cases, and documentation in under 5 seconds per request under typical load. LLM response latency should be optimized. |
| NFR-5 | Availability | The platform should be accessible 24/7 with at least 99.5% uptime, especially during deployments or sprint planning. Downtime should be minimized during updates. |
| NFR-6 | Scalability | The solution must support scaling across multiple users and projects concurrently. Backend services (e.g., model inference) should handle increasing requests via load balancing or microservices. |