Project Design Phase-II
Solution Requirements (Functional & Non-functional)

Date	27 June 2025
Team ID	LTVIP2025TMID38248
Project Name	SmartSDLC - AI-Enhanced Software Development Lifecycle
Maximum Marks	4 Marks

Functional Requirements:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Requirement Classification	PDF Upload Functionality SDLC Phase Classification Text Extraction from PDFs
FR-2	AI Code Generation	Natural Language Prompt Input Python Code Output Display
FR-3	Automated Bug Fixing	Buggy Code Snippet Input Fixed Code Output Display
FR-4	Smart Test Case Generation	Code/Requirement Input for Tests Test Case Generation
FR-5	Code Summarization	Code Snippet Input for Summary Human-Readable Code Summary Display
FR-6	Interactive AI Chatbot Assistant	User Message Input AI Response Display

Non-functional Requirements:

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The platform shall be intuitive and easy to navigate for all user types (developers, PMs, QAs). Input fields and buttons should be clearly labeled and provide immediate feedback.
NFR-2	**Security**	All data transmission between frontend and backend shall use HTTPS. User input for AI models should be handled securely, and API keys shall be protected. (Consider authentication/authorization for future versions).

NFR-3	**Reliability**	The backend API shall maintain high uptime and gracefully handle errors from AI model inference or PDF processing. It should recover quickly from transient failures.
NFR-4	**Performance**	AI model responses (code generation, bug fixing, summarization, chatbot) shall be returned within an acceptable timeframe (e.g., typically under 15-20 seconds). PDF processing should be efficient for documents up to 10-15 pages.
NFR-5	**Availability**	The system should be accessible to users whenever needed. If deployed in a cloud environment, it should leverage cloud provider's availability zones for resilience.
NFR-6	**Scalability**	The backend infrastructure should be able to scale horizontally to accommodate an increasing number of concurrent users and AI requests without significant performance degradation. The AI model's serving infrastructure (Hugging Face) should support scaling.