**Project Design Phase-II**

**Data Flow Diagram & User Stories**

|  |  |
| --- | --- |
| Date | 31 January 2025 |
| Team ID | LTVIP2025TMID32689 |
| Project Name | Smart SDLC -AI-enhanched software development life cycle |
| Maximum Marks | 4 Marks |

# User Stories

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance Criteria | Priority | Release |
| Requirement Analysis | USN-1 | As a developer, I can convert unstructured requirements into structured user stories using NLP. | I can see a list of user stories generated from the unstructured input. | High | Sprint-1 |
| Code Generation | USN-2 | As a developer, I can generate code automatically from user stories. | The generated code matches the requirements specified in the user stories. | High | Sprint-1 |
| Bug Fixing | USN-3 | As a developer, I can auto-detect and fix bugs in the generated code. | Bugs are detected and automatically fixed without manual intervention. | High | Sprint-1 |
| Test Case Generation | USN-4 | As a developer, I can automatically generate test cases based on user stories. | Test cases are generated and can be executed without errors. | Medium | Sprint-1 |
| Code Summary | USN-5 | As a developer, I can generate a summary of the code to improve understanding and collaboration. | A summary of the code is generated with key sections and functionalities outlined. | Medium | Sprint-2 |
| Chatbot Support | USN-6 | As a developer, I can ask the chatbot for assistance on any SDLC-related task and receive real-time help. | The chatbot provides relevant, accurate answers to queries related to SDLC tasks. | High | Sprint-2 |
| Platform Deployment | USN-7 | As a developer, I can deploy SmartSDLC to the cloud for use by the team. | SmartSDLC is successfully deployed on the cloud, and all team members can access it. | High | Sprint-2 |

# Data Flow Diagram (DFD) Description

1. Level 0 (Context Diagram):  
- External Entities:   
 - Users (developers, testers, project managers)  
 - External Systems (e.g., GitHub, Jira, cloud platforms)  
- Process:   
 - SmartSDLC System: Handles requirement analysis, code generation, bug fixing, test case creation, and code summarization.  
- Data Stores:   
 - User Stories Database  
 - Generated Code Repository  
 - Test Case Repository  
- Data Flow:   
 - Users input unstructured requirements.  
 - SmartSDLC processes the input, generates user stories, and produces corresponding code.  
 - The system automatically generates test cases and stores them in the repository.  
 - Chatbot responds to user queries and provides information about the SDLC process.