**PROJECT REPORT**

**Title:** **Smart SDLC: Intelligent Software Development Life Cycle Assistant**

**1. INTRODUCTION**

**1.1 Project Overview**

Smart SDLC is an intelligent assistant powered by IBM Watson that guides users through every phase of the software development life cycle. It provides structured data entry and generates documentation, testing outputs, and phase-wise reports—without requiring users to write code.

**1.2 Purpose**

To simplify and automate the SDLC process by providing a user-friendly, no-code interface that helps students and developers efficiently plan, document, and simulate SDLC activities.

**2. IDEATION PHASE**

**2.1 Problem Statement**

SDLC documentation and phase testing are often tedious and prone to inconsistency, especially for students and beginner developers. There is a need for a simplified, guided tool that can walk users through each stage of the cycle interactively.

**2.2 Empathy Map Canvas**

* **Says:** "I want help documenting and testing each SDLC phase."
* **Thinks:** "Is this format sufficient for my project?"
* **Does:** Searches for templates, fills out documents manually
* **Feels:** Overwhelmed by structure, unsure how to proceed
* **Pains:** Manual effort, lack of structure
* **Gains:** Streamlined, consistent documentation and output generation

**2.3 Brainstorming**

Ideas included:

* Auto-generated documentation
* Form-based phase inputs
* Integrated visual output for design & testing
* One-click PDF export  
  Streamlit was chosen for its simplicity and visual capabilities.

**3. REQUIREMENT ANALYSIS**

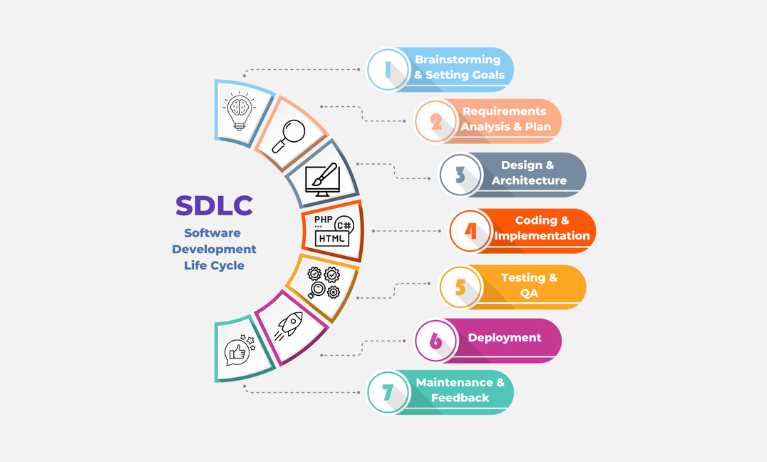
**3.1 Customer Journey Map**

1. Launch application
2. Choose SDLC phase (e.g., Requirements, Design)
3. Input relevant data
4. View auto-generated output
5. Export the phase report as PDF

**3.2 Session Requirements**

* Dynamic inputs for all SDLC phases
* Editable phase summaries
* Functional test input/output handling
* Real-time preview and export option

**3.3 Data Flow Diagram**



**3.4 Technology Stack**

* **Frontend:** Streamlit
* **Backend:** Python
* **AI Service:**IBM Watson M
* **Deployment:** Localhost / Streamlit Cloud

**4. PROJECT DESIGN**

**4.1 Problem-Solution Fit**

The application addresses the difficulty of manually documenting SDLC by guiding the user step-by-step and providing formatted, exportable output.

**4.2 Proposed Solution**

A form-based SDLC assistant that lets users input and view results for each phase (Requirement, Design, Testing, etc.) and exports a PDF with all collected outputs.

**4.3 Solution Architecture**

* **UI Layer:** Streamlit widgets (forms, dropdowns, buttons)
* **Application Logic:** Python modules for each SDLC phase
* **Output Layer:** Rendered HTML/PDF
* **Data Storage:** Session-based memory (temporary)

**5. PROJECT PLANNING & SCHEDULING**

**5.1 Project Planning**

|  |  |  |
| --- | --- | --- |
| **Week** | **Dates** | **Activites** |
| Week 1 | June 1 – June 7 | Ideation, Phase planning, UI layout design |
| Week 2 | June 8 – June 14 | Backend phase logic, UI coding, form setup |
| Week 3 | June 15 – June 21 | Testing, debugging, final PDF integration |
| Week 4 | June 22 – June 26 | |  | | --- | | Report generation, documentation, submission | |

**6. FUNCTIONAL AND PERFORMANCE TESTING**

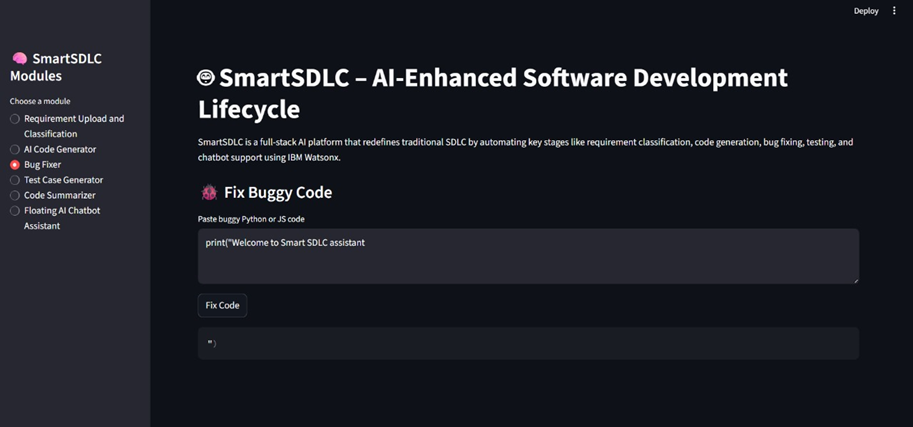
**6.1 Performance Testing**

* **UI Speed:** Instantaneous response to phase selections
* **Load Handling:** Able to process long inputs without lag
* **PDF Accuracy:** Consistent formatting across all phase outputs
* **Cross-Browser:** Tested on Chrome, Firefox, Edge

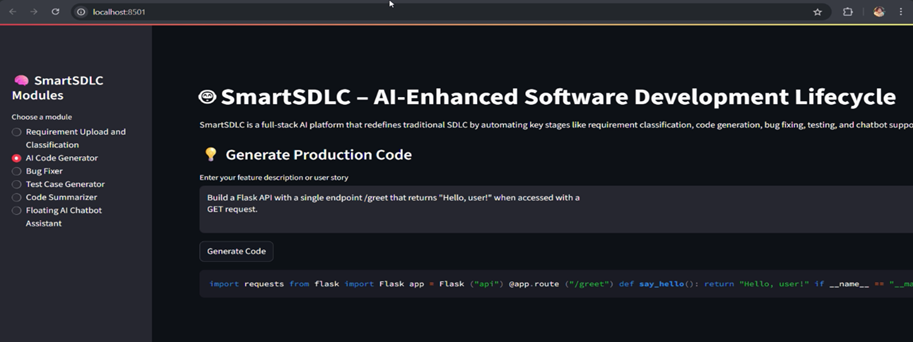
**7. RESULTS**

**7.1 Output Screenshots**

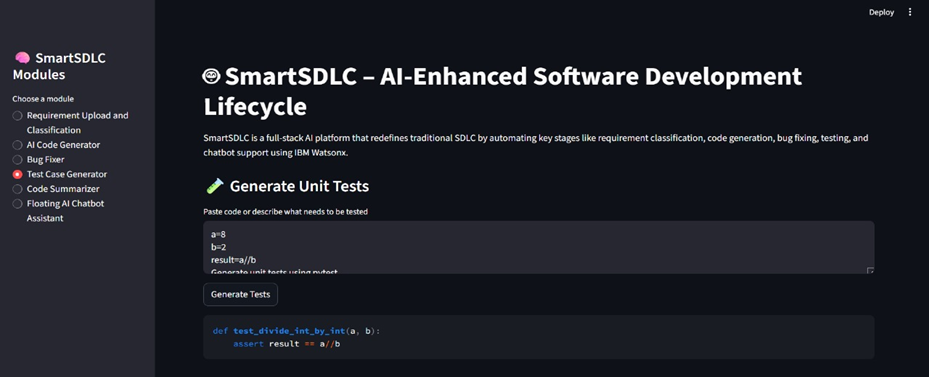
* Screenshot: Fix Debug Code



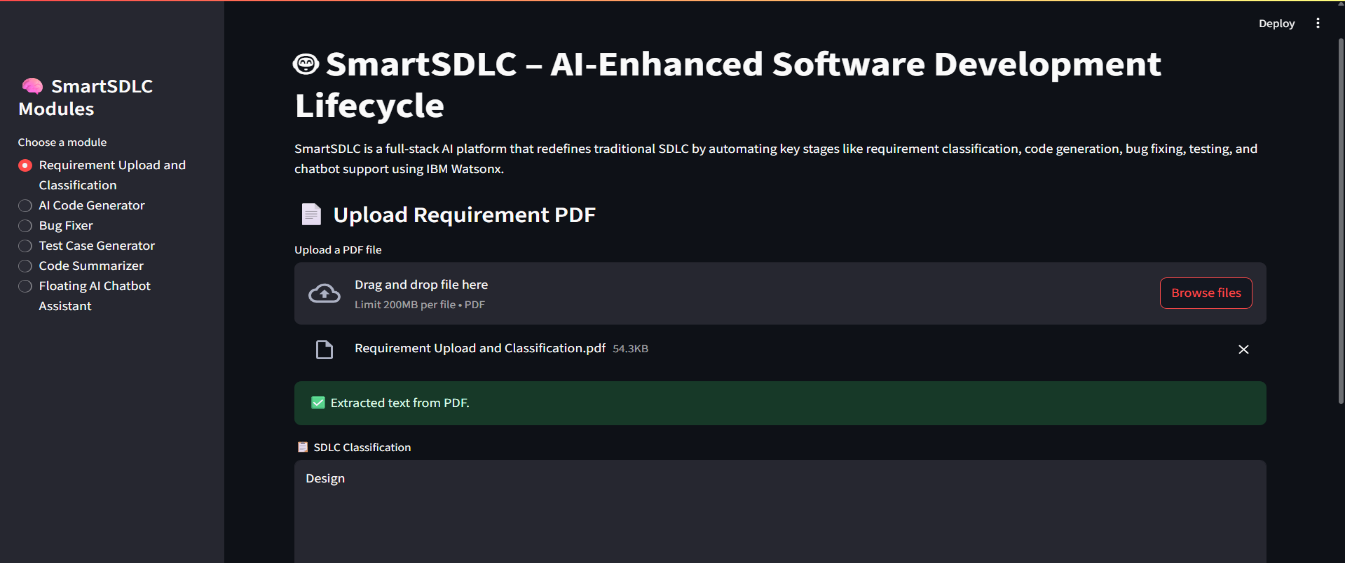
* Screenshot: Generate Production Code



* Screenshot: Testing Phase I/O



* Screenshot: Exported PDF Output



**8. ADVANTAGES & DISADVANTAGES**

**Advantages**

* Simple, interactive UI
* Phase-wise input and output
* No need for backend logic or programming
* Quick PDF export

**Disadvantages**

* No user authentication or saved sessions
* Limited customization
* Doesn’t support full backend execution or live database

**9. CONCLUSION**

Smart SDLC with Streamlit provides an innovative approach to software development planning by converting a complex manual process into a user-friendly digital assistant. It serves as an ideal platform for academic projects, documentation, and prototyping.

**10. FUTURE SCOPE**

* Add AI-assisted recommendations per phase
* Enable GitHub/Jira integration for real-time updates
* Add user login and data persistence
* Enhance visualization with real-time diagrams and charts

**11. APPENDIX**

* **GitHub Link:** https://github.com/prasanna-maradana/Smart-SDLC-Intelligent-Software-Development-Life-Cycle-Assistant
* **Demo Link:** *https://github.com/prasanna-maradana/Smart-SDLC-Intelligent-Software-Development-Life-Cycle-Assistant/tree/main/DemoVideo*
* **Source Files:** main.py, phase\_handler.py, pdf\_generator.py