**Project Report Format**

1. **INTRODUCTION** 
   1. Project Overview

**Project Title:**  
**SmartSDLC – AI-enhanced Software Development Lifecycle Assistant**

**Description:**  
SmartSDLC is an AI-powered platform designed to assist software developers and teams throughout the Software Development Lifecycle. It provides intelligent tools such as requirement classification, code generation, bug fixing, test case creation, code summarization, and PDF document processing. The system integrates IBM WatsonX AI for natural language understanding and code analysis, offering real-time support in streamlining development tasks.

**Objective:**  
To automate and enhance various phases of the SDLC using generative AI, reducing manual effort, improving accuracy, and accelerating software delivery.

**Core Features:**

* Requirement Classification
* AI-powered Code Generation
* Bug Detection & Fixing
* Test Case Generation
* Code Summarization
* PDF Text Extraction
* Chat-based Development Assistant

**Technologies Used:**

* **Frontend:** Streamlit (Python)
* **Backend:** FastAPI
* **AI Model:** IBM WatsonX Granite
* **Authentication:** Custom basic login
* **Deployment:** Local (can be extended to cloud platforms)
* **Version Control:** Git + GitHub

**Target Users:**

* Software Developers
* Computer Science Students
* Project Teams looking to automate documentation and code writing tasks

**Impact:**  
This platform simplifies developer workflows, especially during initial development phases. It boosts productivity and reduces dependency on multiple tools by offering an all-in-one AI assistant tailored to SDLC needs.

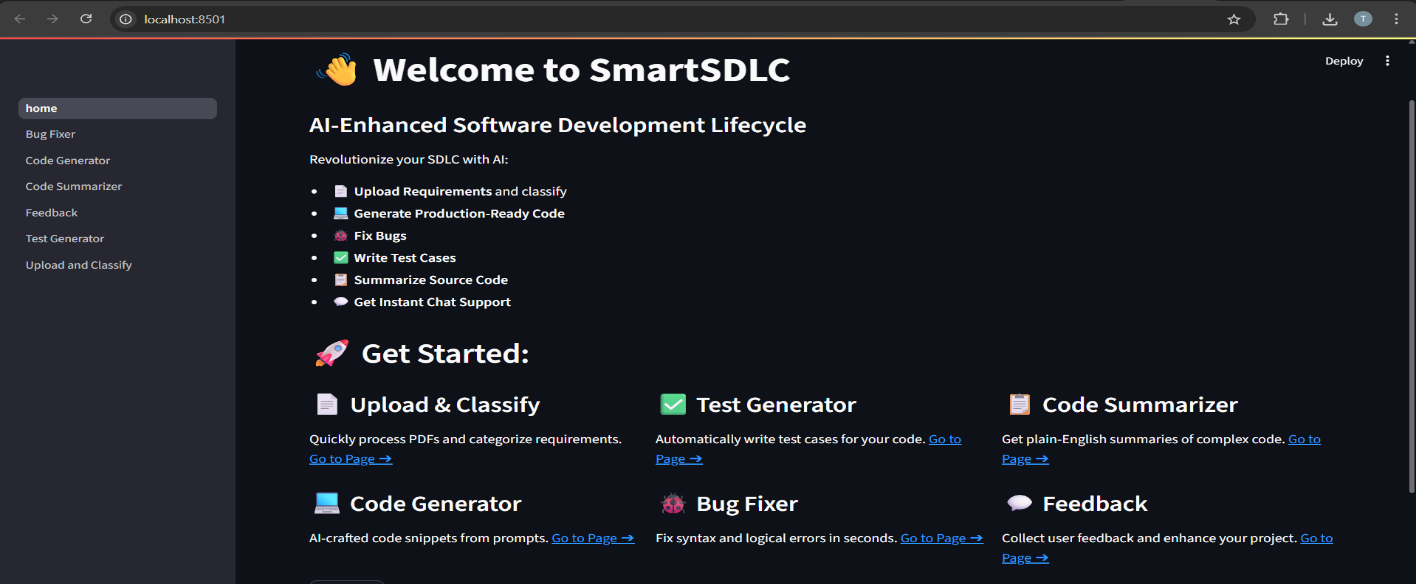
* 1. Purpose

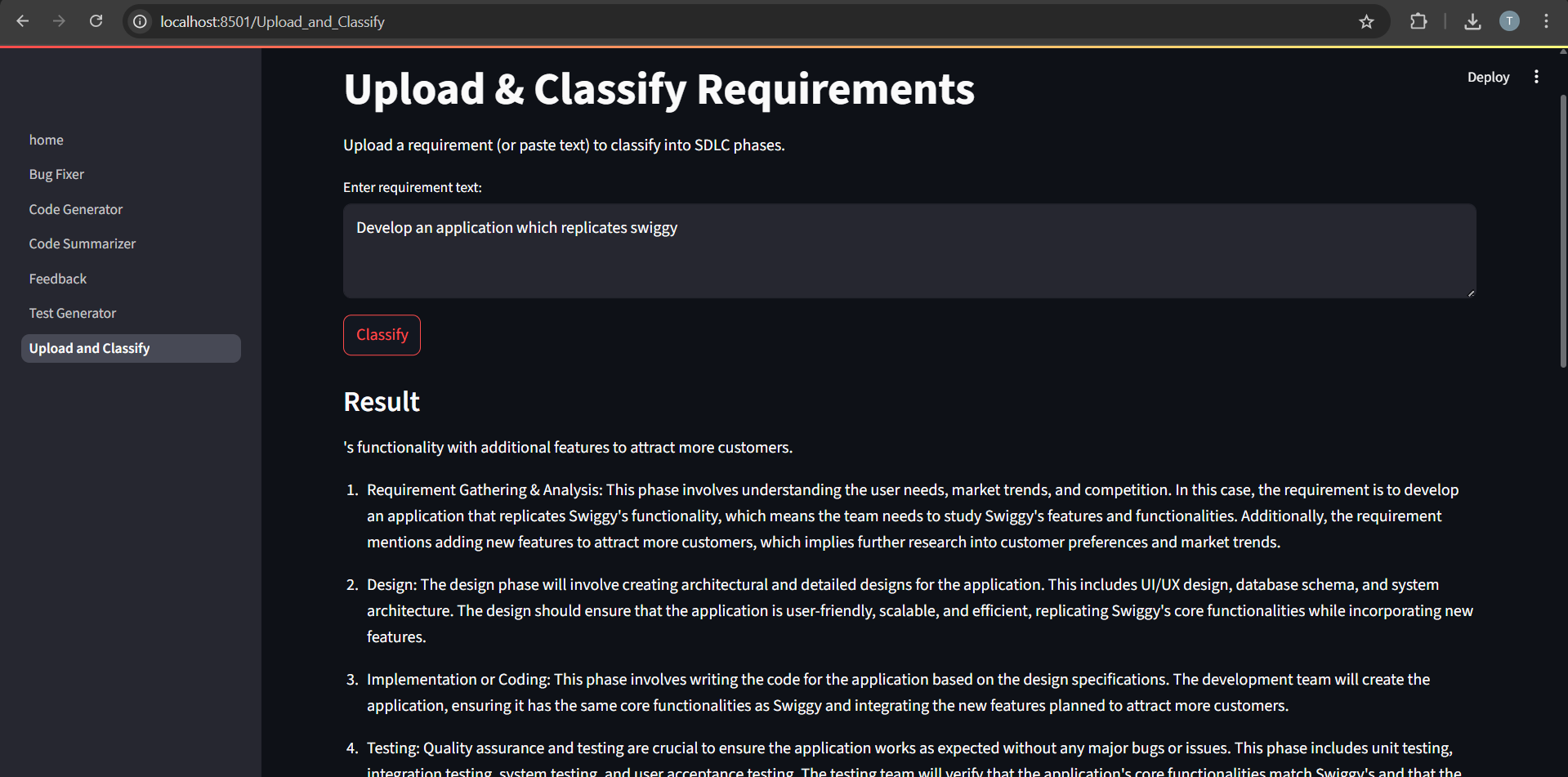
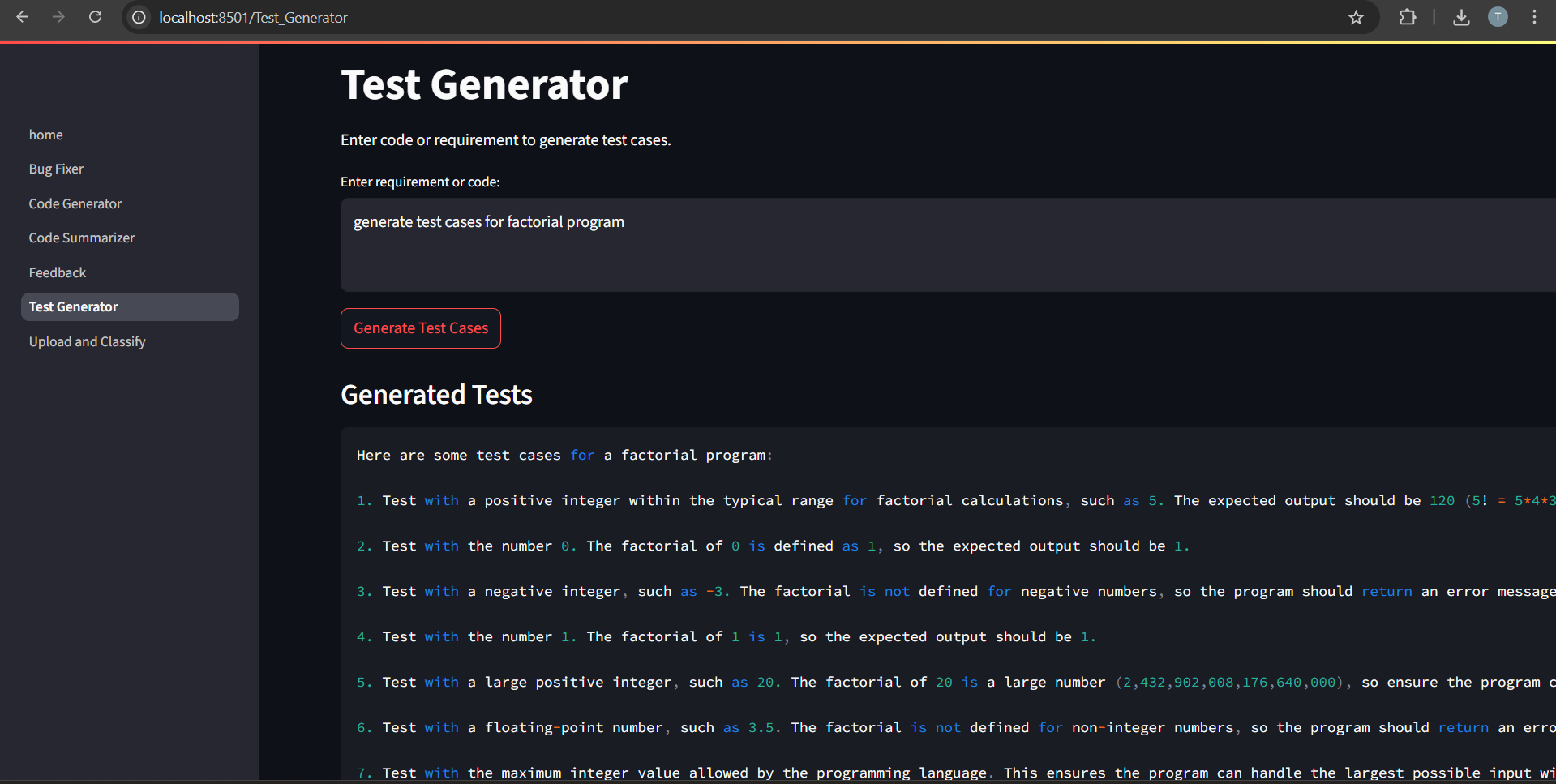
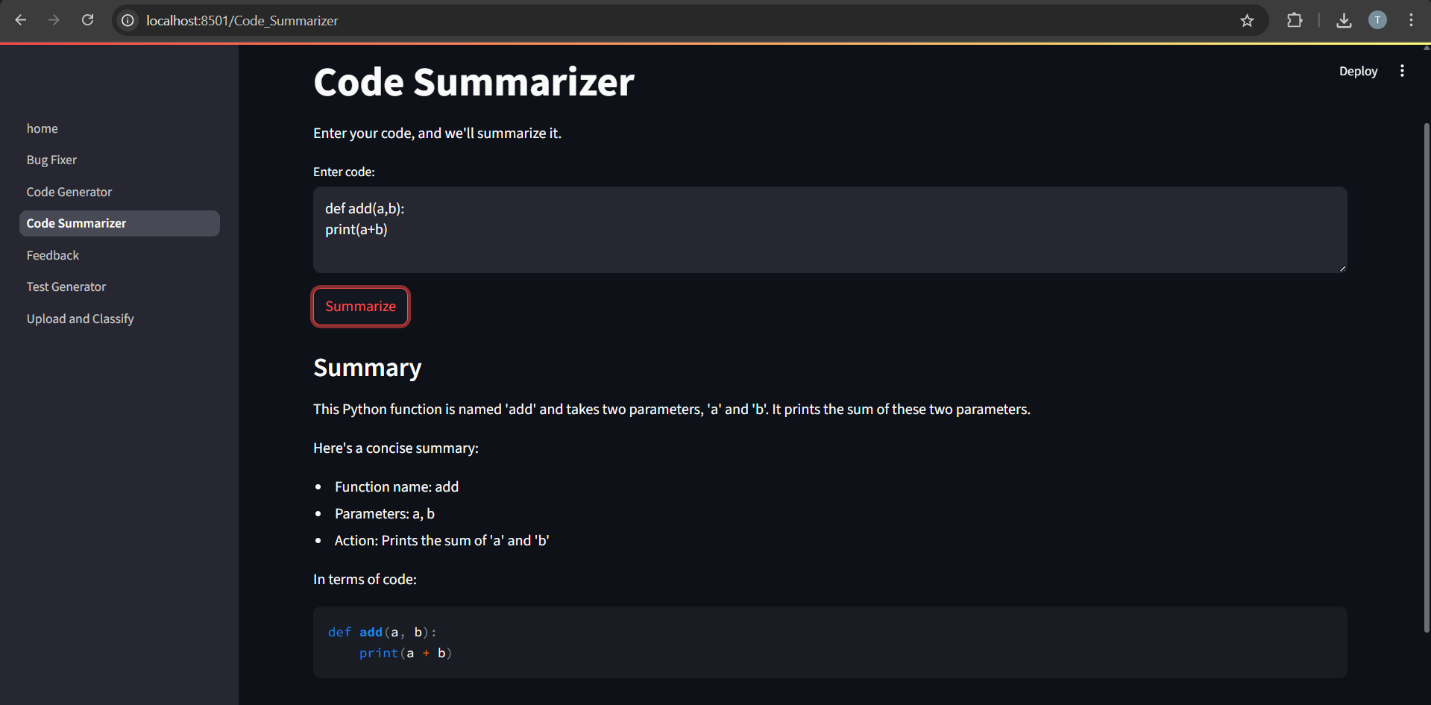
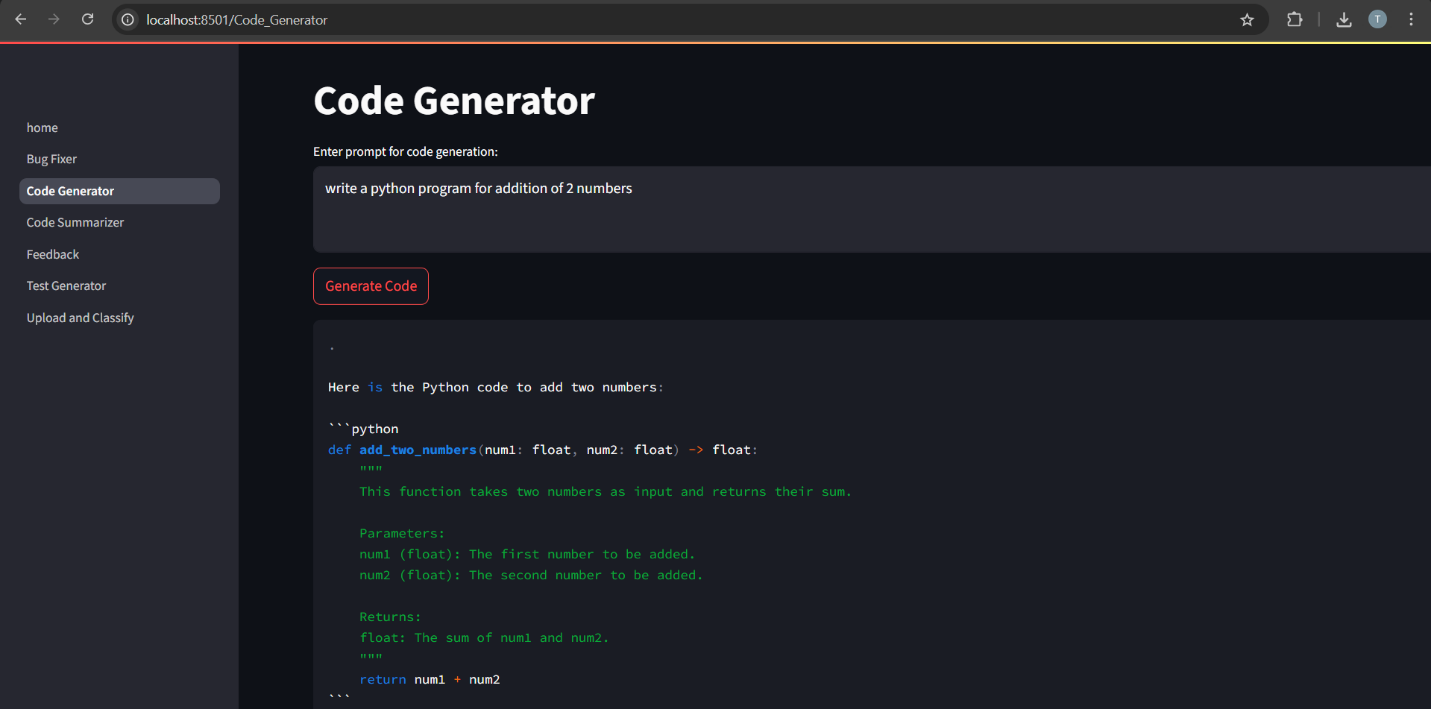
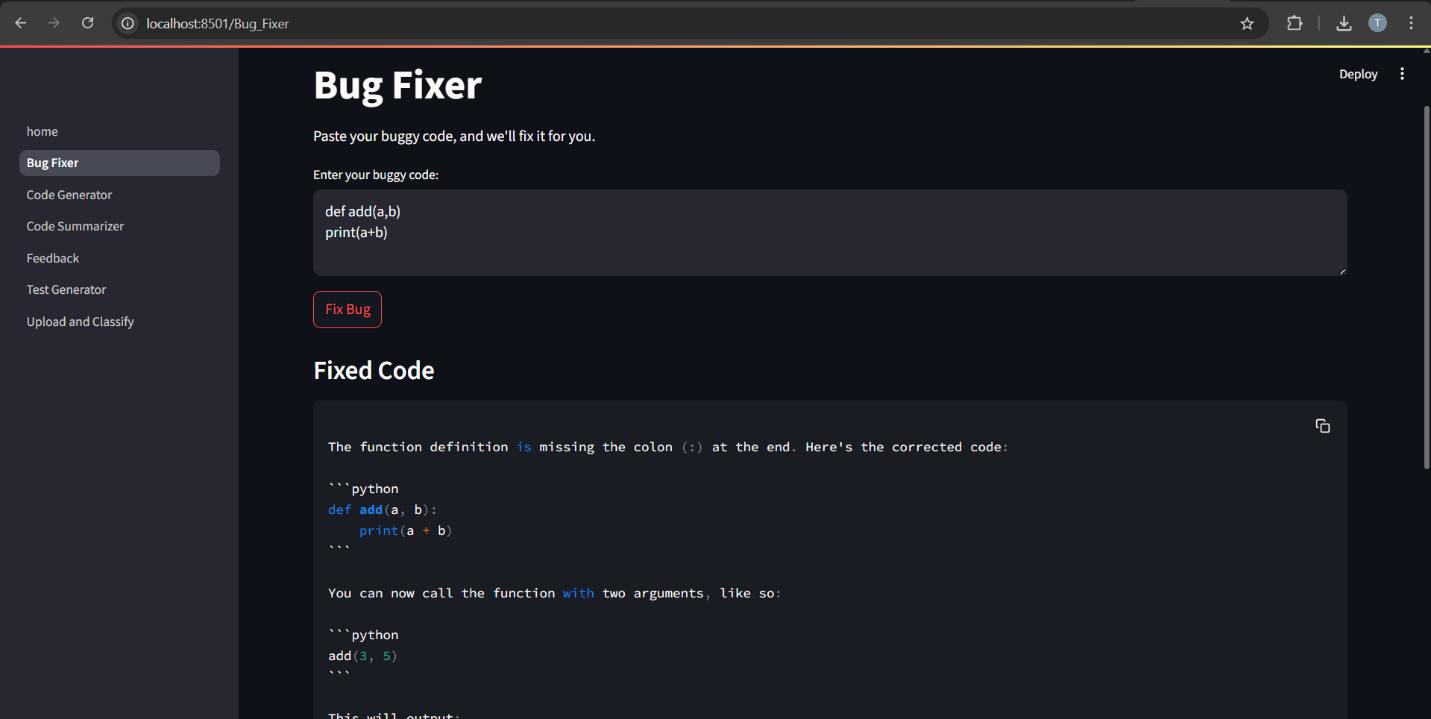
The purpose of *SmartSDLC* is to streamline and accelerate the Software Development Lifecycle by integrating AI-driven tools that assist with requirement analysis, code generation, bug fixing, testing, and documentation. This project aims to:

* Minimize manual effort and repetitive tasks in software development
* Enhance development accuracy and consistency using AI suggestions
* Provide developers with intelligent assistance for faster decision-making
* Simplify documentation and communication throughout SDLC phases
* Make modern AI capabilities accessible to students, developers, and teams

By automating routine development tasks, SmartSDLC boosts productivity, shortens development time, and fosters efficient collaboration in software projects.

1. **IDEATION PHASE**
   1. Problem Statement
   2. Empathy Map Canvas
   3. Brainstorming
2. **REQUIREMENT ANALYSIS**
   1. Customer Journey map
   2. Solution Requirement
   3. Data Flow Diagram
   4. Technology Stack
3. **PROJECT DESIGN** 
   1. Problem Solution Fit
   2. Proposed Solution
   3. Solution Architecture
4. **PROJECT PLANNING & SCHEDULING** 
   1. Project Planning
5. **FUNCTIONAL AND PERFORMANCE TESTING** 
   1. Performance Testing
6. **RESULTS** 
   1. Output Screenshots





1. **ADVANTAGES & DISADVANTAGES**

**Advantages**

1. **Automation of SDLC Tasks**  
   Reduces manual work in requirement classification, code writing, and test generation.
2. **Increased Productivity**  
   Accelerates development by generating code and fixing bugs with minimal input.
3. **Consistency & Accuracy**  
   Reduces human errors and ensures standard-compliant outputs using AI logic.
4. **Natural Language Interface**  
   Users can interact with the system using plain language, no deep technical input required.
5. **All-in-One Platform**  
   Integrates multiple development tools in a single unified interface.
6. **Real-time Chat Support**  
   A floating chatbot offers instant SDLC-related assistance to users.

**Disadvantages**

1. **AI Dependency**  
   Over-reliance on AI-generated outputs might lead to reduced developer learning or oversight.
2. **Limited Customization**  
   AI outputs may not always meet highly specific or complex development requirements.
3. **No Database Integration (Current Scope)**  
   Project is limited to local state; no persistent data storage (as per internship scope).
4. **Model Accuracy Constraints**  
   Output quality depends heavily on prompt quality and AI model capabilities.
5. **Requires Internet & API Access**  
   Needs active IBM Watsonx credentials and network for real-time AI interaction.
6. **CONCLUSION**

The SmartSDLC platform successfully demonstrates how AI can revolutionize the traditional Software Development Lifecycle by automating critical stages such as requirement analysis, code generation, bug fixing, testing, and documentation. By integrating IBM Watsonx and leveraging a modular architecture with FastAPI and Streamlit, this solution empowers developers with intelligent assistance, improving both speed and accuracy in software delivery.

Through this project, we explored how generative AI models can be applied practically in real-world development scenarios. While the tool is designed without persistent databases to suit internship constraints, it delivers fully functional core features and can be easily extended for production use.

SmartSDLC stands as a scalable and user-friendly innovation that enhances developer efficiency and provides a modern approach to managing software lifecycles intelligently.

1. **FUTURE SCOPE 11. APPENDIX**

Source Code(if any)

Dataset Link

GitHub & Project Demo Link : https://github.com/TYogesh12/MyGenproject