SmartSDLC : AI-Enhanced Software Development Lifecycle

**Team Members** :

SHAIK AKHILA

SHAIK SHAKEELA

SHAIK ANSHU

SHAIK RIZVANA

SHAIK SHARMILA

Faculty Mentor: Manoj

# 1. Introduction

**Smart SDLC** (Software Development Life Cycle) refers to a modern, efficient approach to software development that emphasizes **agility, automation, collaboration, and continuous improvement**. It integrates smart tools and practices to streamline each phase—planning, development, testing, deployment, and maintenance—for faster, higher-quality software delivery.

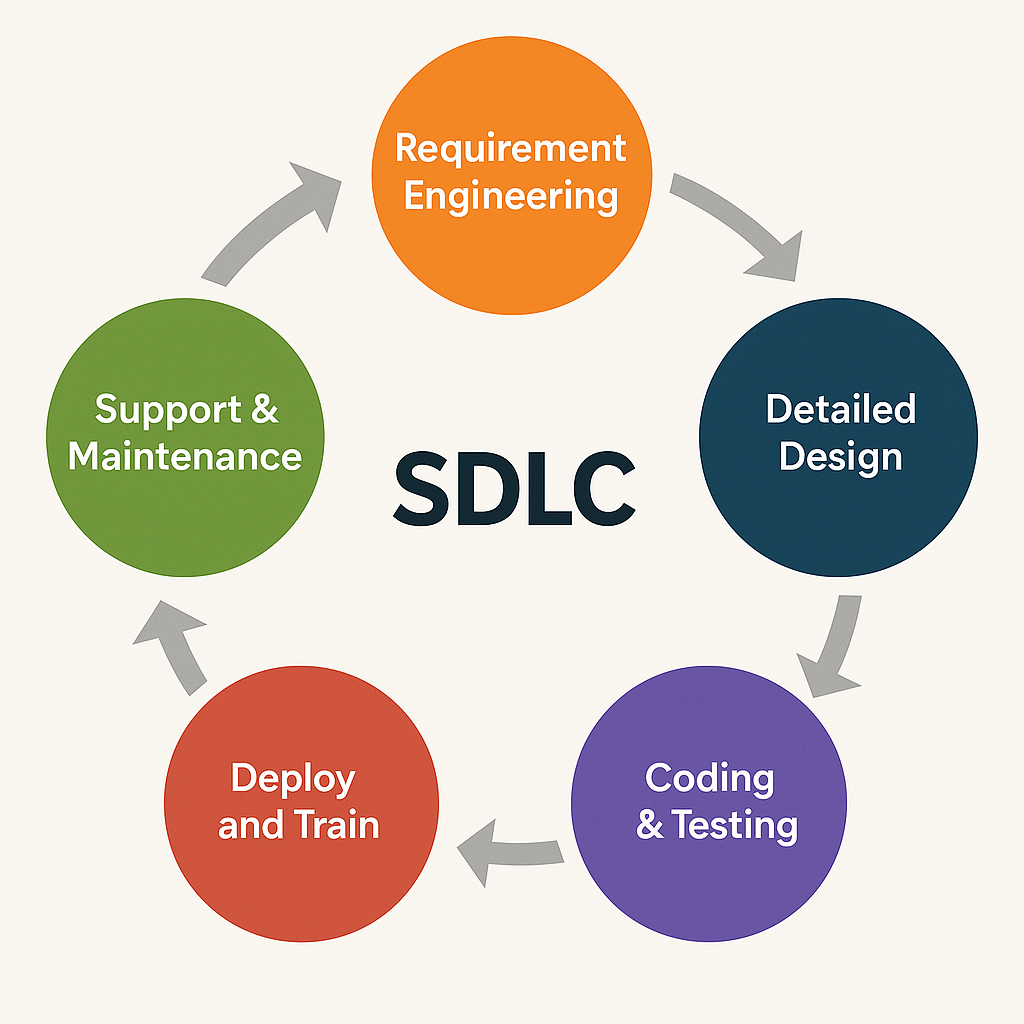
**2. Tools & Technologies**

- IBM Granite (LLM API from IBM watsonx.ai)  
- Python 3, Flask  
- HTML, CSS, JavaScript (AJAX)  
- VS Code  
- Replit / Render for deployment

**3. Technical Architecture**

The system consists of a front end interface (HTML/CSS/JS) where the user types a query. The back end is a Flask server that sends the user query to the IBM Granite model via Open AI- compatible API, fetches the response, and

returns it to the user in real time.



# 4. Modules Implemented

- Model Selection and Architecture: IBM Granite 3-3b Chat model is used.  
- Core Functionality: Text-based question-answering system.  
- Application Logic: Flask handles the API call and response formatting.  
- UI Design: Simple chat interface for end users.  
- Deployment: Easily deployable on platforms like Render or Replit.

# 5. Sample Code Snippet

Backend logic to query IBM Granite:

response = openai.ChatCompletion.create(  
 model="ibm/granite-3-3B granite instruct”,  
 messages=[  
 {"role": "system", "content": "You are a helpful chat bot assistant."}

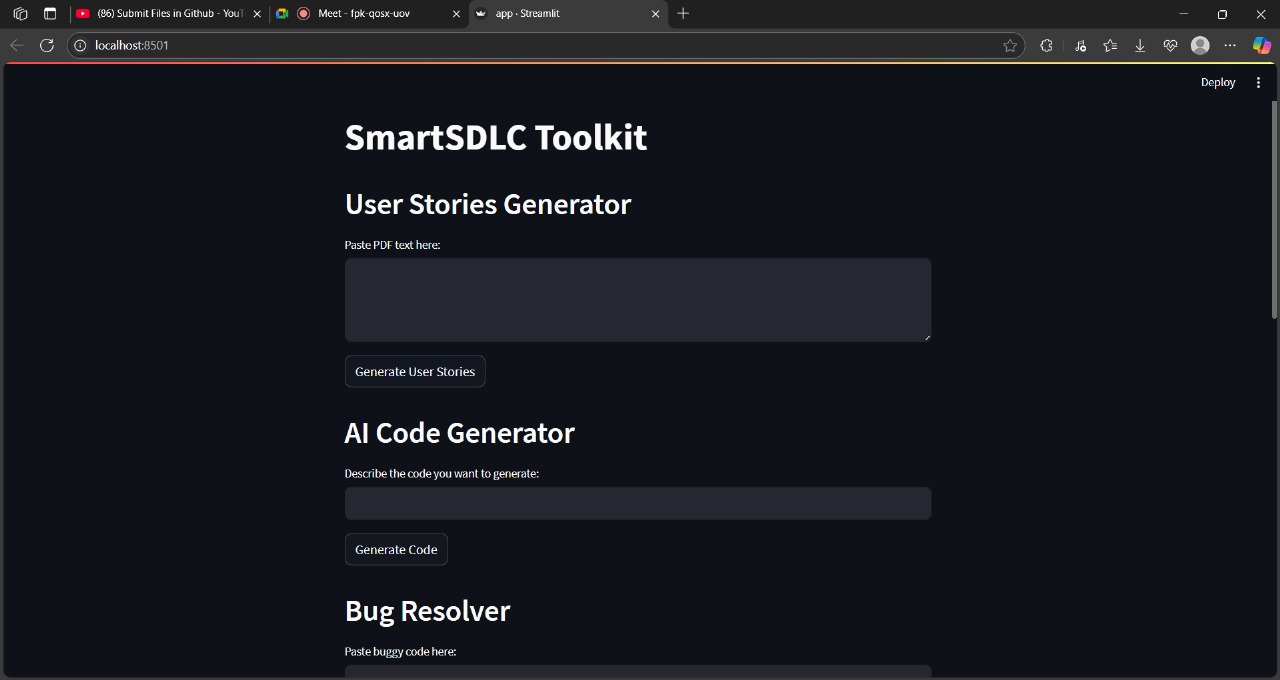
{"role": "user", "content": user\_input}

]

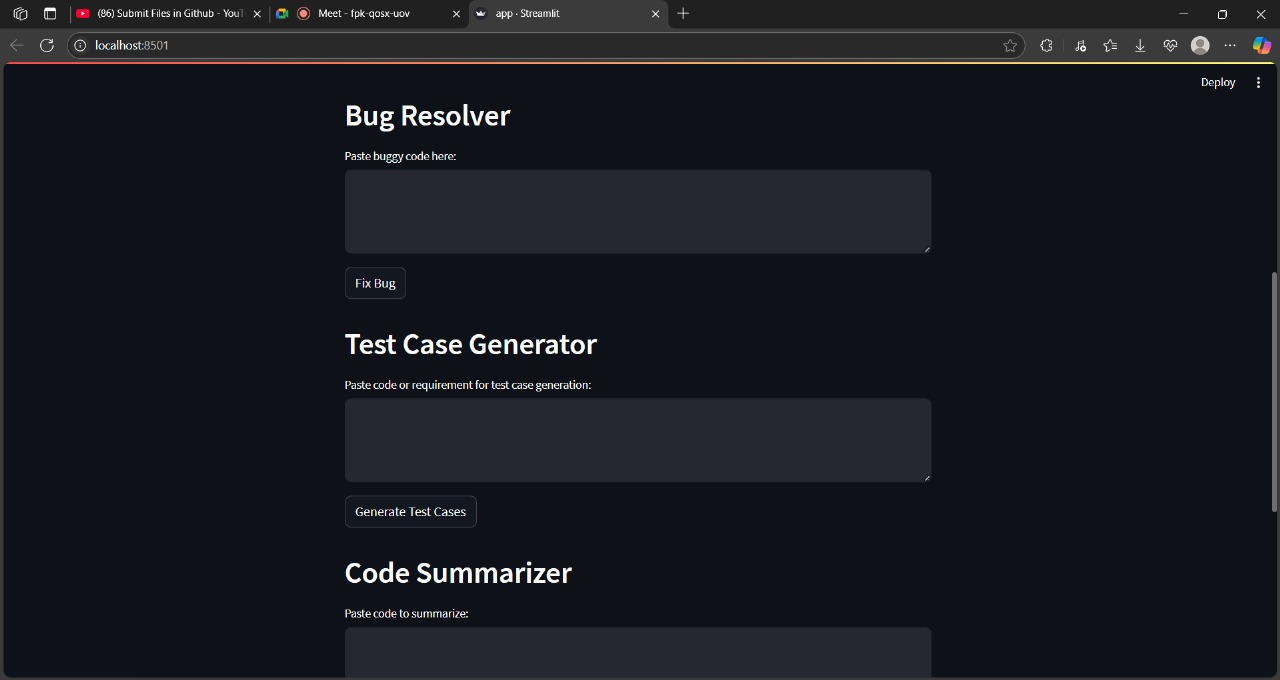
}

**6. Output (Demo)**

Starting interface of the Website:



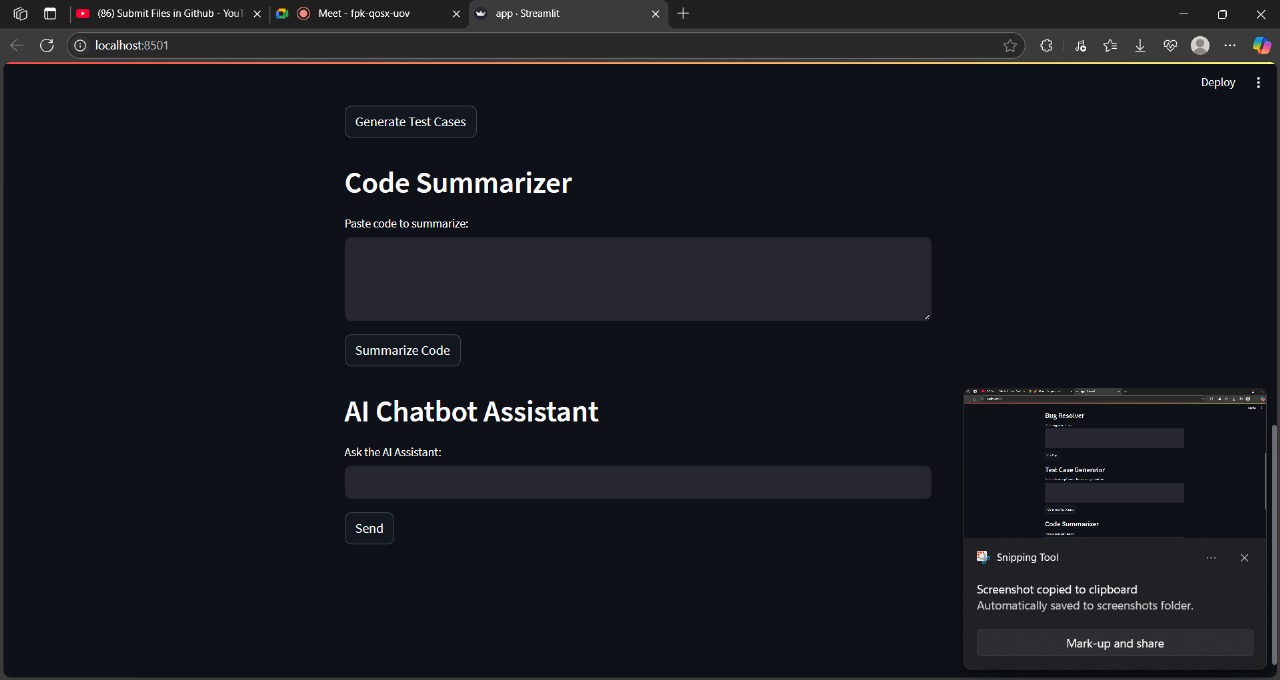
Output of the AI Chatbot Assistance:



Output for Pdf Summarizer:

# 

Output of the Multilingual Code Generator:



# Advantages & Disadvantages

**Advantages :**

**-** Automation

**-** Faster Delivery

**-** Improved Collaboration

**-** Higher Quality Software

**-** Scalabilty

**Disadvantages:**

**-** High Initial Cost

**-** Tool Complexity

**-** Over dependence on Tools

**-** Security Risks

**-** Requires Skilled Workforce

# 8. Conclusion

The Smart SDLC platform represents a significant advancement in the automation of the Software Development Life cycle by integrating AI - powered intelligence into each phase - from requirement analysis to code generation, testing, bug, fixing, and documentation.

Overall, Smart SDLC not only improves productivity and accuracy but also foundation for future enhancements like CI/CD integration, team collaboration, version control, and cloud deployment

**9.Future Scope**

* Add multilingual support
* Connect with telemedicine services
* Integrate voice input and wearable data

# Apendix

* Source Code: Included
* Dataset Link: IBM cloud Model Documentation
* GitHub Link: https://github.com/Aakhilashaik/SmartSDLC-AI-Enhanced-Software-Development-Lifecycle