# SmartSDLC – AI-Enhanced Software Development Lifecycle

**Abstract**—The SmartSDLC project leverages IBM Granite models with Generative AI to accelerate the software development lifecycle. It simplifies requirement gathering, code generation, testing, debugging, documentation, and AI-assisted collaboration. The solution is implemented using Google Colab for ease of deployment and efficiency.

## I. INTRODUCTION

The software development lifecycle (SDLC) has evolved significantly with the introduction of artificial intelligence. SmartSDLC aims to enhance development by utilizing IBM Granite models, enabling automatic requirement generation, coding assistance, test creation, debugging, and documentation. The project is deployed in Google Colab, offering a reliable environment with GPU support for faster execution.

## II. PRE-REQUISITES

To implement SmartSDLC effectively, the following are required:
1. Gradio framework knowledge.
2. Familiarity with IBM Granite models from Hugging Face.
3. Proficiency in Python programming.
4. Version control with Git.
5. Understanding of Google Colab's T4 GPU environment.

## III. PROJECT WORKFLOW

A. Activity-1: Exploring Naan Mudhalvan Smart Interz Portal
Users begin by accessing the Smart Interz portal, logging into their accounts, and enrolling in the project "SmartSDLC – AI-Enhanced Software Development Lifecycle."

B. Activity-2: Choosing an IBM Granite Model
From Hugging Face, users select the IBM Granite model. In this project, the granite-3.2-2b-instruct model is chosen for its lightweight and fast performance.

C. Activity-3: Running the Application in Google Colab
The project runs in Google Colab by installing required libraries such as Transformers, Torch, Gradio, and PyPDF2. A T4 GPU is selected to ensure optimal execution. The Gradio application is then launched for user interaction.

D. Activity-4: Uploading the Project to GitHub
The final step involves uploading the project to GitHub for version control and collaboration. Users create a repository, upload their files, and commit changes to maintain the project

online.

## IV. RESULTS

The integration of IBM Granite with Gradio in Google Colab demonstrates how AI can enhance software development workflows. The deployed application provides an interactive interface for requirement analysis, code generation, and execution, proving the efficiency of AI-assisted SDLC.

## V. CONCLUSION

SmartSDLC demonstrates the potential of Generative AI in revolutionizing the traditional SDLC. With IBM Granite models, developers can accelerate tasks such as coding, debugging, and documentation. The project's deployment in Google Colab makes it accessible and efficient, while GitHub integration ensures proper version control and collaboration.

## REFERENCES

[1] Gradio Documentation. [Online]. Available: https://www.gradio.app/guides/
[2] IBM Granite Models. [Online]. Available: https://huggingface.co/ibm-granite
[3] Python Documentation. [Online]. Available: https://docs.python.org/3/
[4] Git Documentation. [Online]. Available: https://git-scm.com/docs/git
[5] Google Colab GPU Usage. [Online]. Available: https://www.geeksforgeeks.org/python/how-to-use-gpu-in-google-colab/
[6] Naan Mudhalvan Smart Interz Portal. [Online]. Available: https://naanmudhalvan.smartinternz.com/
[7] GitHub. [Online]. Available: https://github.com/