Index

Contents

[Introduction 3](#_Toc208773734)

[Objectives 3](#_Toc208773735)

[Pre-requisites: 3](#_Toc208773736)

[Project Workflow: 4](#_Toc208773737)

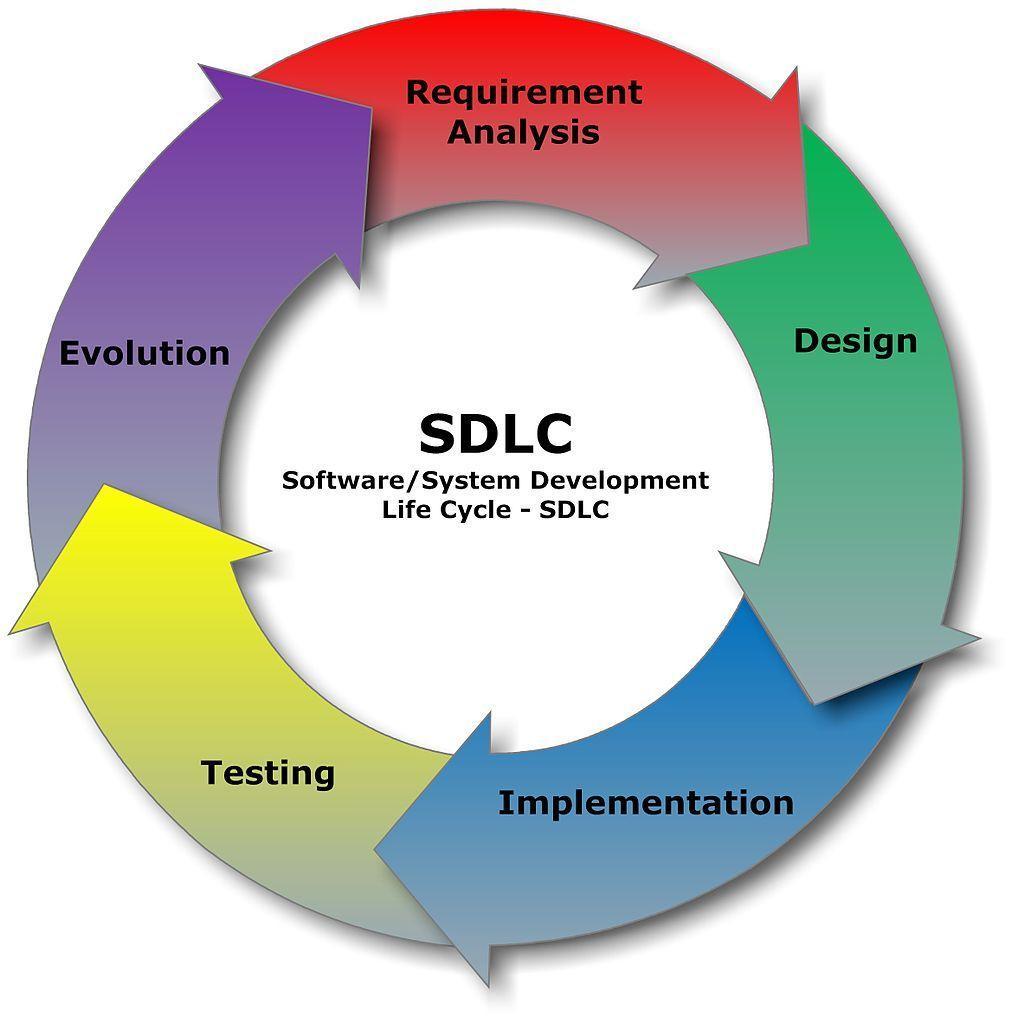
[Activity-1: Exploring Naan Mudhalavan Smart Interz Portal. 4](#_Toc208773738)

[Activity-2: Choose a IBM Granite model From Hugging Face. 7](#_Toc208773739)

[Activity-3: Running Application in Google Collab. 9](#_Toc208773740)

[Activity-4: Upload Your Project in GitHub. 15](#_Toc208773741)

SmartSDLC – AI-Enhanced Software Development Lifecycle



# Introduction

Software development can sometimes feel overwhelming — This project combines the power of **IBM Granite AI models** (via Hugging Face) with a simple and interactive interface built using **Gradio**. The goal is to assist developers at every stage of the Software Development Lifecycle (SDLC)—from requirement gathering to deployment.

Imagine uploading a PDF of requirements and instantly getting clear, structured insights. Or writing a natural language prompt and watching AI generate working code, complete with test cases. That’s what SmartSDLC brings to the table.

# Objectives

* Speed up requirement gathering with AI-powered document understanding
* Convert natural language prompts into clean, functional code
* Auto-generate test cases to save time in validation
* Help debug code and suggest fixes
* Write documentation with AI assistance
* Provide a seamless workspace using **Google Colab**

Abstract

The rapid growth of software development has increased the demand for tools that can generate, analyse, and optimize code efficiently. This project, AI Code Analysis & Generator, is designed to simplify the coding process by utilizing Artificial Intelligence to generate code snippets based on natural language prompts. The system allows users to provide a requirement in plain English, select a programming language, and receive automatically generated code.

In addition to code generation, the system can be extended to provide features such as line-by-line code explanation, error detection, optimization suggestions, and secure coding recommendations. By integrating vulnerability detection, it ensures that the generated code follows safe coding practices, reducing the risk of common security issues such as SQL injection and unsafe function calls. Furthermore, the tool has the potential to integrate with version control systems like GitHub and GitLab, enabling developers to directly push generated code to repositories.

The project also envisions future enhancements such as automatic language detection, domain-specific fine-tuning for areas like data science and web development, and reusable prompt templates for tasks like CRUD operations or API creation. With these capabilities, the AI Code Analysis & Generator is not only a code generation tool but also a smart coding assistant that improves productivity, enhances learning, and promotes secure software development practices.

# Project Workflow:

**Activity-1**: Exploring Naan Mudhalavan Smart Interz Portal.

**Activity-2:** Choosing a IBM Granite Model From Hugging Face.

**Activity-3:** Running Application In Google Colab.

**Activity-4:** Upload your Project in GitHub

## Activity-1: Exploring Naan Mudhalavan Smart Interz Portal.

* Search for “Naan Mudhalavan Smart Interz” Portal in any Browser

A screenshot of a computer

AI-generated content may be incorrect.

* Then Click on the first link. (Naanmudhalvan Smartinternz) Then login with your details.



* Then you will be redirected to your account then click on “Projects” Section. There you can see which project you have enrolled in here it is “SmartSDLC – AI-Enhanced Software Development Lifecycle”.

A screenshot of a computer

AI-generated content may be incorrect.

* Then click on “Access Resources” and go to the “Guided Project” Section.

A screenshot of a computer program

AI-generated content may be incorrect.

* Click on the “Go to workspace” section. Then you can find the detailed explanation of Generative AI Project using IBM WatsonX API key

A screenshot of a computer

AI-generated content may be incorrect.

* Click on “Project Workspace”, there you can find your project progress and Place to upload “Demo link”.

A screenshot of a computer

AI-generated content may be incorrect.

* Now we have gone through portal understanding, now lets find a IBM granite model from hugging face to integrate in our project.

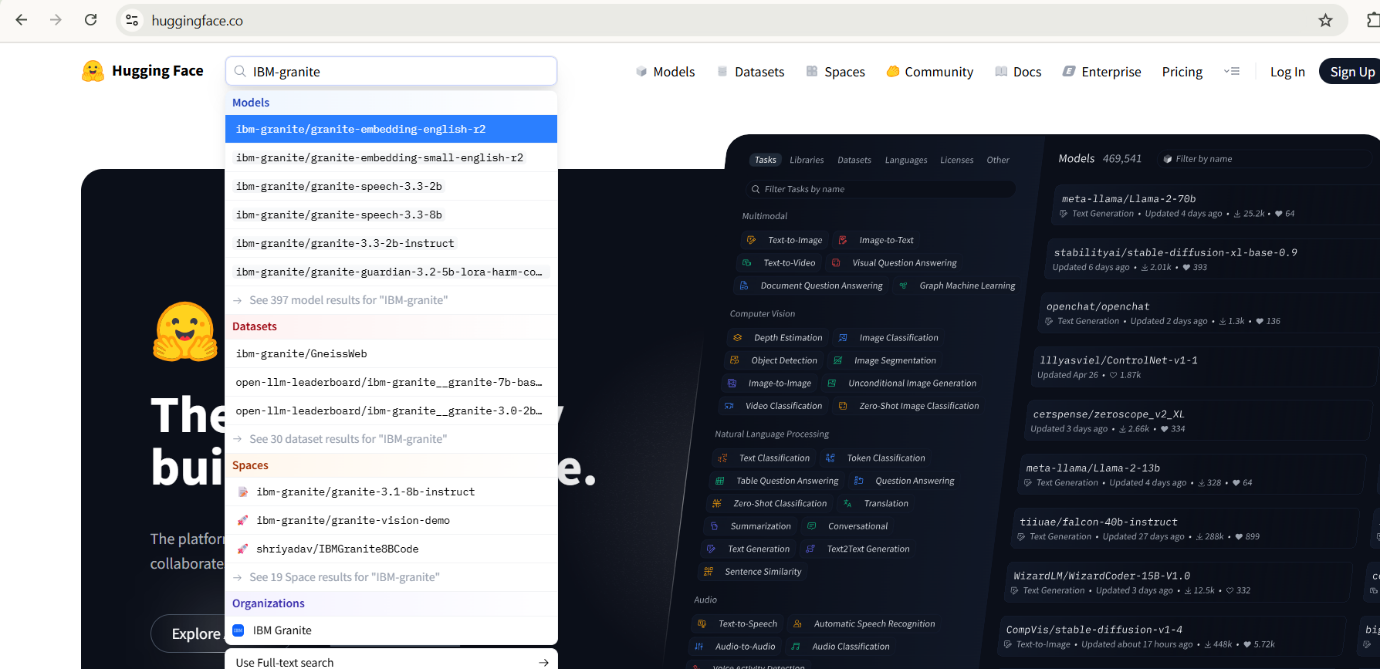
## Activity-2: Choose a IBM Granite model From Hugging Face.

* Search for “Hugging face” in any browser.

A screenshot of a computer

AI-generated content may be incorrect.

* Then click on the first link (Hugging Face), then click on signup and create your own account in Hugging Face. Then search for “IBM-Granite models” and choose any model.



* Here for this project we are using “granite-3.2-2b-instruct” which is compatible fast and light weight.

A screenshot of a computer

AI-generated content may be incorrect.

* Now we will start building our project in Google collab.

## Activity-3: Running Application in Google Collab.

* Search for “Google collab” in any browser.

**A screenshot of a computer

AI-generated content may be incorrect.**

* Click on the first link (Google Colab), then click on “Files” and then “Open Notebook”.

**A screenshot of a computer

AI-generated content may be incorrect.**

* Change the title of the notebook “Untitled” to “Health AI”. Then click on “Runtime”, then go to “Change Runtime Type”.

**A screenshot of a computer

AI-generated content may be incorrect.**

* Choose “T4 GPU” and click on “Save”

**A screenshot of a computer program

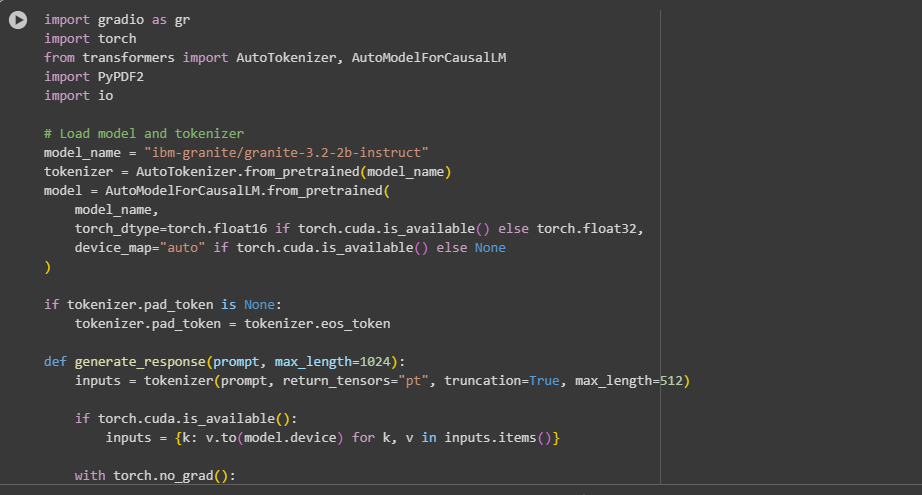
AI-generated content may be incorrect.**

Then run this command in the first cell “!pip install transformers torch gradio PyPDF2 -q”. To install the required libraries to run our application.

A screenshot of a computer

AI-generated content may be incorrect.

* Then run the rest of the code in the next cell



A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

* you can find the code here in this link: [SmartSDLC](https://drive.google.com/file/d/1M3RSuAKSAFmPBQSMHJh3-yoGMPVcLIhB/view)

A screenshot of a computer

AI-generated content may be incorrect.

Colab notebook detected. To show errors in colab notebook, set debug=True in launch()

\* Running on public URL: <https://adaf6f7bd3064b3929.gradio.live/>

* Click on the URl to open the Gradio Application click on the link

A screenshot of a computer

AI-generated content may be incorrect.

* You can View the Application is the running in the other tab

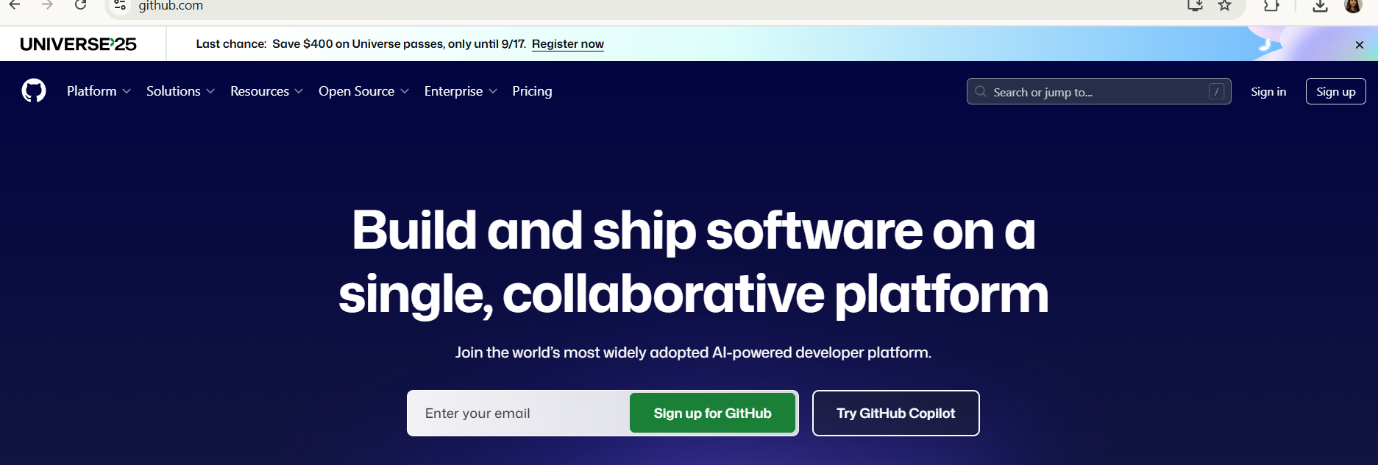
## Activity-4: Upload Your Project in GitHub.

* Search for “GitHub” in any browser, then click on the first link ([GitHub](https://github.com/))

A screenshot of a web page

AI-generated content may be incorrect.

* Then click on “Signup” and create your own account in GitHub. If you already have an account click on “Sign in”



* Click on “Create repository”.

A screenshot of a computer

AI-generated content may be incorrect.

* In “General” Name your repo. (Here I have given “IBM-Project” as my repo name and it is available)
* In “Configurations” Turn On “Add readme” file Option.

A screenshot of a computer

AI-generated content may be incorrect.

* Now Download your code from Google collab by Clicking on “File”, then Go to “Download” then download as “.py”.

A screenshot of a computer

AI-generated content may be incorrect.

* Then your repository is created, then Click on “Add file” Option. Then Click Upload files” to upload your files

A screenshot of a computer

AI-generated content may be incorrect.

* Click on “choose your files

A screenshot of a computer

AI-generated content may be incorrect.

* Choose your project file and click on “Open”.
* After your file has Uploaded Click on “Commit changes”.

A screenshot of a computer

AI-generated content may be incorrect.

Successfully uploaded code into GitHub.

# FUTURE ENHANCEMENT

* Along with code generation, provide a detailed explanation for each line, making it easier for learners and developers to understand the logic.
* Provide safe coding recommendations to encourage secure software development.
* Allow developers to push generated code directly to version control platforms like GitHub/GitLab for seamless collaboration and project management.
* Support fine-tuning of AI models for specialized areas such as data science, web development, or mobile app development.
* Introduce prompt templates (e.g., CRUD operations, API creation, data preprocessing) to save time and improve consistency.

# Pre-requisites:

* Gradio Framework Knowledge: [Gradio Documentation](https://www.gradio.app/docs)
* IBM Granite Models (Hugging Face): [IBM Granite models](https://huggingface.co/ibm-granite)
* Python Programming Proficiency: [Python Documentation](https://docs.python.org/3/)
* Version Control with Git: [Git Documentation](https://git-scm.com/docs/git)
* Google Collab’s T4 GPU Knowledge: [Google collab](https://www.geeksforgeeks.org/python/how-to-use-gpu-in-google-colab/)