

Introduction

1.1 Project Overview

Introduction

The SmartSDLC project is an advanced software development lifecycle (SDLC) assistant application that integrates IBM's Granite 3.3-2B Instruct language model to automate and streamline the software development process. Designed to empower developers, analysts, and project managers, SmartSDLC provides a unified interface for requirement analysis, code generation, testing, and AI-driven consulting, all accessible through a web interface built using Streamlit.

System Architecture

At its core, SmartSDLC utilizes:

- IBM Granite Model hosted locally using Hugging Face transformers.
- Streamlit for interactive UI.
- PyMuPDF (fitz) for PDF parsing and requirements extraction.
- Torch for GPU/CPU-based model inference.
- Ngrok for temporary tunneling and public access.

This architecture ensures that users can deploy the application without API key dependencies and run it entirely on local infrastructure—ideal for enterprise environments focused on privacy, performance, and security.

Key Functional Components

1. Requirement Analysis Module

- Extracts software requirements from PDF documents or plain text inputs.
- Automatically categorizes extracted data into functional, non-functional, and technical requirements.
- Generates user stories following enterprise modeling techniques.

2. Code Generation Module

- Accepts user prompts and generates language-specific and framework-specific backend or frontend code.
- Supports popular languages like Python, JavaScript, Java, and frameworks like React, Django, Flask, etc.
- Ensures generated code aligns with IBM coding and enterprise standards.

3. Test Generator Module

- Automatically creates unit, integration, API, or functional test cases.
- Accepts both raw code and requirement-based inputs.
- Outputs ready-to-use test files, with structured documentation and security considerations.

4. AI Assistant Module

- Acts as a consulting expert on software engineering, design patterns, best practices, microservices, CI/CD, and more.
- Maintains session context and generates professional-grade responses for development-related queries.

1.2 Purpose

Goal

The primary purpose of the SmartSDLC project is to revolutionize and automate key stages of the software development lifecycle by integrating artificial intelligence, specifically using IBM's Granite model. It aims to minimize human error, accelerate development, and bridge the gap between requirement gathering, design, development, and quality assurance.

Specific Objectives

1. Enhance Requirement Clarity
2. Transform vague, unstructured, or bulky requirements documents into clearly defined functional, non-functional, and technical components, reducing miscommunication between stakeholders and developers.
3. Accelerate Development with AI
4. Enable developers to convert business needs or prompts into clean, modular, and production-ready code tailored for a specified language and framework with best practices embedded.
5. Improve Testing Efficiency
6. Auto-generate comprehensive test suites for backend/frontend applications, minimizing the effort required by QA teams and increasing overall test coverage.
7. Provide Enterprise-Grade Insights
8. Offer AI-driven advice on architecture, security, optimization, database strategies, deployment, etc., tailored to enterprise environments.
9. Boost Developer Productivity
10. By reducing time spent on mundane, repetitive coding and documentation tasks, developers can focus more on logic, creativity, and value creation.

Benefits to Stakeholders

- For Business Analysts: Transforms business jargon into clear technical tasks.
- For Developers: Jumpstarts codebase creation and reduces boilerplate development.
- For Testers: Produces a foundation of test cases with less manual effort.
- For Managers: Ensures alignment of output with business goals and industry standards.

Enterprise Orientation

SmartSDLC emphasizes security, traceability, and standard compliance:

- Uses local inference to eliminate reliance on external APIs.
- Ensures privacy and isolation of sensitive documents and code.
- Aligns outputs with IBM's enterprise methodologies, providing trust and reliability in generated assets.