**Proposed Solution – SmartSDLC**

**Date**: 25 JUNE 2025  
**Team ID**: LTVIP2025TMID31783  
**Project Name**: SmartSDLC  
**Maximum Marks**: 2

**Problem Statement**

In software development education and project execution, students, mentors, and teams often struggle to map documentation to specific Software Development Life Cycle (SDLC) phases. This misalignment leads to confusion, ineffective feedback, and difficulty tracking progress across the project lifecycle. Manual classification of project artifacts into SDLC phases is time-consuming and error-prone.

**Idea / Solution Description**

**SmartSDLC** is an AI-powered tool that intelligently classifies project documentation into appropriate SDLC phases using advanced Natural Language Processing (NLP). The application is built using **Streamlit** for interactive UI and currently uses the **Google Gemini API** to simulate LLM-powered classification. Future iterations are planned to integrate **IBM Granite-3.3-2B-instruct** through **IBM Watson Machine Learning**.

**Key Features Include:**

* **Document Upload & Extraction**: Accepts PDF uploads and automatically extracts content for processing.
* **SDLC Phase Classification**: Classifies each paragraph or section of text into SDLC phases like Requirement, Design, Implementation, Testing, Deployment, or Maintenance using a zero-shot learning approach.
* **Visual Dashboard**: A Plotly-based analytics dashboard showing phase-wise distribution, document completeness, and segment summaries.
* **Correction Interface**: Allows manual correction of AI predictions with dropdown feedback.
* **Session Management**: Preserves extracted data, classifications, and user feedback during the app session.

**Novelty / Uniqueness**

* **AI-Based SDLC Mapping**: First-of-its-kind use of Generative AI to automate SDLC phase detection in documentation.
* **Single-Platform Solution**: Brings document extraction, AI classification, visualization, and correction in one cohesive tool.
* **Prompt Engineering for SDLC**: Uses custom prompt templates for SDLC-specific logic and definitions.
* **Streamlit-Based Feedback Loop**: Users can correct AI-labeled phases, enabling iterative learning and quality refinement.
* **Ready for IBM Watson ML**: Modular design allows for seamless integration with enterprise-grade AI services.

**Social Impact / Customer Satisfaction**

* **Enhances Learning**: Helps students and mentors understand how each document contributes to the SDLC.
* **Saves Time**: Speeds up the review process by highlighting misaligned or missing documentation areas.
* **Increases Accuracy**: Reduces human error in categorizing large or complex documents.
* **Enables Mentorship**: Facilitates informed feedback by mentors using phase-based summaries and analytics.
* **Empowers Teams**: Encourages documentation best practices and helps teams self-evaluate project progress.

**Business Model (Revenue Model)**

Though currently an academic prototype, SmartSDLC has commercialization potential through:

* **Institutional Licensing**: Offer as a yearly license to colleges, universities, or training institutes.
* **SaaS Subscription**: Tiered subscription plans for individual students, teams, or educational organizations.
* **Marketplace API**: Offer classification as an API service to educational tools, LMS platforms, or project management software.

**Scalability of the Solution**

* **Model-Ready Backend**: Easily switch between Gemini and IBM Granite models without changing frontend logic.
* **Cloud Deployable**: Works on Streamlit Cloud or can be ported to IBM Cloud for enterprise scaling.
* **Decoupled UI & Backend**: Backend classification logic and frontend dashboard operate independently for flexibility.
* **Extensible Features**: Future integration with GitHub, project management tools (e.g., Jira), or file exports (PDF, CSV).
* **Feedback-Driven Model Tuning**: Continuous learning through user correction of predictions (future ML fine-tuning pipeline).

**Conclusion**

**SmartSDLC** bridges a critical gap in project-based learning and development by aligning documentation with the structured phases of the SDLC. Its AI-first approach, modular design, and visual feedback loop make it a powerful tool for enhancing the accuracy, clarity, and accountability of software documentation in both academic and real-world contexts.