SmartSDLC-AI-Enhanced Software Development Lifecycle

1.Introduction:-

- Project title:Smart SDLC-AI-Enhanced Sbrtware Development Lifecycle
- Team member:R.Dhanasekaran
- Team member:C.Dhinesh
- Team member:C.Kumaran
- Team member:C.Sudharsan

2.project overview

- Purpose:

The purpose of a Smart SDLC AI Enhanced Software.Development Lifecycle is to empower cities
and their residents to thrive in a more eco-conscious and connected urban environment.By leveraging AI and real-time
data,the assistant helps optimizeessential resources like energy,water,and waste.

- Features:

Resource Forecasting:
Key Point:Predictive analytics
Functionality:Estimates future energy,water,and waste usage using historical and real-time data.
Eco-Tip Generator:-
Key Point:Personalized sustainability advice
Functionality:Recommends dally actions to reduce environmental impact based on user behavior.

3.Architecture:

Frontend (Stream lit):

The frontend is built with Stream lit,offering an interactive web UI with multiple pages including dashboards, file uploads,chat interface,feedback forms, and report viewers. Navigation is handled through a sidebar using the stream lit-option-menu library.Each page is modularized for scalability.

Backend (Fast API):

Fast API serves as the backend REST framework that powers API endpoints for document processing, chat interactions,eco tip generation,report creation, and vector embedding.It is optimized for asynchronous performance and easy

Swagger integration.

LLM Integration (IBM Watsonx Granite):

Granite LLM models from IBM Watsonx are used for natural language understanding and generation.Prompts are carefully designed to generate summaries sustainability tips,and reports.

Vector Search (Pinecone):

Uploaded policy documents are embedded using Sentence Transformers and stored in Pinecone.Semantic search is implemented using cosine similarity to allow users to search documents using natural language queries.

ML Modules (Forecasting and Anomaly Detection):

Lightweight ML models are used for forecasting and anomaly detection using Scikit-learn.Time-series data is parsed,modeled,and visualized using pandas and matplotlib.

4.Setup Instructions

Prerequisites:

- Python 3.9 or later
- pip and virtual environment tools
- API keys for IBM Watsonx and Pinecone
- Internet access to access cloud services

* Clone the repository
* Install dependencies from requirements.txt
* Create a.env file and configure credentials
* Run the backend server using Fast API
* Launch the frontend via Stream lit
* Upload data and interact with the modules

5.Folder Structure

app/- Contains all Fast API backen logic include routers,models,and integration modules.

app/api/- Subdirectory for modular API routes like chat,feedback,report,and document vectorization.

ui/- Contains frontend components for Stream lit pages,card layouts,and form UIs.

smart_dashboard.py- Entry script for launching the main Stream lit dashboard.

granite_llm.py-Handles all communication with IBM Watsonx Granite model including summarization and chat.

document_embedder.p - Converts documents to embeddings and stores in Pinecone.

kpi_file_forecaster.py- Forecasts future energy/water trends using regression.

report_generator.py-ConstructAI-generated sustainability reports.

6.Running the Application

To start the project:

* Run the Streamlit dashboard to access the web interface.

Navigatethroughpagesviathesidebar.

* Upload documents or CSVs,interact with the chat assistant,and view outputs like reports,summaries, and predictions.

>All interactions are real-time and use backend APIs to dynamically update the frontend.

Frontend (Stream lit):

The frontend is built with Stream lit,offering an interactive web UI with multiple pages including dashboards,file uploads,chat interface,feedback forms,and report viewers.Navigation is handled through a sidebar using the stream lit-option-menu library.Each page is modularized for scalability.

Backend (Fast API):

Fast API serves as the backend REST framework that powers API endpoints for document processing,chat interactions,eco tip generation,report creation, and vector embedding.It is optimized for asynchronous performance and easy Swagger integration.

7.API Documentation

Backend APIs available include:

POST /chat/ask-Accepts a user query and responds with an AI-generated message

POST /upload-doc-Uploads and embeds documents in Pinecone

GET /search-docs -Returns semantically similar policies to the input query

GET /get-eco-tips-Provides sustainability tips for selected topics like energy, water,or waste

8.Authentication

each endpoint is tested and documented in Swagger UI for quick inspection and trial during development.

This version of the project runs in an open environment for demonstration. However,secure deployments can integrate:

* Token-based authentication (JWT or API keys)
* OAuth2 with IBM Cloud credentials
* Role-based access (admin,citizen,researcher)
*. Planned enhancements include user sessions and history tracking.8. Authentication

9.User Interface

The interface is minimalist and functional,focusing on accessibility for non-technical users.It includes:

Sidebar with navigation

KPI visualizations with summary cards

Tabbed layouts for chat,eco tips,and forecasting

Real-time form handling

PDF report download capability

The design prioritizes clarity,speed,and user guidance with help texts and intuitive flows.

10.Testing

Testing was done in multiple phases:

Unit Testing:For prompt engineering functions and utility scripts

API Testing:Via Swagger UI,Postman,and test scripts

Manual Testing:For file uploads,chat responses,and output consistency

Edge Case Handling:Malformed inputs,large files,invalid API keys

Each function was validated to ensure reliability in both offline and API-connected modes.

Plastic waste

Generate Eco Tips

Generate practical and actionable eco-friendly tips for sustainable living related to: Plastic waste. Provide specific solutions and suggestions:

1. **Refuse Single-Use Plastics:**
 - Carry a reusable shopping bag for groceries and errands. Opt for cloth bags designed with pockets for easy storage.
 - Bring your reusable water bottle, thermos, or travel mug to limit plastic consumption from beverages.
 - Use reusable produce bags made of mesh or