High-Level to Low-Level Architecture Pipeline

(https://github.com/rahulogoel/High-to-Low-Level-Architecture)

Objective

Create an AI based automation tool which simplifies the process of converting high-level business requirements into low-level technical specifications

Technologies Used

- **Python**: The primary language used for scripting the tool.
- **Google Gemini API**: For generating the AI-driven specifications.
- **Libraries**: google-generativeai: To interact with the Gemini model.

Pipeline Overview

1. **Input parsing**: The user provides the business idea (e.g., “Build an e-commerce platform”).
2. **Interaction with Gemini API**: The tool sends the input as a prompt to the **Gemini API** for processing.
3. **Output generation**: The model processes the prompt and returns a breakdown into:
    - Modules to be implemented
    - Suggested database schema and relationships
    - Pseudocode for one or more features
4. **Output saving**: The tool saves the results in a text file (output_examples.txt) and also prints them to the console.

Input Examples

- Example input 1: "Build an app where users can create and share workout plans"

- Example input 2: "Create a platform where users can browse products, add them to a cart, purchase them, and track their orders."

For each example, show the generated outputs (modules, schemas, pseudocode) as they appeared in the console and in the saved output_examples.txt.

Results

Checkout the output_samples file (https://github.com/rahulogoel/High-to-Low-Level-Architecture)

Challenges

- Giving bad results on poorly structured business
- Not for complex business database requirements
- Model hallucinate while making database schemas

TODOS/Future Improvements

- Better Input Parsing by adding a GUI interface
- Fine-tune the AI model with more domain-specific training examples so that it produces better, more context-aware output.