



Project: Business Idea Creator

Objective: Design a prompt that generates innovative business ideas based on specific industries, target audiences, and market trends. This project aims to enhance students' abilities to formulate prompts that inspire creativity and entrepreneurial thinking.

Expected Outcome: A collection of prompts that yield unique business concepts, illustrating how to guide AI in generating innovative and viable business solutions.

OR

Project: Story Idea Generator

Objective: Create a prompt that generates creative writing prompts or story ideas based on genres, themes, and character types specified by the user. This project encourages students to think creatively and structure prompts that foster narrative development.

Expected Outcome: A collection of prompts that produce engaging story ideas, highlighting the students' skills in using AI to inspire creative writing.

OR

Project: Conversational AI Chatbot

Objective: Create a prompt framework for developing a conversational AI chatbot that can engage users on various topics, provide information, and offer recommendations. This project focuses on crafting conversational prompts that maintain context and coherence.

Expected Outcome: A set of prompts that enable the chatbot to have meaningful conversations, demonstrating students' ability to guide AI in conversational contexts.

For Students who have knowledge in web development should attempt this project:

Major Project for Prompt Engineering: "AI Assistant Development"

Objective:

In this project, you will design and implement a basic **AI Assistant** that can perform a variety of tasks based on user prompts. This project will allow you to showcase your skills in crafting effective prompts and understanding how different prompts can drive various responses from an AI model.

Project: Build Your Own AI Assistant

Your task is to create a command-line or web-based AI Assistant capable of performing tasks such as answering questions, providing summaries, generating text, or offering suggestions based on user input.

Requirements:

1. **Functionality:**

- Your AI Assistant should be able to perform at least **three distinct functions**.

Examples include:

- Answering factual questions (e.g., "What is the capital of France?")
- Summarizing a given text or article.
- Generating creative content (e.g., stories, poems, or essays).
- Providing advice on a specific topic (e.g., "What are some tips for studying effectively?").

2. **Prompt Design:**

- For each function, design at least **three different prompts** that effectively guide the AI in generating the desired output. Consider varying:
 - Length and specificity.
 - Tone and style.
 - Complexity and context.

3. **User Interaction:**

- Implement a user-friendly interface (command-line or simple web app) where users can:
 - Input their queries or commands.
 - Choose which function they want the assistant to perform.
 - Receive and view responses clearly.

4. **Feedback Loop:**

- Include a mechanism for users to provide feedback on the responses. For example:
 - "Was this response helpful? (yes/no)"
 - Use this feedback to refine the prompt design or suggest improvements in the AI's responses.

5. **Documentation:**

- Write a brief **user guide** explaining how to use your AI Assistant and detailing the different functions it can perform in a PPT.

Project Guidance: Building Your Own AI Assistant

Step-by-Step Guide

1. Setting Up the Environment:

- Choose your development environment. You can use:
 - A local IDE (e.g., Visual Studio Code, PyCharm).
 - Online platforms like Replit or Jupyter Notebook.
- Ensure you have the necessary libraries installed. If using Python, you may need:

```
pip install openai # For OpenAI API access
pip install flask   # If you are building a web-based assistant
```

2. Define the Functionality:

- Decide on the **three main functions** your AI Assistant will perform. For example:
 - **Answering Questions:** Use the AI to provide factual information.
 - **Summarizing Text:** Input a block of text and get a summary.
 - **Generating Creative Content:** Generate stories or poems based on user input.
- Outline the expected input and output for each function.

3. Crafting Prompts:

- For each function, create at least **three different prompts**. Consider:
 - **Answering Questions:**
 - "What is the capital of France?"
 - "Can you explain the significance of the Eiffel Tower?"
 - "Tell me three facts about Paris."
 - **Summarizing Text:**
 - "Summarize the following article: [insert article text]."
 - "What are the main points of this text: [insert text]?"
 - "Provide a brief overview of this document: [insert document]."
 - **Generating Creative Content:**
 - "Write a short story about a dragon and a princess."
 - "Create a poem about autumn."
 - "Generate an idea for a science fiction novel."

4. Building the Interface:

- **Command-Line Interface (CLI):**

- Use Python's built-in `input()` function to gather user input.
- Create a loop that allows users to select a function and input their queries.
- Example:

```
def main():
    while True:
        print("Welcome to your AI Assistant!")
        print("1. Answer Questions")
        print("2. Summarize Text")
        print("3. Generate Creative Content")
        print("4. Exit")
        choice = input("Select an option: ")
        # Handle user choices...
```

- **Web Interface (using Flask):**

- Create a simple web app with HTML forms to capture user input.
- Set up routes to handle requests and return responses.
- Example:

```
from flask import Flask, request, render_template

app = Flask(__name__)

@app.route('/', methods=['GET', 'POST'])
def index():
    if request.method == 'POST':
        user_input = request.form['user_input']
        # Process input and return response...
    return render_template('index.html')
```

5. Implementing Feedback Mechanism:

- After providing a response, ask users if it was helpful. Collect their input.
- Store the feedback in a simple list or log file for later analysis.

6. Testing Your Assistant:

- Test each function thoroughly. Ensure that prompts yield the expected results.
- Refine your prompts based on the AI's responses and user feedback.