Custom Itineraries for Your Next Journey

Google Cloud Generative AI

MENTOR NAME : **SRI LANKA LAKSHMI NARAYANA**

FACULTY MENTOR NAME : **K.RATNA KUMARI**

TEAM ID : **LTVIP2026TMIDS38636**

MAIL : **phanianjali066@gmail.com**

ID NO : **SBAP0032724**

Project Submitted By

"B.P.S.S.D.RAMYA ANJALI"

Introduction

Travel planning can be overwhelming and time-consuming due to the extensive research and customization required to create detailed and personalized itineraries. Travelers often face challenges in finding reliable and tailored travel recommendations that suit their preferences, interests, and constraints. This problem is exacerbated for travel agencies and tour operators who need to generate high-quality itineraries efficiently to meet diverse client needs. A solution that automates and personalizes the creation of travel itineraries can streamline this process, providing valuable assistance to both individual travelers and industry professionals.

Scenario 1: Personalized Travel Itineraries for Individual Travelers

TravelGuideAI can transform how individual travelers plan their vacations by offering personalized itinerary generation. Travelers enter their destination, trip duration, and specific interests into the app. The AI uses this information to create a detailed, tailored itinerary, including daily activities, local dining options, and key attractions. This personalized approach simplifies the planning process, allowing travelers to easily organize and enjoy a customized trip without extensive manual research.

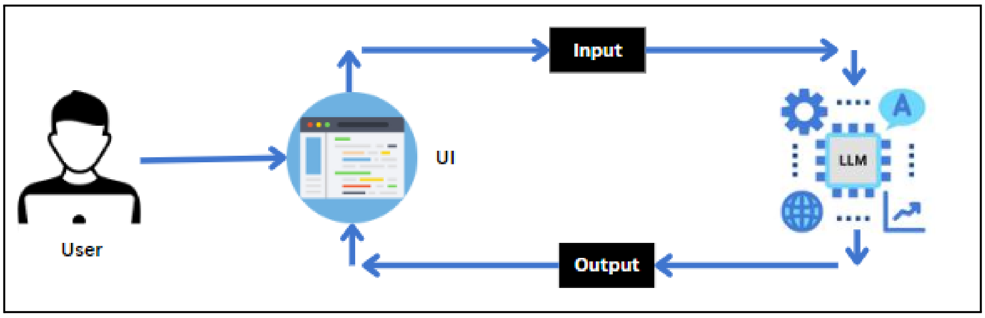
Scenario 2: Streamlined Itinerary Creation for Travel Agencies

Incorporating TravelGuideAI into the operations of travel agencies can significantly enhance efficiency. Agencies input client preferences and trip details into the system, and the AI generates comprehensive itineraries automatically. This feature allows agencies to quickly produce high-quality, customized travel plans for various clients. By automating itinerary creation, travel agents can focus more on client relationships and strategic planning, improving service delivery and client satisfaction.

Scenario 3: Engaging Travel Content for Websites and Blogs

TravelGuideAI can be integrated into travel websites and blogs to produce dynamic and engaging content. The AI generates well-researched articles, destination guides, and travel tips based on current trends and audience interests. This integration helps maintain a continuous stream of fresh, relevant content, attracting and retaining readers. By automating content generation, travel sites and blogs can enhance their online presence and engage their audience more effectively.

Architecture



This diagram illustrates a high-level **Generative AI Workflow**, specifically focusing on how a user interacts with a Large Language Model (LLM) through a digital interface.

Here is a breakdown of the process:

The Interaction Cycle

The architecture follows a continuous loop, starting and ending with the user.

- **User:** The person initiating the request. This represents the starting point of the human-computer interaction.
- **UI (User Interface):** This is the bridge between the human and the machine (e.g., a chatbot window, a mobile app, or a web dashboard). It captures the user's intent and displays the final result.

Summary of the Loop

1. **Request:** User types into the **UI**.
2. **Processing:** The request travels as **Input** to the **LLM**.
3. **Generation:** The **LLM** computes the response.
4. **Delivery:** The generated **Output** travels back to the **UI** for the user to read.

Project Flow

- Users enter their desired travel destination, the number of days, and nights they plan to stay into the Streamlit UI. Additional preferences or interests can also be specified if needed.

- The input details are sent to the TravelGuideAI backend, which utilizes the generative AI model to process the information.
- The AI model processes the user’s input to generate a detailed and personalized travel itinerary based on the specified destination, duration, and interests.
- The AI autonomously creates a well-structured and engaging travel itinerary, including daily activities, local attractions, dining recommendations, and other relevant travel tips.
- The generated itinerary is sent back to the frontend of the Streamlit app for display to the user.
- Users can review the generated itinerary, make additional customizations if desired, and either export or copy the content for their travel planning.

To accomplish this, we have to complete all the activities listed below,

- Initialize Gemini Pro LLM:
  - Generate Gemini Pro API
  - Initialize the pre-trained model
- Interfacing with Pre-trained Model
  - Travel itinerary Generation
- Model Deployment
  - Deploy the application using Streamlit

Prior Knowledge

You must have prior knowledge of the following topics to complete this project.

- LLM & Gemini Pro:

A **large language model** is a type of artificial intelligence algorithm that applies neural network techniques with lots of parameters to process and understand human languages or text using self-supervised learning techniques. Tasks like text generation, machine translation, summary writing, image generation from texts, machine coding, chat-bots, or Conversational AI are applications of the Large Languag.e Model. Examples of such LLM models are Chat GPT by open AI, BERT (Bidirectional Encoder Representations from Transformers) by Google, etc.

https://www.geeksforgeeks.org/large-language-model-llm/

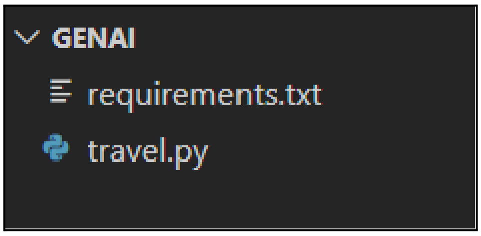
https://cloud.google.com/vertex-ai/docs/generative-ai/learn-resources

- Streamlit:

Basic knowledge of building interactive web applications using Streamlit.
Understanding of Streamlit's UI components and how to integrate them with backend logic.
https://www.datacamp.com/tutorial/streamlit

Project Structure

Create the Project folder which contains application file as shown below

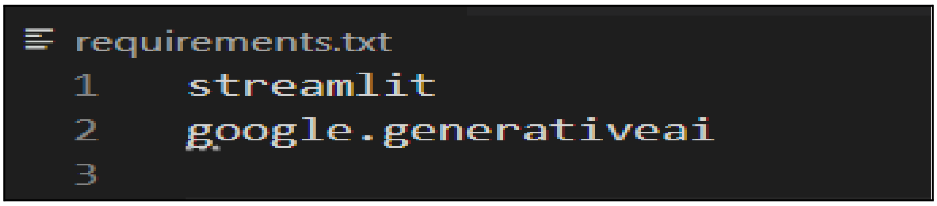


```
GENAI
    requirements.txt
    travel.py
```

Milestone 1: Requirements Specification

Specifying the required libraries in the requirements.txt file ensures seamless setup and reproducibility of the project environment, making it easier for others to replicate the development environment.

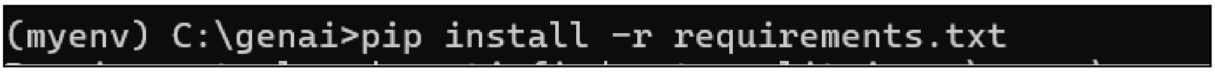
Activity 1: Create a requirements.txt file to list the required libraries.



requirements.txt

```
streamlit
google.generativeai

```

Activity 2: Install the required libraries.



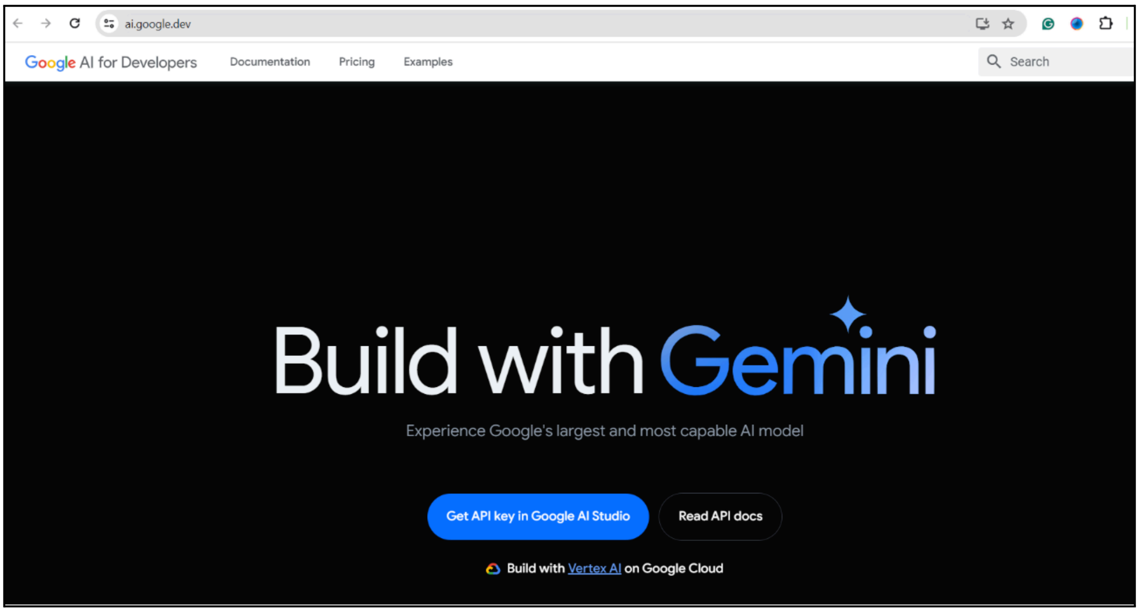
```
(myenv) C:\genai>pip install -r requirements.txt
```

Milestone 2: Initializing the Model

For initializing the model we need to generate PALM API.

Activity 1: Generate PALM API

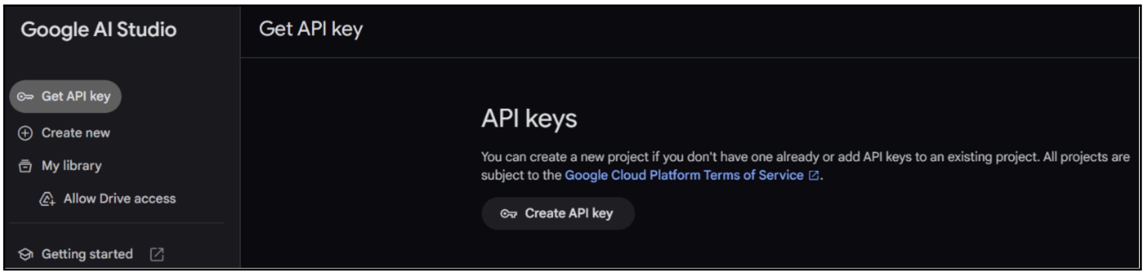
- Click on the link (https://developers.generativeai.google/).
- Then click on “Get API key in Google AI Studio”.
- Click on “Get API key” from the right navigation menu.
- Now click on “Create API key”. (Refer the below images)
- Copy the API key.



Build with Gemini

Experience Google's largest and most capable AI model

Build with Vertex AI on Google Cloud

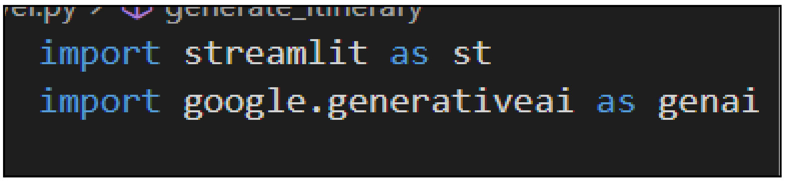


API keys

You can create a new project if you don't have one already or add API keys to an existing project. All projects are subject to the Google Cloud Platform Terms of Service.

Activity 2 : Initialize The Pre - Trained Model

Activity 2.1: Import necessary files

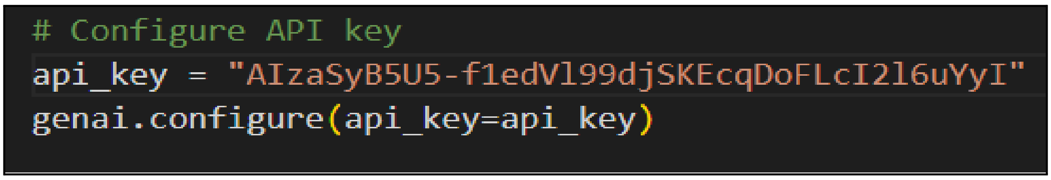


```
import streamlit as st
import google.generativeai as genai
```

- Streamlit, a popular Python library, is imported as st, enabling the creation of user interfaces directly within the Python script.

Google Generative AI (genai): Imported to interact with the Gemini Pro model.

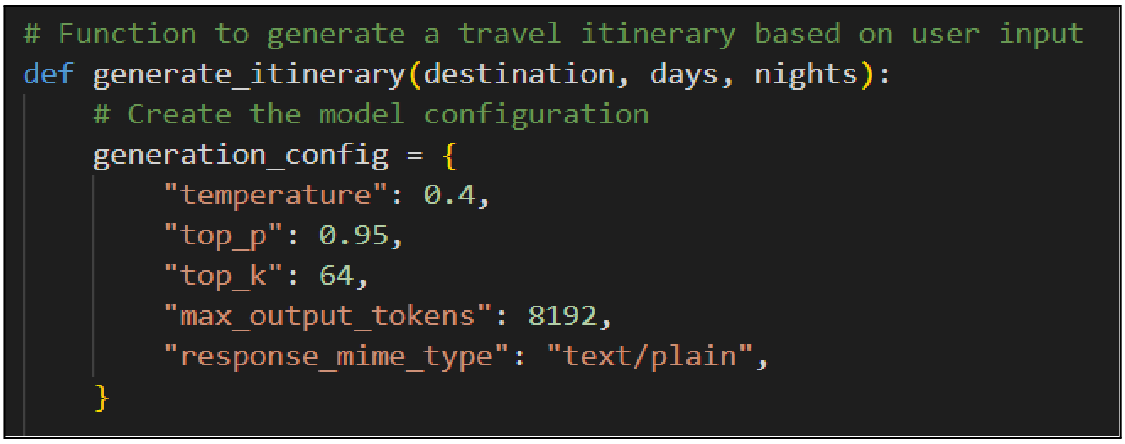
Activity 2.2: Configuration of the Gemini Pro API



```
# Configure API key
api_key = "AIzaSyB5U5-f1edVl99djSKEcqDoFLcI2l6uYyI"
genai.configure(api_key=api_key)
```

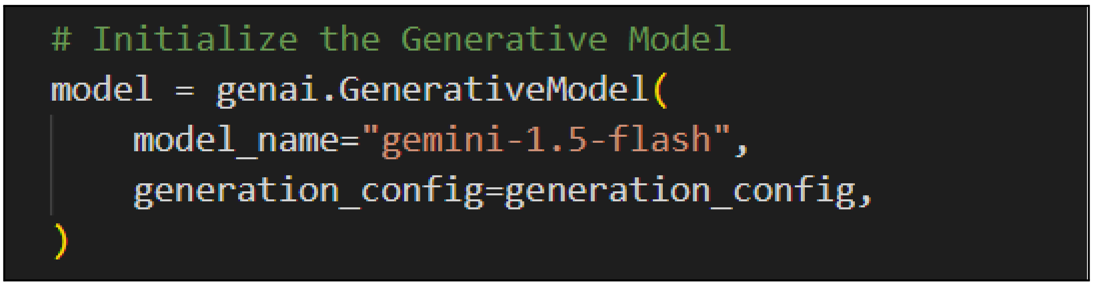
Configuring the API key: The configure function is used to set up or configure the Google API with an API key. The provided API key, in this case, is "AIzaSyB5U5-f1edVl99djSKEcqDoFLcIXXXXXX".

Activity 2.3: Define the model to be used



```python
# Function to generate a travel itinerary based on user input
def generate_itinerary(destination, days, nights):
    # Create the model configuration
    generation_config = {
        "temperature": 0.4,
        "top_p": 0.95,
        "top_k": 64,
        "max_output_tokens": 8192,
        "response_mime_type": "text/plain",
    }
```

- **Generation Settings:** Configured parameters such as temperature, top_p, top_k, max_output_tokens, and response_mime_type to control the output characteristics.



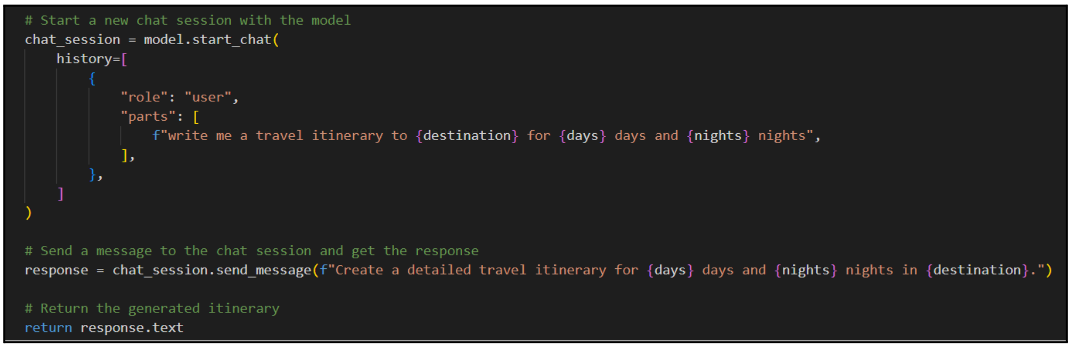
```python
# Initialize the Generative Model
model = genai.GenerativeModel(
    model_name="gemini-1.5-flash",
    generation_config=generation_config,
)
```

- **Created an instance of GenerativeModel with the model_name set to "gemini-1.5-flash".**

Milestone 3: Interfacing with Pre-trained Model

In this milestone, we will build a prompt template to generate feedback based on the project details entered by the user.

Activity 1: Create a function to generate travel guide



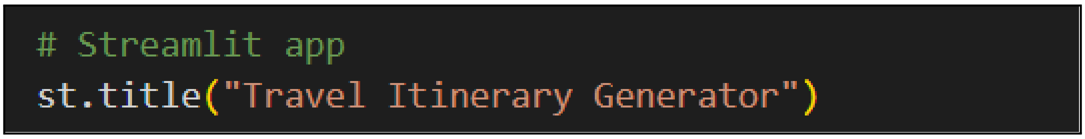
```python
# Start a new chat session with the model
chat_session = model.start_chat(
    history=[
        {
            "role": "user",
            "parts": [
                f"write me a travel itinerary to {destination} for {days} days and {nights} nights",
            ],
        },
    ]
)

# Send a message to the chat session and get the response
response = chat_session.send_message(f"Create a detailed travel itinerary for {days} days and {nights} nights in {destination}.")

# Return the generated itinerary
return response.text
```

- The generate_itinerary function is designed to create a travel itinerary based on the user's specified destination, number of days, and nights.
- The function takes three parameters: destination (the travel location), days (the number of days for the trip), and nights (the number of nights to stay).
- It starts a chat session with the AI model using the provided details, initializing the session with a user message that specifies the request for a travel itinerary.
- The function sends a detailed message to the chat session to generate the travel itinerary, then retrieves and returns the generated content as text.

Milestone 4: Model Deployment

In this milestone, we are deploying the created model using streamlit. Model deployment using Streamlit involves creating a user-friendly web interface, enabling users to interact with the model through a browser. Streamlit provides easy-to-use tools for developing and deploying data-driven applications, allowing for seamless integration of models into web-based applications.

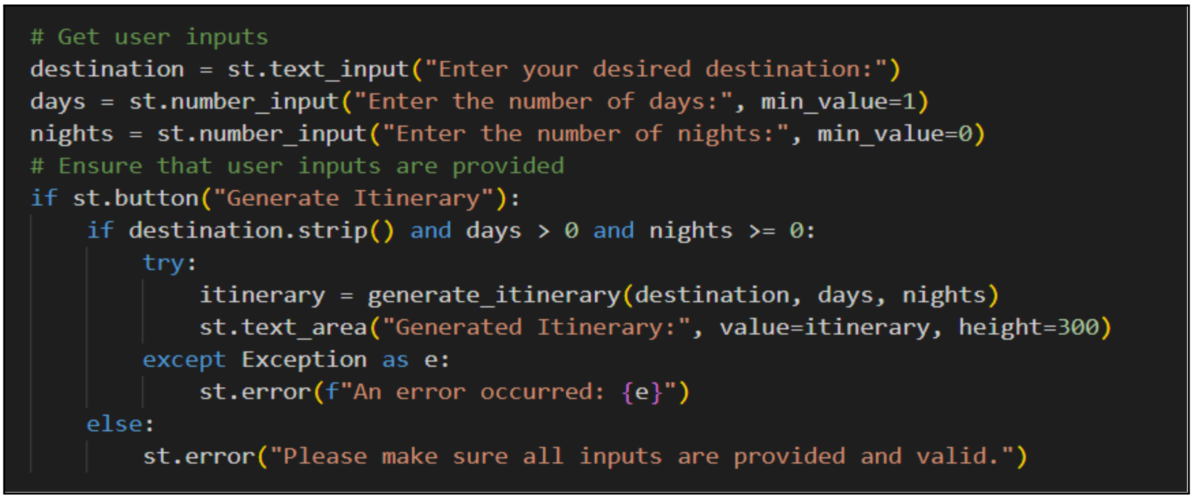
Activity 1: Give the project title



```
# Streamlit app
st.title("Travel Itinerary Generator")
```

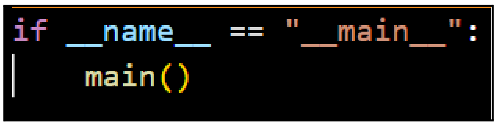
- The main function to start the Streamlit application, allowing users to interact with the web app.
- `st.title("Travel Itinerary Generator")`
- Sets the title of the Streamlit app to "Travel Itinerary Generator." This title is prominently displayed at the top of the web page, providing users with a clear understanding of the app’s purpose and functionality.

Activity 2: Create fields for user to input data for generating blog



```python
# Get user inputs
destination = st.text_input("Enter your desired destination:")
days = st.number_input("Enter the number of days:", min_value=1)
nights = st.number_input("Enter the number of nights:", min_value=0)
# Ensure that user inputs are provided
if st.button("Generate Itinerary"):
    if destination.strip() and days > 0 and nights >= 0:
        try:
            itinerary = generate_itinerary(destination, days, nights)
            st.text_area("Generated Itinerary:", value=itinerary, height=300)
        except Exception as e:
            st.error(f"An error occurred: {e}")
    else:
        st.error("Please make sure all inputs are provided and valid.")
```

- destination = st.text_input("Enter your desired destination:")
  Creates a text input field labeled "Enter your desired destination:" where users can type in their travel destination. The default value is an empty string.
- days = st.number_input("Enter the number of days:", min_value=1)
  Provides a number input field labeled "Enter the number of days:" for users to specify the duration of their trip in days. The minimum value is 1.
- nights = st.number_input("Enter the number of nights:", min_value=0)
  Provides a number input field labeled "Enter the number of nights:" for users to specify the number of nights for their stay. The minimum value is 0.
- if st.button("Generate Itinerary"):
  Adds a button labeled "Generate Itinerary." When clicked, the app processes the input and generates a travel itinerary based on the provided destination, days, and nights.
- if destination.strip() and days > 0 and nights >= 0:
  Checks if the destination field is not empty, and that days are greater than 0 and nights are non-negative. This ensures that all inputs are provided and valid.
- itinerary = generate_itinerary(destination, days, nights)
  Calls the generate_itinerary function with the destination, days, and nights provided by the user. This function generates the itinerary and stores it in the itinerary variable.
- st.text_area("Generated Itinerary:", value=itinerary, height=300)
  Displays the generated itinerary in a text area on the web page, allowing users to view the detailed itinerary.
- else:
  Executes if any input field is empty or invalid.
- st.error("Please make sure all inputs are provided and valid.")
  Shows an error message asking the user to ensure that all inputs are provided and valid if any field is empty or invalid.

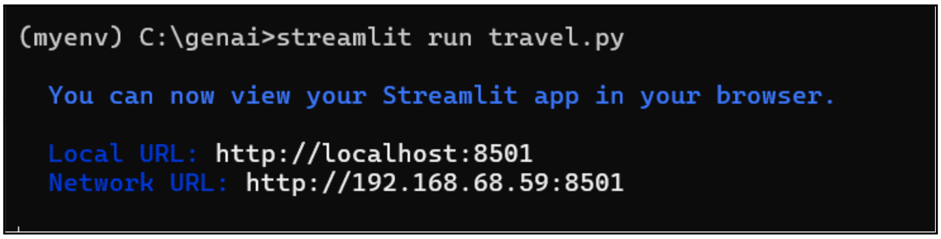


```python
if __name__ == "__main__":
    main()
```

- Finally, the main() function is called to execute the Streamlit app.

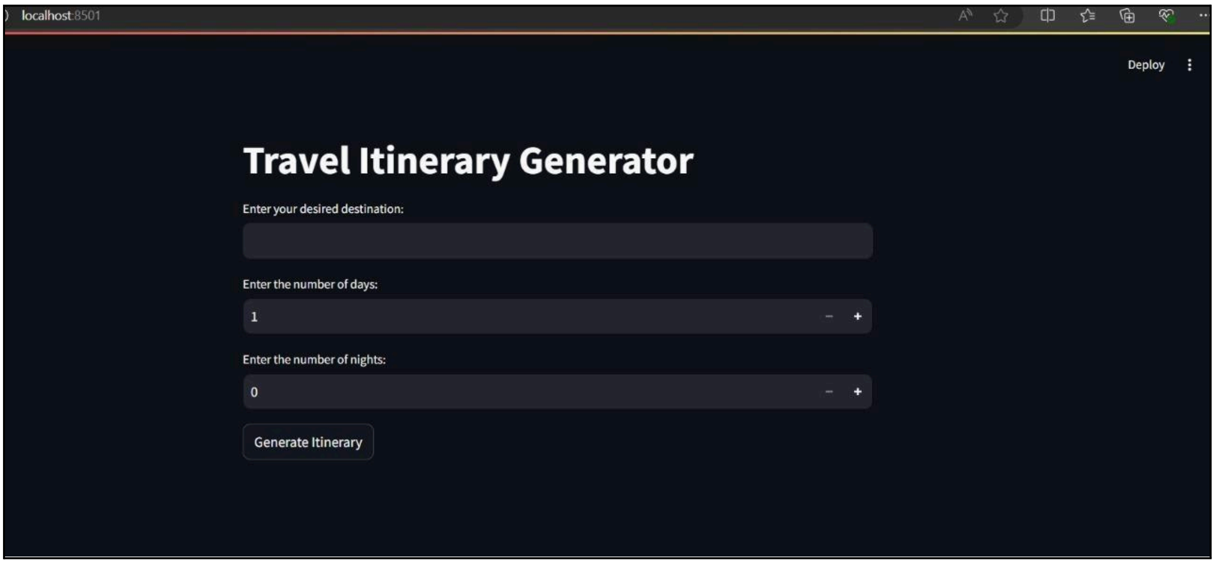
Activity 5: Run the web application

- Open the anaconda prompt from the start menu
- Navigate to the folder where your Python script is.
- Now type "streamlit run app.py" command
- Navigate to the localhost where you can view your web page



```
(myenv) C:\genai>streamlit run travel.py

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://192.168.68.59:8501
```

Now, the application will open in the web browser,



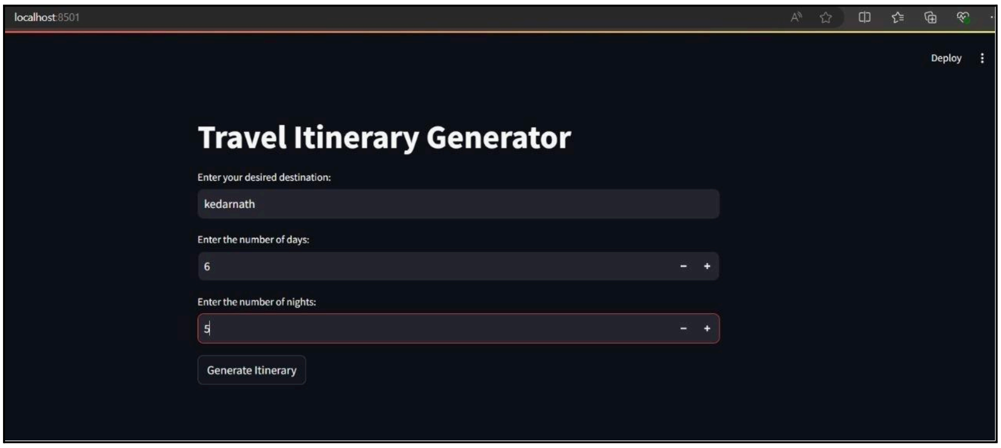
Travel Itinerary Generator

Enter your desired destination:

Enter the number of days: 1

Enter the number of nights: 0

After giving the input:



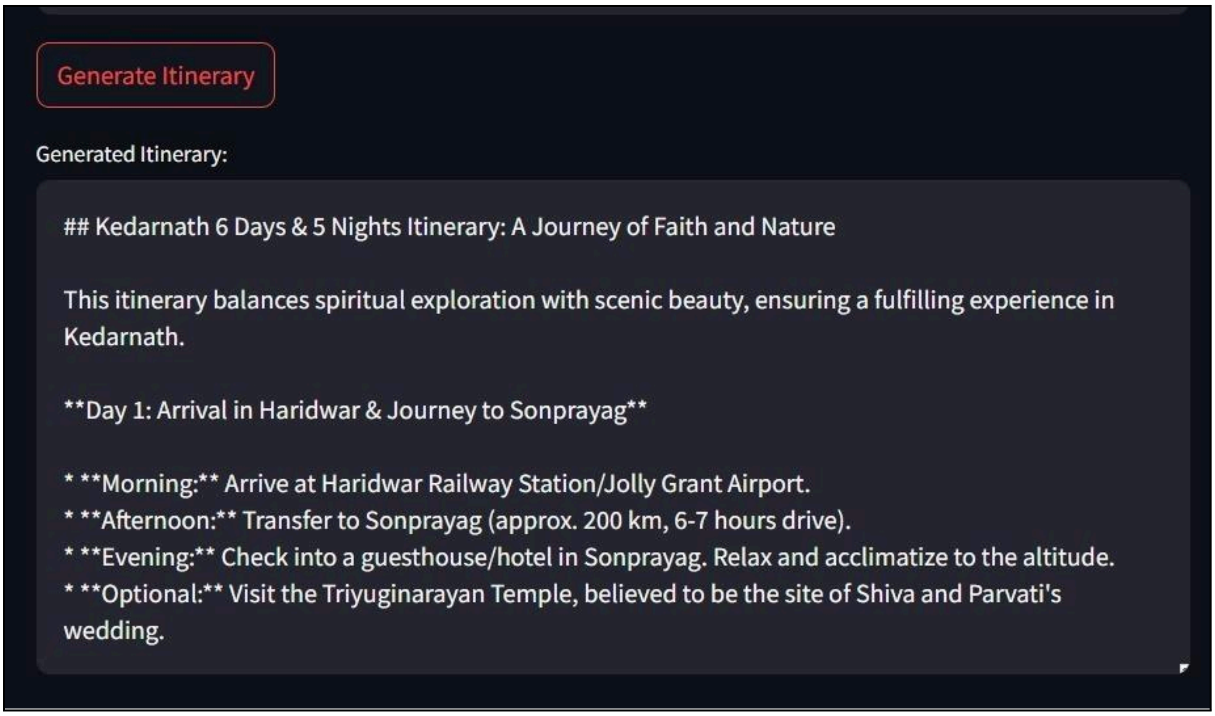
Travel Itinerary Generator

Enter your desired destination: kedarnath

Enter the number of days: 6

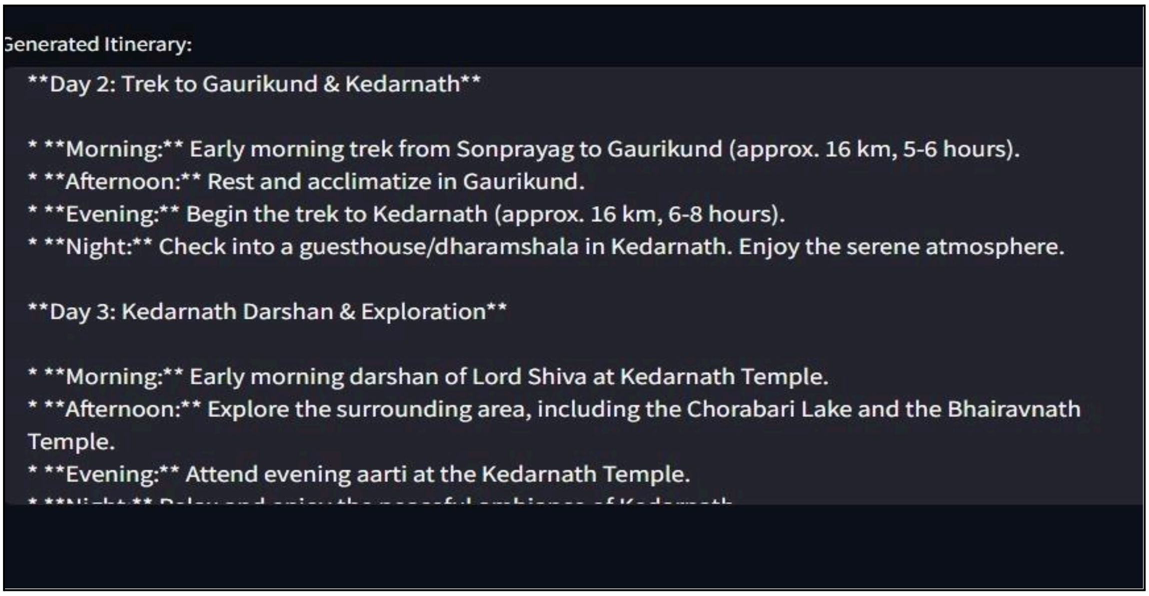
Enter the number of nights: 5

The Output generated:



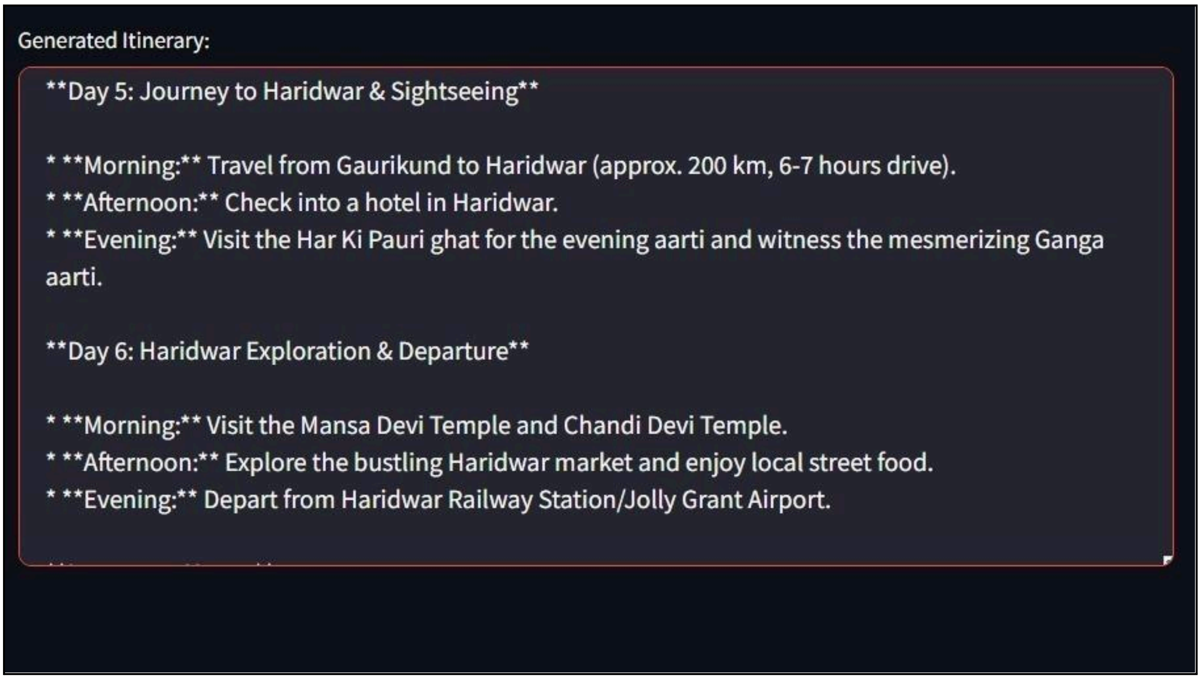
Generated Itinerary:

```
## Kedarnath 6 Days & 5 Nights Itinerary: A Journey of Faith and Nature

This itinerary balances spiritual exploration with scenic beauty, ensuring a fulfilling experience in Kedarnath.

**Day 1: Arrival in Haridwar & Journey to Sonprayag**

* **Morning:** Arrive at Haridwar Railway Station/Jolly Grant Airport.
* **Afternoon:** Transfer to Sonprayag (approx. 200 km, 6-7 hours drive).
* **Evening:** Check into a guesthouse/hotel in Sonprayag. Relax and acclimatize to the altitude.
* **Optional:** Visit the Triyuginarayan Temple, believed to be the site of Shiva and Parvati's wedding.
```



Generated Itinerary:

```
**Day 2: Trek to Gaurikund & Kedarnath**

* **Morning:** Early morning trek from Sonprayag to Gaurikund (approx. 16 km, 5-6 hours).
* **Afternoon:** Rest and acclimatize in Gaurikund.
* **Evening:** Begin the trek to Kedarnath (approx. 16 km, 6-8 hours).
* **Night:** Check into a guesthouse/dharamshala in Kedarnath. Enjoy the serene atmosphere.

**Day 3: Kedarnath Darshan & Exploration**

* **Morning:** Early morning darshan of Lord Shiva at Kedarnath Temple.
* **Afternoon:** Explore the surrounding area, including the Chorabari Lake and the Bhairavnath Temple.
* **Evening:** Attend evening aarti at the Kedarnath Temple.
* **Night:** Relax and enjoy the peaceful ambiance of Kedarnath.
```



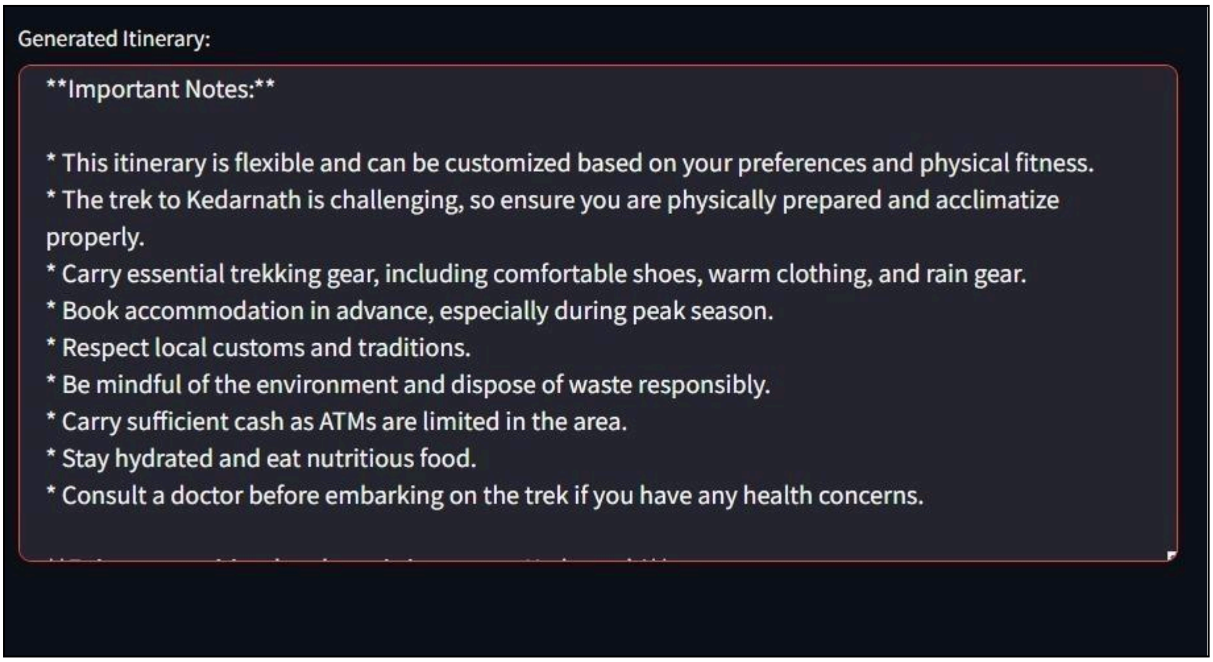
Generated Itinerary:

Day 5: Journey to Haridwar & Sightseeing

* **Morning:** Travel from Gaurikund to Haridwar (approx. 200 km, 6-7 hours drive).
* **Afternoon:** Check into a hotel in Haridwar.
* **Evening:** Visit the Har Ki Pauri ghat for the evening aarti and witness the mesmerizing Ganga aarti.

Day 6: Haridwar Exploration & Departure

* **Morning:** Visit the Mansa Devi Temple and Chandi Devi Temple.
* **Afternoon:** Explore the bustling Haridwar market and enjoy local street food.
* **Evening:** Depart from Haridwar Railway Station/Jolly Grant Airport.



Generated Itinerary:

Important Notes:

* This itinerary is flexible and can be customized based on your preferences and physical fitness.
* The trek to Kedarnath is challenging, so ensure you are physically prepared and acclimatize properly.
* Carry essential trekking gear, including comfortable shoes, warm clothing, and rain gear.
* Book accommodation in advance, especially during peak season.
* Respect local customs and traditions.
* Be mindful of the environment and dispose of waste responsibly.
* Carry sufficient cash as ATMs are limited in the area.
* Stay hydrated and eat nutritious food.
* Consult a doctor before embarking on the trek if you have any health concerns.

Conclusion

The Travel Itinerary Generator project is an intuitive web application designed to simplify travel planning through the power of AI. By integrating Streamlit with Google's generative AI model, users can easily create customized travel itineraries by entering their destination, the number of days, and nights. The application ensures a seamless experience with real-time validation of inputs and generates detailed, tailored itineraries to meet user preferences. This project showcases the practical use of AI in enhancing travel planning, offering a straightforward and interactive tool for crafting personalized travel experiences.

COMMENTS

FIRST AND FOREMOST, I SINCERELY GRATITUDE TO OUR ESTEEMED INSTITUTE SRI VASAVI DEGREE COLLEGE, FOR GIVING ME THIS OPPORTUNITY TO FULFILL OUR WARM DREAM TO BECOME A GRADUATE. OUR SINCERE GRADITUDE TO OUR LONG-TERM INTERNSHIP GUIDE **SRI L LAKSHMI NARAYANA**, LECTURER DEPARTMENT OF COMPUTER SCIENCE FOR TIMELY COOPERATION AND VALUABLE SUGGESTIONS WHILE CARRYING OUT THIS INTERNSHIP.

I EXPRESS MY SINCERE THANKS AND HEARTFUL GRATITUDE TO **SRI L LAKSHMI NARAYANA**, HOD IN COMPUTER SCIENCE FOR PERMITTING ME TO DO MY PROJECT INTERNSHIP. I EXPRESS MY SINCERE THANKS AND HEARTFUL GRATITUDE TO **SRI M RAMA KRISHNA**, PRINCIPAL FOR PROVIDING A FAVOURABLE ENVIRONMENT AND SUPPORTING ME DURING THE DEVELOPMENT OF THIS INTERNSHIP.

THANK YOU,SMART BRIDGE

----BANDI PHANI SIVA SAI DEEPIKA RAMYA ANJALI
TEAM LEADER

THE END

SIGNATURE OF THE HOD

SIGNATURE OF THE PRINCIPAL