

EXPLORE WITH AI -

Custom Itineraries for Your Next Journey

Topics :

- 1. Abstract**
- 2. Introduction**
- 3. Problem Statement**
- 4. Objectives**
- 5. Literature Review**
- 6. System Architecture**
- 7. Project Flow**
- 8. Technologies Used**
- 9. Project Structure**
- 10. Implementation**
- 11. Use Case Scenarios**
- 12. Features of the System**
- 13. Advantages**
- 14. Limitations**
- 15. Future Enhancements**
- 16. Conclusion**

1. ABSTRACT

Travel planning is often a complex and time-consuming process that requires detailed research, comparison of options, and organization of activities. Many travelers struggle to create structured and personalized itineraries that match their interests, time constraints, and preferences. Travel agencies also spend significant time preparing customized travel plans for clients.

This project, Explore with AI, presents an AI-powered travel itinerary generator that automates the creation of personalized, day-wise travel plans. The system integrates Google's Gemini Large Language Model (LLM) with a Streamlit-based web application to generate structured travel itineraries based on user inputs such as destination and duration.

The application demonstrates the practical implementation of Generative AI in solving real-world travel planning challenges efficiently and intelligently.

2. INTRODUCTION

Planning a trip involves selecting destinations, identifying tourist attractions, arranging accommodation, and organizing daily schedules. Traditional travel planning requires extensive manual research, which can be overwhelming for many users.

With advancements in Artificial Intelligence, especially Large Language Models (LLMs), it is now possible to automate content generation tasks such as itinerary creation. Generative AI models like Gemini can understand user input and produce meaningful, structured responses.

This project leverages the power of Generative AI to simplify travel planning through an interactive web application built using Streamlit.

3. PROBLEM STATEMENT

1. Travel planning faces several challenges:
 - Time-consuming research process
 - Lack of personalization
 - Difficulty in organizing day-wise activities
 - Manual effort required by travel agencies
 - Limited access to customized recommendations
2. There is a need for an intelligent system that can:
 - Automatically generate structured itineraries
 - Provide personalized recommendations
 - Reduce manual effort
 - Improve travel planning efficiency

4. OBJECTIVES

The main objectives of this project are:

- To develop an AI-based travel itinerary generator
- To integrate Google Gemini LLM into a web application
- To provide personalized travel plans
- To automate itinerary creation
- To deploy the model using Streamlit

5. LITERATURE REVIEW

5.1 Large Language Models (LLM)

Large Language Models are AI systems trained on massive datasets to understand and generate human-like text. They use deep learning techniques and transformer architectures.

Examples:

- ChatGPT

- BERT
- Gemini

LLMs are widely used for:

- Text generation
- Chatbots
- Content writing
- Code generation

5.2 Google Gemini

Gemini is a Generative AI model developed by Google. It is capable of understanding prompts and generating structured responses. In this project, Gemini is used to generate personalized travel itineraries.

5.3 Streamlit

Streamlit is a Python framework used to build interactive web applications quickly. It allows developers to create web apps using simple Python scripts without needing HTML or JavaScript.

6. SYSTEM ARCHITECTURE



The system architecture follows this structure:

User → Streamlit UI → Backend Application → Gemini AI Model → Generated Output → User

Explanation:

1. User enters travel details.
2. The input is processed by the backend.

3. The backend sends the request to Gemini AI.
4. The AI generates a structured itinerary.
5. The output is displayed to the user.

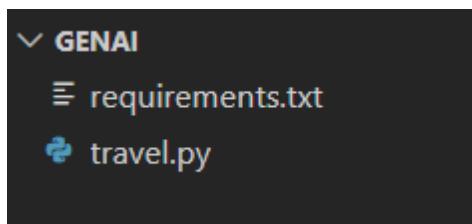
7. PROJECT FLOW

1. User opens the Streamlit application.
2. User enters:
 - o Destination
 - o Number of days
3. The input is sent to the Gemini model.
4. The model processes the request.
5. A personalized, day-wise itinerary is generated.
6. The itinerary is displayed on the screen.
7. User can copy or use the itinerary for planning.

8. TECHNOLOGIES USED

- Python
- Streamlit
- Google Gemini API
- Environment Variables

8. PROJECT STRUCTURE



```
GENAI/  
└── travel.py  
└── requirements.txt
```

9.1 travel.py

- Contains main application logic

- Initializes Gemini model
- Configures API key
- Generates itinerary
- Displays output

9.2 requirements.txt

- Lists required Python libraries
- Ensures reproducibility

10. IMPLEMENTATION

10.1 Milestone 1 – Requirements Specification

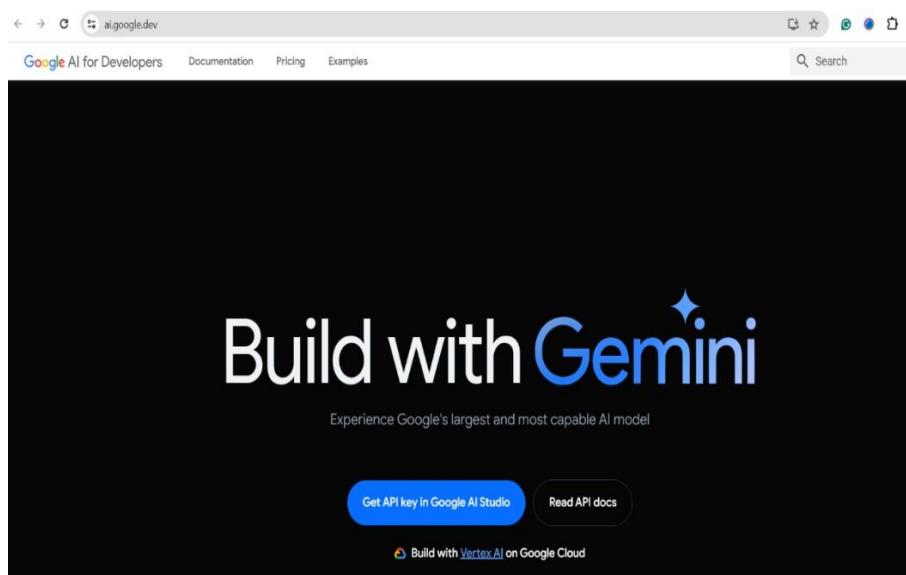
```
☰ requirements.txt
1 streamlit
2 google.generativeai
3
```

- Created requirements.txt

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

- Installed required libraries

10.2 Milestone 2 – Model Initialization



The screenshot shows the Google AI Studio interface. On the left, there's a sidebar with options like 'Get API key', 'Create new', 'My library', 'Allow Drive access', 'Getting started', and a help icon. The main area is titled 'Get API key' and contains a sub-section titled 'API keys'. It says you can create a new project or add API keys to an existing one. A 'Create API key' button is visible. The overall theme is dark.

```
src.py > ⚡ generate_itinerary
    import streamlit as st
    import google.generativeai as genai
```

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

```
# 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",
    }
```

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

- Generated Gemini API key
- Configured API using environment variable
- Initialized pre-trained model

10.3 Milestone 3 – Model Integration

```

# 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

```

- Created generate_itinerary() function
- Designed prompt template
- Sent input to model
- Retrieved response

10.4 Milestone 4 – Deployment

```

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

```

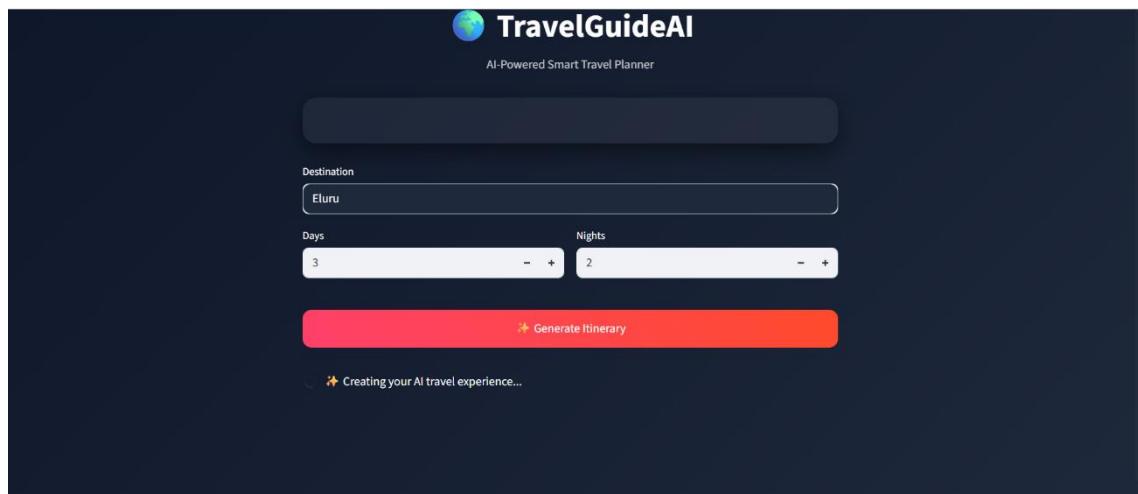
```

# 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.")

```

```
(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
```



Eluru – 3 Days & 2 Nights

Eluru, a city in the West Godavari district of Andhra Pradesh, offers a blend of natural beauty, historical sites, and spiritual experiences. Known for its proximity to Kolleru Lake (one of Asia's largest freshwater lakes) and its rich Buddhist heritage, Eluru provides a unique glimpse into the cultural tapestry of the region.

Here's a detailed 3-day and 2-night itinerary for exploring Eluru and its surroundings:

Eluru: 3-Day & 2-Night Itinerary

Best Time to Visit

The ideal time to visit Eluru is during the winter months, from October to March. The weather is pleasant and cool, making it perfect for sightseeing and birdwatching at Kolleru Lake. Summers (April to June) are extremely hot and humid, while the monsoon season (July to September)

Transportation Suggestions

- **To Eluru:**
 - **By Train:** Eluru is a major railway junction on the Howrah-Chennai main line. Many express trains stop here, connecting it to major cities like Hyderabad, Chennai, Kolkata, and Visakhapatnam.
 - **By Bus:** Eluru has a well-connected bus station with frequent services by APSRTC (Andhra Pradesh State Road Transport Corporation) and private operators to cities across Andhra Pradesh and Telangana.
 - **By Air:** The nearest airport is Gannavaram Airport (Vijayawada) (VGA), approximately 35-40 km away. From the airport, you can hire a taxi or take a bus to Eluru.
- **Within Eluru & Excursions:**
 - **Auto-rickshaws:** Readily available for short distances within the city. Negotiate fares beforehand.
 - **Local Buses:** APSRTC operates city buses, which are an economical option for getting around.
 - **Hired Car/Taxi:** For day trips to Kolleru Lake, Guntupalli Caves, and Dwaraka Tirumala, it is highly recommended to hire a private car with a driver. You can arrange this through your hotel or local taxi stands. This offers flexibility and comfort.

Food Recommendations

Andhra cuisine is known for its spicy and flavourful dishes. Eluru offers a good taste of authentic Andhra food.

- **Andhra Thali:** A must-try! It typically includes rice, sambar, rasam, a variety of curries (vegetable and non-vegetarian), dal (pappu), pachadi (pickle, especially Gongura Pachadi), curd, and sometimes a sweet.
- **Gongura Pachadi:** A spicy and tangy chutney made from sorrel leaves, a staple in Andhra households.
- **Pulusu:** A sour and tangy gravy made with tamarind, vegetables, or fish.
- **Biryani:** Chicken or Mutton Biryani prepared in the spicy Andhra style.
- **Seafood:** Given its proximity to Kolleru Lake and the coast, you might find fresh fish and prawn preparations.
- **Local Tiffins:** For breakfast or snacks, try Idli, Dosa, Vada, Upma, and Puri with various chutneys and sambar.
- **Sweets:** While not a local specialty of Eluru, you can find popular Andhra sweets like Pootharekulu (paper-thin sweet from Atreyapuram) and Kaja.
- **Recommended Eateries:** Look for local restaurants in the city center. Ask locals for recommendations for authentic taste. Many mid-range hotels also have good multi-cuisine restaurants.

Day-wise Breakdown

Day 1: Arrival, Buddhist Heritage & Local Charm

- **Morning (Arrival & Check-in):**
 - Arrive in Eluru by train, bus, or taxi from Vijayawada Airport.
 - Check into your chosen hotel. Freshen up and relax.
 - **Lunch:** Head to a local restaurant for an authentic Andhra Thali to kickstart your culinary journey.
- **Afternoon (Guntupalli Buddhist Caves):**
 - Post-lunch, embark on a trip to the **Guntupalli Buddhist Caves and Monasteries** (approx. 40-50 km from Eluru). These are ancient rock-cut caves dating back to the 3rd-2nd century BC, significant for their stupas, chaityas, and viharas. It's a serene and historically rich site offering beautiful views.
 - **Travel Tip:** Hire a private car for this trip as public transport might be infrequent.
- **Evening (Eluru City Exploration):**
 - Return to Eluru. Visit the **Jaladurga Temple**, a prominent local goddess temple known for its spiritual significance.
 - Explore the **Eluru Local Market** to soak in the city's vibe, observe local life, and perhaps pick up some regional handicrafts or snacks. Eluru is famous for its hand-woven carpets (Durries), though the production centers might be outside the main market.

- Built Streamlit UI
- Displayed itinerary
- Deployed locally using:

```
streamlit run travel.py
```

11. USE CASE SCENARIOS

11.1 Individual Travelers

Users can quickly generate personalized travel plans without manual research.

11.2 Travel Agencies

Agencies can automate itinerary creation, improving efficiency and productivity.

11.3 Travel Blogs

AI can generate structured travel content dynamically.

12. FEATURES

- AI-powered itinerary generation
- Personalized travel recommendations
- Day-wise structured plan
- Interactive web interface
- Secure API integration

13. FUTURE ENHANCEMENTS

- Budget-based recommendations
- Hotel and flight suggestions
- Map integration
- Weather-based planning
- PDF export option
- User login system

14. ADVANTAGES

- Saves time
- Reduces manual effort
- Provides customized travel plans
- Easy to use
- Scalable system

15. LIMITATIONS

- Requires internet connection
- Depends on API availability
- AI-generated results may require minor adjustments

16. CONCLUSION

Explore with AI successfully demonstrates the integration of Generative AI into a real-world web application. The system automates travel itinerary generation, making trip planning easier, faster, and more efficient. By combining Gemini AI with Streamlit, the project provides an intelligent solution that can benefit individual travelers, travel agencies, and online travel platforms.