1. INTRODUCTION

1.1 Project Overview

The AI Travel Itinerary Generator is an intelligent web-based application that automatically creates personalized travel plans using Artificial Intelligence. The system allows users to enter a destination, number of travel days, and number of nights. Based on this input, the application generates a detailed day-by-day itinerary including morning, afternoon, evening, and night activities, along with travel tips, food recommendations, and hotel suggestions.

The application is built using Streamlit for the user interface and integrates Google Generative AI (Gemini) to generate human-like travel content. Additionally, the system converts the generated itinerary into a professionally formatted PDF document, including a destination-based cover image, making it easy for users to download and use during their trip.

This project demonstrates how AI can simplify complex planning tasks and enhance user experience in the travel and tourism domain.

1.2 Purpose

The main purpose of this project is to:
· Reduce the time and effort required for manual travel planning
· Provide personalized travel itineraries based on user input
· Use AI to generate structured and meaningful travel content
· Offer a downloadable PDF itinerary for convenience
· Demonstrate the practical application of Generative AI in realworld problem solving

This system benefits individual travelers by acting as a smart virtual travel assistant.

2. LITERATURE SURVEY

S.No	Title / System	Author / Source	Method Used	Limitations	Relevance to Our Project
1	AI-Based Travel Recommendation System	Research Papers on Recommendation Systems	Machine Learning algorithms for suggesting destinations	Focuses only on destination recommendation	Generates complete day-wise itinerary
2	Travel Planning Websites	Online Travel Platforms	Rule-based recommendations and user reviews	No fully personalized AI-generated itinerary	Uses Generative AI for customized content
3	Chatbot-Based Travel Assistants	AI Chatbot Research	NLP-based conversational systems	Requires multiple interactions	Generates structured itinerary in one request
4	Generative AI Content Systems	Research on Large Language Models	LLMs for text generation	General-purpose content, not structured	Applies strict structured travel format
5	PDF Report Automation Systems	Python Automation Tools	Automated document generation using FPDF	Mostly used for static reports	Creates dynamic travel itinerary PDF

From the above study, we observed that:

- Traditional systems focus only on destination recommendations.

- Many travel platforms provide generic information but not personalized day-wise plans.

- Chatbot systems require continuous interaction to build an itinerary.

- Generative AI models can create high-quality content but require proper prompt structuring.

- Automated PDF generation improves usability and professional presentation.

Therefore, our project combines:

- Generative AI (for smart content generation)

- Structured formatting (for clarity)

- Web interface (for accessibility)

- Automated PDF generation (for convenience)

- This integration makes the system more practical and user-friendly compared to existing approaches.

2.1 Existing Problem

- Traditional travel planning involves:

- Searching multiple websites for places to visit

- Manually organizing activities day by day

- Finding hotels, food options, and travel tips separately

- Spending hours comparing information

- This process is time-consuming, confusing, and often overwhelming, especially for new travelers. Many existing travel websites provide generic recommendations but do not offer fully customized, AI-generated day-wise itineraries in a single place.
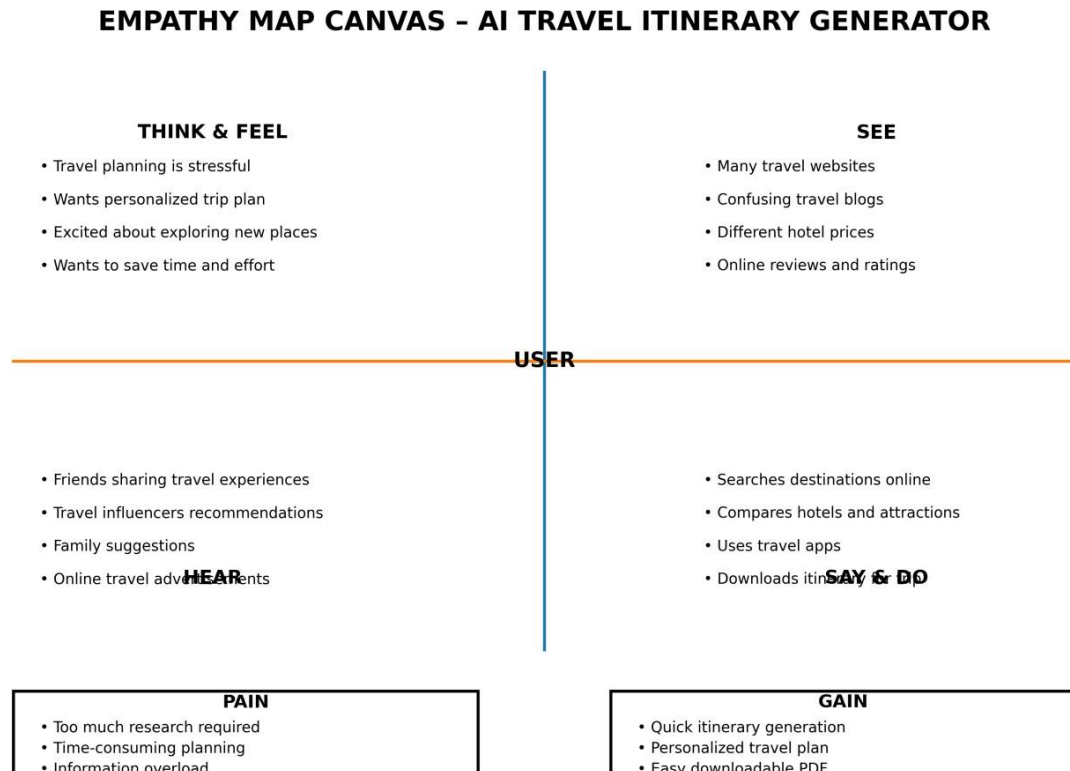
2.2 References

- The project is based on technologies and concepts from:

- Streamlit – For building interactive web applications

- Google Generative AI (Gemini API) – For generating travel itineraries using natural language

- FPDF Library – For creating downloadable PDF documents

- Unsplash API – For fetching destination images dynamically

- Research on AI-based recommendation systems and Generative AI for content creation

2.3 Problem Statement Definition

Travel planning is often complex and time-consuming due to scattered information and lack of personalization. There is a need for an intelligent system that can automatically generate structured, customized travel itineraries based on user preferences, reducing manual effort and improving travel planning efficiency.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

EMPATHY MAP CANVAS – AI TRAVEL ITINERARY GENERATOR

THINK & FEEL
- Travel planning is stressful
- Wants personalized trip plan
- Excited about exploring new places
- Wants to save time and effort

SEE
- Many travel websites
- Confusing travel blogs
- Different hotel prices
- Online reviews and ratings

USER

HEAR
- Friends sharing travel experiences
- Travel influencers recommendations
- Family suggestions
- Online travel advertisements

SAY & DO
- Searches destinations online
- Compares hotels and attractions
- Uses travel apps
- Downloads itinerary as PDF

PAIN
- Too much research required
- Time-consuming planning
- Information overload

GAIN
- Quick itinerary generation
- Personalized travel plan
- Easy downloadable PDF

3.2 Ideation & Brainstorming

Different ideas were explored:

- Static travel guide website ✗(Not personalized)

- Manual itinerary templates ✗(Still requires user effort)

- AI-based chatbot for travel ✓

- Automated itinerary generator with PDF export ✓

- The final idea chosen was a Generative AI-based itinerary generator that produces structured travel plans instantly.

3.3 Proposed Solution

The proposed solution is a web-based AI Travel Planner that:

1. Accepts destination, days, and nights as input

2. Uses Gemini AI to generate a detailed itinerary

3. Structures content into:

- Morning

- Afternoon

- Evening

- Night

4. Adds:

- Travel tips

- Food recommendations

- Hotel suggestions

4. Converts the result into a downloadable PDF with a cover page

3.4 Problem Solution Fit

PROBLEM – SOLUTION FIT CANVAS
AI TRAVEL ITINERARY GENERATOR

WHO IS YOUR CUSTOMER?

Individual travelers
Students
Families planning trips
Busy professionals

LIMITATIONS TO BUY / USE

Lack of time for planning
Too much information online
Difficulty organizing trip
Budget confusion

HOW ARE YOU DIFFERENT?

AI-generated itinerary
Personalized planning
Day-wise structured plan
Instant PDF download

FREQUENT / COSTLY PROBLEM

Manual travel planning takes hours
Searching many websites
No clear daily schedule
Unorganized trip experience

CAUSE OF PROBLEM

Information scattered online
No automation tools
Lack of personalization
Manual planning required

SUPPORT EXISTING BEHAVIOR

Users search destinations online
Compare hotels & attractions
Use travel apps
Look for reviews

REAL LIFE TRIGGERS

Vacation planning
Holiday season trips
Family travel decisions

YOUR DOWN-TO-EARTH SOLUTION

AI Travel Itinerary Generator

Enter destination & duration
AI creates full travel plan
Provides tips, food & hotels
Download ready-to-use PDF

BE WHERE CUSTOMERS ARE

Web-based application
Accessible from anywhere
Simple user interface
Quick results

EMOTIONS

Excitement for travel
Stress about planning
Need convenience

4 .REQUIREMENT ANALYSIS

4.1 Functional Requirements

- The system must:

- Allow user to enter destination, number of days, and nights

- Generate itinerary using AI

- Display itinerary on screen

- Provide travel tips, food, and hotel suggestions

- Create and allow download of itinerary in PDF format

- Validate user inputs (days, nights, etc.)

4.2 Non-Functional Requirements

- The system should be user-friendly

- It should generate results within a few seconds

- The interface should be responsive

- The system should handle invalid inputs gracefully

- The generated PDF should be properly formatted

5. PROJECT DESIGN

5.1 Data Flow Diagrams (DFD)

DFD Level 1 – Detailed System Flow
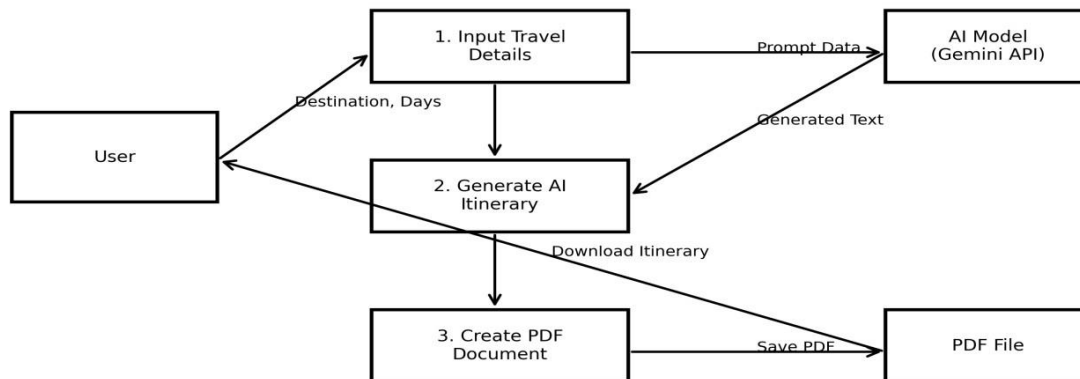
**DFD LEVEL 2 – AI TRAVEL ITINERARY GENERATOR
DETAILED INTERNAL PROCESSING**
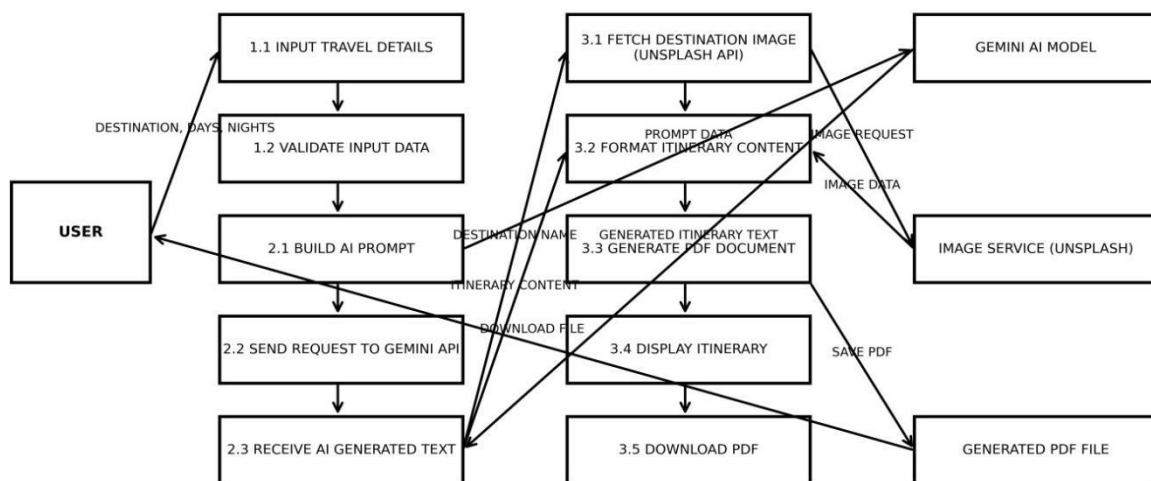
5.2 Solution & Technical Architecture

- The proposed system is a web-based AI travel planning application that automatically generates personalized travel itineraries based on user input. The architecture follows a simple layered structure consisting of the user interface, application logic, AI service integration, and output generation.

Technical Components

- Frontend (User Interface):
  Developed using Streamlit to collect user inputs such as destination, number of days, and nights, and display the generated itinerary.

- Application Logic (Backend):

  Implemented in Python to handle input validation, prompt creation, AI communication, text processing, and PDF generation.

- AI Processing Layer:
  The system integrates with the Google Gemini Generative AI model to generate structured day-wise travel itineraries based on user requirements.

- External Services:

- Unsplash API is used to fetch destination images.

- FPDF library is used to generate downloadable itinerary PDFs.

- Output Layer:

The generated itinerary is displayed on the web interface and provided as a downloadable PDF file.

Technical Architecture:

AI TRAVEL ITINERARY GENERATOR

```
                 ┌─────────────────────────────┐
                 │      USER DEVICE            │
                 │  (Web Browser / Mobile)     │
                 └─────────────────────────────┘
                               │
                               ▼
                 ┌─────────────────────────────┐
                 │     PRESENTATION LAYER       │
                 │   Streamlit Web Interface    │
                 └─────────────────────────────┘
```

APPLICATION LOGIC LAYER

Input Validation	Prompt Generation	Itnerary Processing & Formatting

```
                 ┌─────────────────────────────┐
                 │   GENERATIVE AI SERVICE      │
                 │    Google Gemini Model       │
                 └─────────────────────────────┘
```

EXTERNAL SERVICES

Image Service	PDF Generation
Unsplash API	FPDF Library

```
                 ┌─────────────────────────────┐
                 │       OUTPUT LAYER           │
                 │ Display Itinerary & Download PDF │
                 └─────────────────────────────┘
```
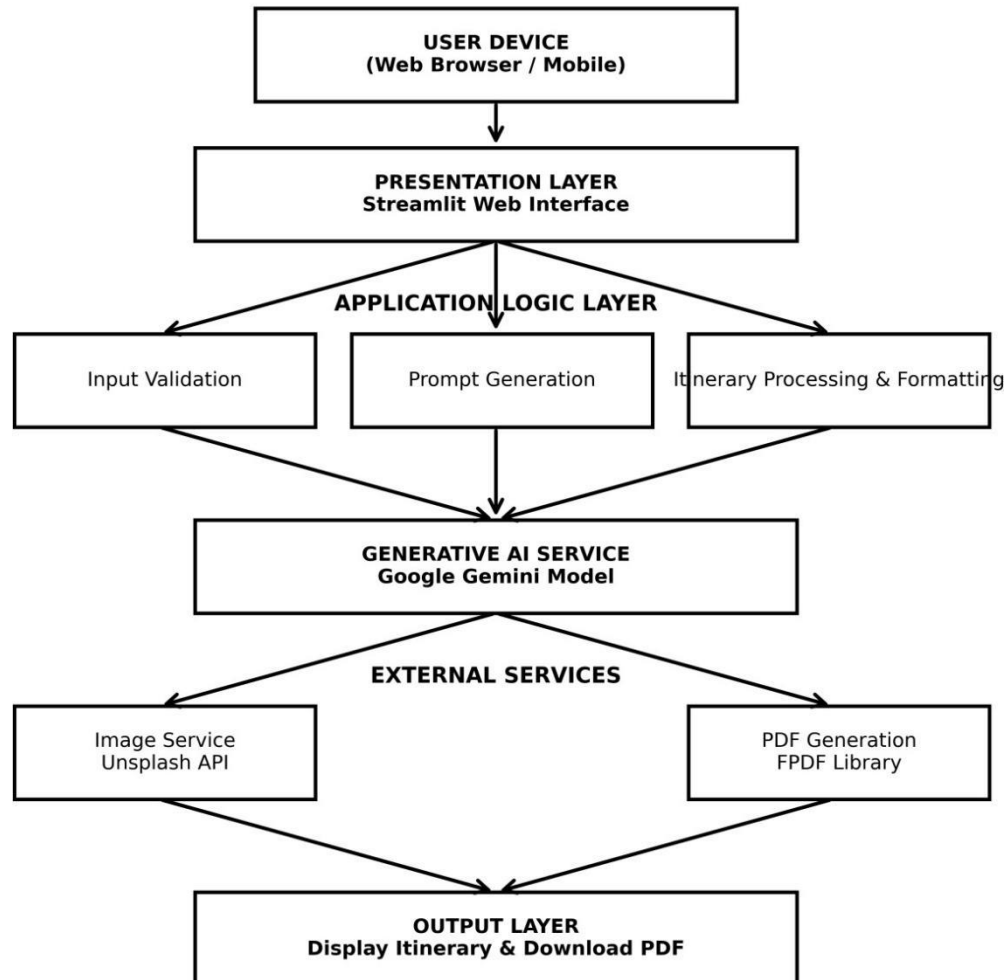
5.3 User Stories

User Type	Functional Requirement	User Story Number	User Story / Task	Acceptance Criteria	Priority	Release
Traveler	Enter travel details	US-1	Enter destination, days, nights to generate plan	Valid input accepted	High	Sprint-1
Traveler	Validate input	US-2	System validates user input	Error shown for invalid data	High	Sprint-1
Traveler	Generate itinerary	US-3	Generate AI-based day-wise itinerary	Structured itinerary generated	High	Sprint-1
Traveler	View itinerary	US-4	View generated travel plan	Itinerary displayed clearly	High	Sprint-1
Traveler	Travel recommendations	US-5	Get food, hotel and travel tips	Recommendations included	Medium	Sprint-1
Traveler	Download PDF	US-6	Download itinerary as PDF	PDF generated and downloaded	High	Sprint-2
Traveler	Destination image	US-7	Include destination image in PDF	Image shown in PDF	Medium	Sprint-2
System Admin	AI integration	US-8	Connect with Gemini AI	AI response received	High	Sprint-1
System Admin	Error handling	US-9	Handle system errors	Error messages shown	Medium	Sprint-2
System Admin	Performance	US-10	Generate itinerary quickly	Fast response time	Medium	Sprint-2

6. PROJECT PLANNING & SCHEDULING

The project development followed an Agile-based sprint methodology to ensure systematic implementation and continuous progress. The complete project was divided into multiple sprints, each focusing on specific functional components such as requirement analysis, AI model integration, web application development, and deployment.

Each sprint included clearly defined tasks, priorities, timelines, and expected deliverables. This structured planning helped in efficient resource management, timely completion of milestones, and continuous testing of system functionality.

The detailed sprint schedule, product backlog, and sprint-wise execution plan are presented below.

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Team Members	Sprint Start Date	Sprint End Date (Planned)
Sprint-1	Requirement Analysis & Setup	AI-1	Understanding project scope and features	High	Gandi Rajesh	29/01/2026	04/02/2026
Sprint-1	Requirement Analysis & Setup	AI-2	Setting up development environment	High	Gandi Rajesh	29/01/2026	04/02/2026
Sprint-1	Requirement Analysis & Setup	AI-3	Researching AI itinerary generation approach	Medium	Gandi Rajesh	29/01/2026	04/02/2026
Sprint-2	AI Model Integration	AI-4	Connecting Generative AI API	High	Gandi Rajesh	05/02/2026	10/02/2026
Sprint-2	AI Model Integration	AI-5	Designing prompt structure for itinerary	High	Gandi Rajesh	05/02/2026	10/02/2026
Sprint-2	AI Model Integration	AI-6	Testing AI-generated responses	Medium	Gandi Rajesh	05/02/2026	10/02/2026

Sprint-3	Web Application Development	AI-7	Designing Streamlit UI	High	Gandi Rajesh	11/02/2026	15/02/2026
Sprint-3	Web Application Development	AI-8	Taking user inputs (destination, days, nights)	High	Gandi Rajesh	11/02/2026	15/02/2026
Sprint-3	Web Application Development	AI-9	Displaying generated itinerary	Medium	Gandi Rajesh	11/02/2026	15/02/2026
Sprint-4	Additional Features & Deployment	AI-10	Generating downloadable PDF itinerary	Medium	Gandi Rajesh	16/02/2026	19/02/2026
Sprint-4	Additional Features & Deployment	AI-11	Adding travel images via API	Low	Gandi Rajesh	16/02/2026	19/02/2026
Sprint-4	Additional Features & Deployment	AI-12	Final testing and bug fixing	High	Gandi Rajesh	16/02/2026	19/02/2026

Sprint Explanation

☐ Sprint 1 – Requirement Analysis & Setup

In this sprint, the project idea was finalized and the required technologies were identified. The development environment was set up, including installing Python, Streamlit, and required libraries. Research was conducted on how generative AI can be used to create travel itineraries

Screenshot:

☐ **Sprint 2 – AI Model Integration**

This sprint focused on integrating the Generative AI model into the project. The API key was configured, prompts were designed to generate structured itineraries, and responses were tested for accuracy and formatting.

Screenshot:
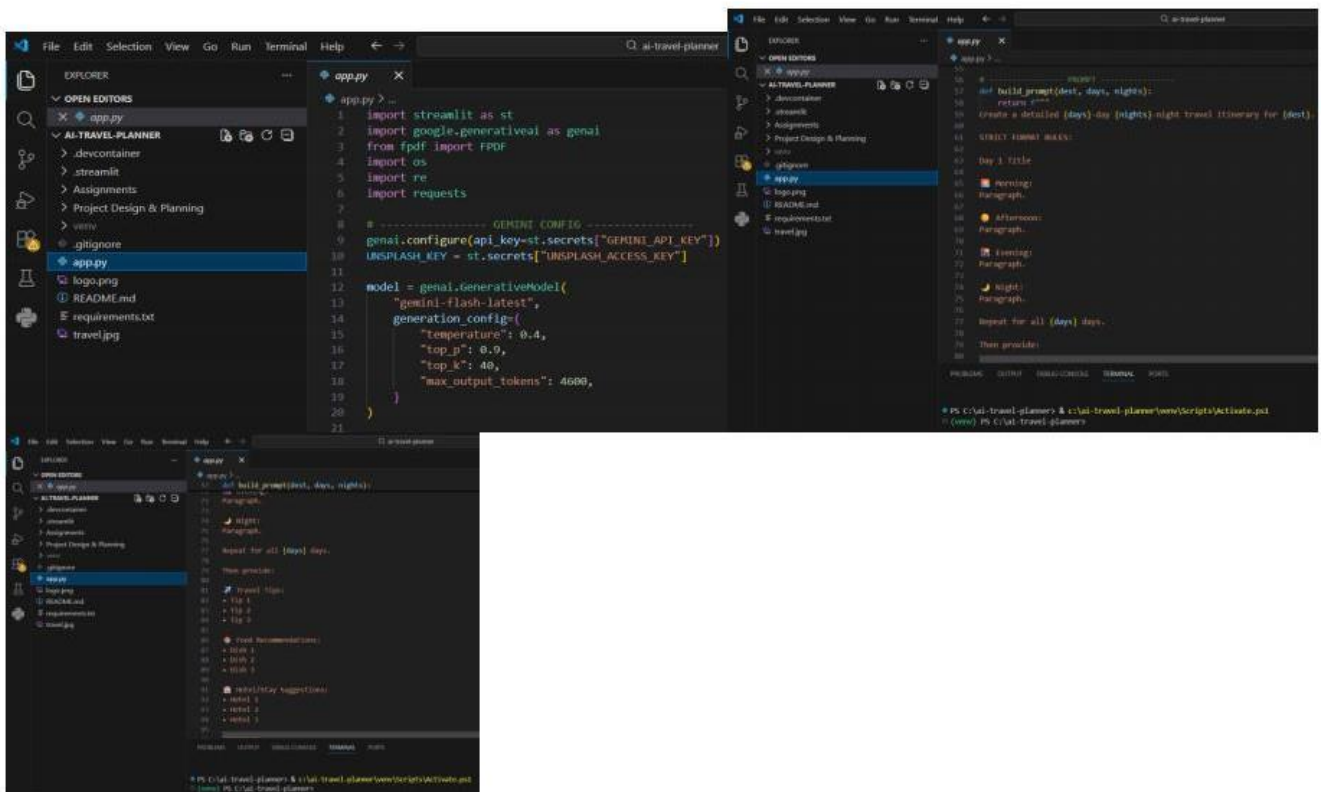
□ **Sprint 3 – Web Application Development**

During this sprint, the Streamlit-based user interface was developed. Input fields for destination, days, and nights were added, and the AI-generated itinerary was displayed dynamically on the webpage.

Screenshot:
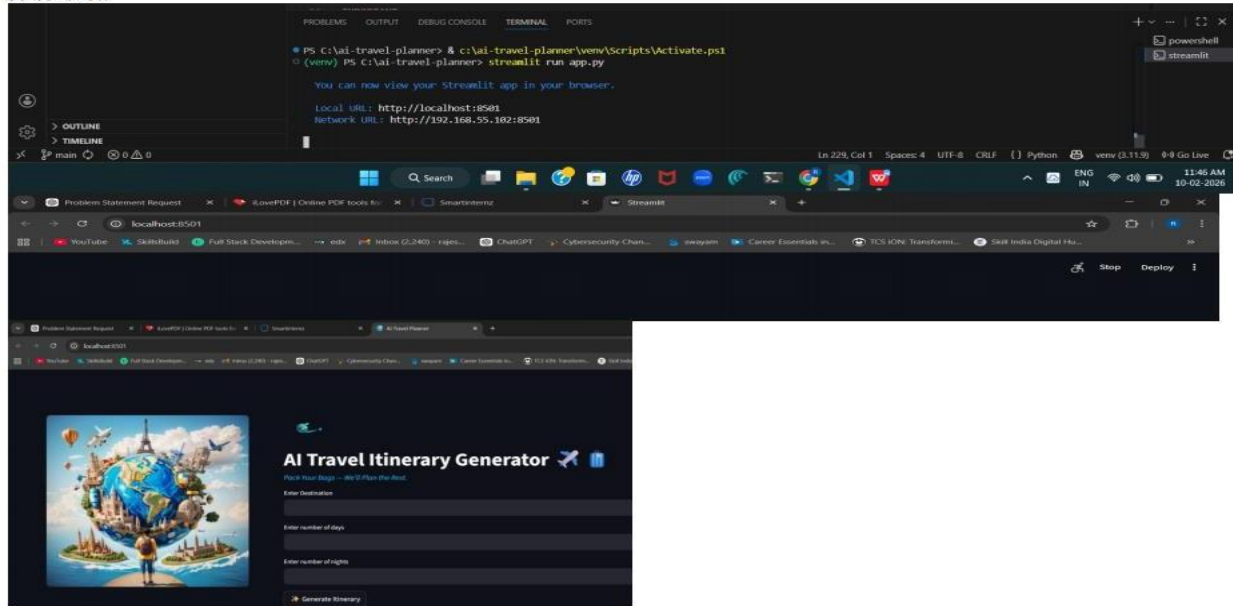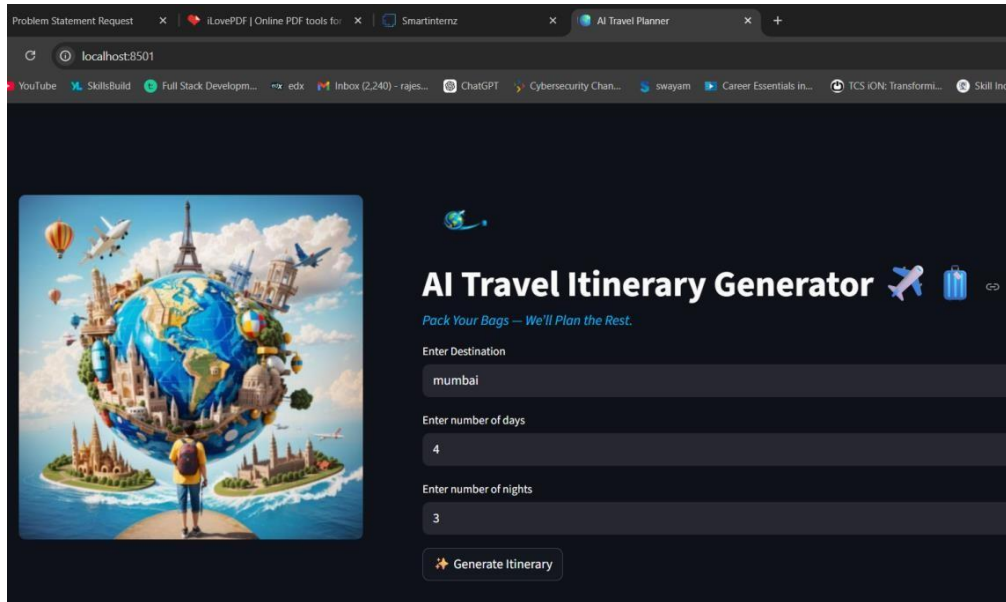
□ **Sprint 4 – Additional Features & Deployment**

This sprint included generating a downloadable PDF of the itinerary, adding travel images using an API, and performing final testing. The application was refined for better user experience and stability.

Screenshot:
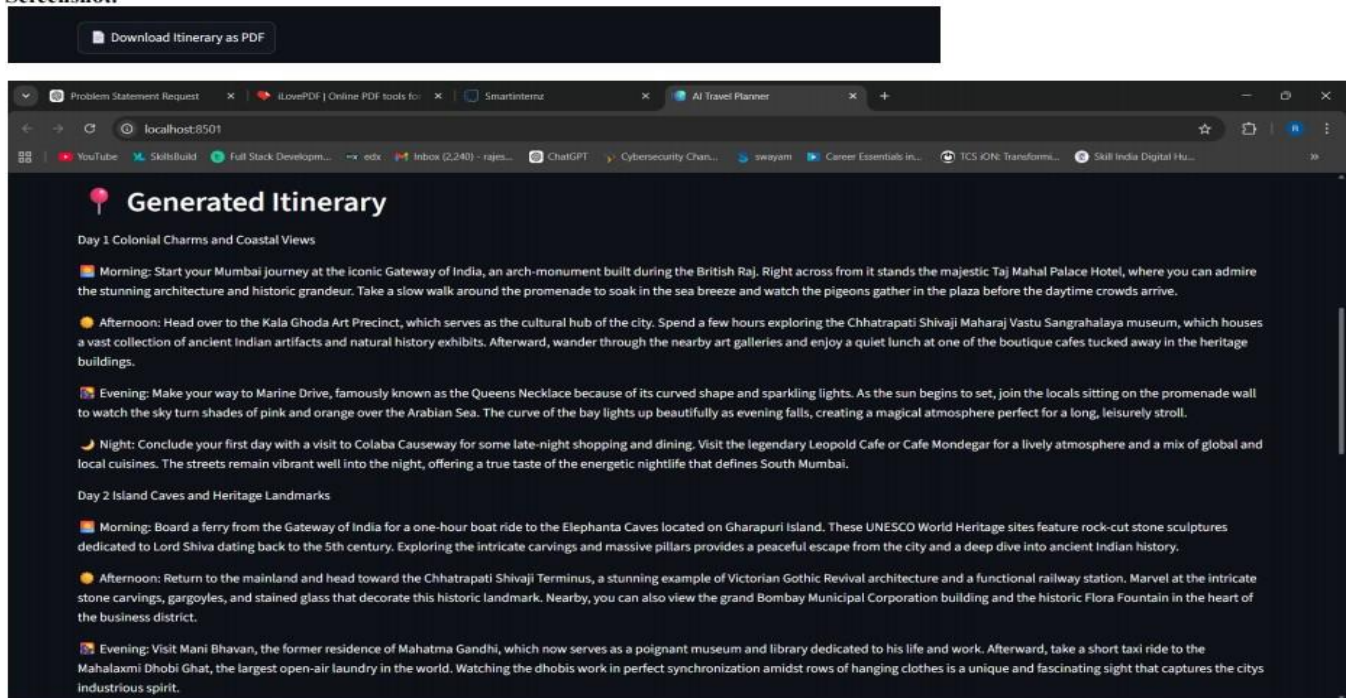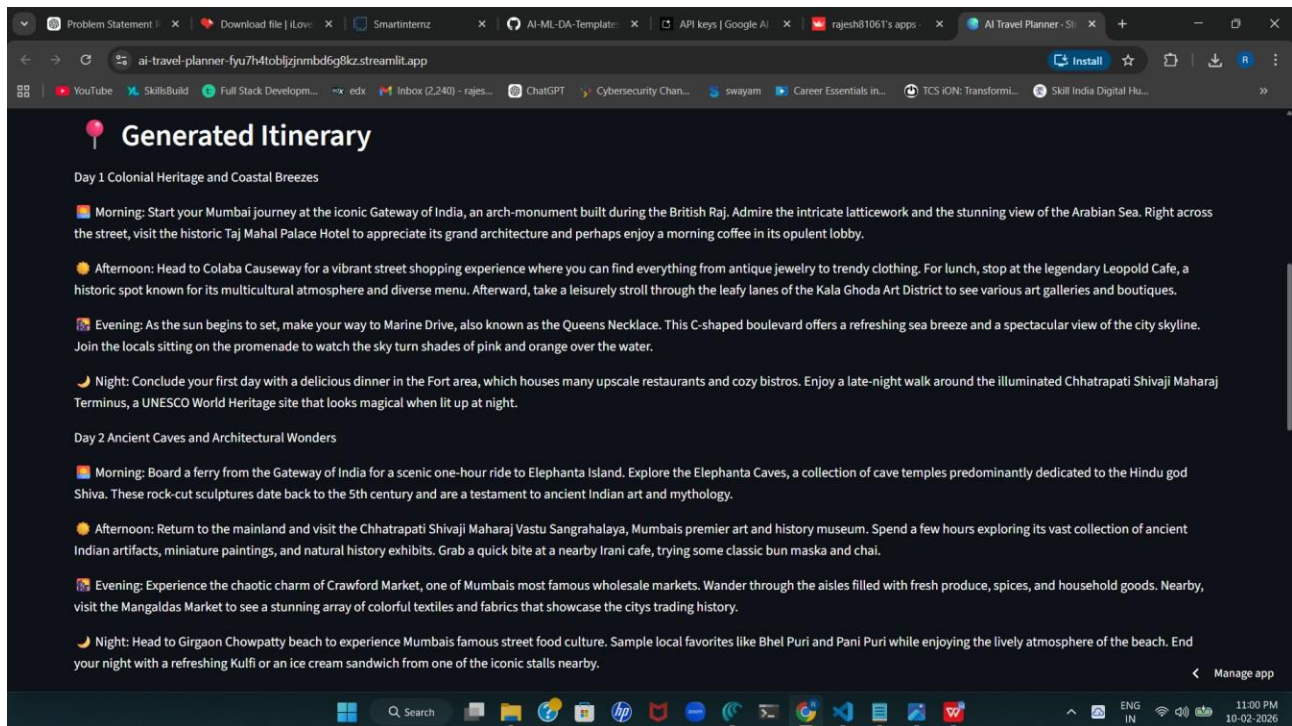
RESULTS (OUTPUT SCREENSHOTS):

Day 3 Suburban Glamour and Seaside Serenity

🏙️ Morning: Travel north to the trendy suburb of Bandra, often called the Queen of the Suburbs. Walk along Bandstand to see the houses of famous Bollywood stars, including Shah Rukh Khans mansion, Mannat. Visit the historic Mount Mary Basilica, a beautiful Roman Catholic church perched on a hill overlooking the sea.

☀️ Afternoon: Explore the vibrant street art in the villages of Bandra, such as Chuim or Ranwar, which feature colorful murals and colonial-style houses. For lunch, choose one of the many chic cafes in Pali Hill. Afterward, visit the Bandra Fort ruins for a panoramic view of the Bandra-Worli Sea Link and the coastline.

🌆 Evening: Take a taxi ride across the Bandra-Worli Sea Link, an architectural marvel that connects the suburbs to South Mumbai. The drive offers breathtaking views of the ocean and the citys skyline. Head toward Juhu Beach to catch the sunset and watch the shore come alive with families and food vendors.

🌙 Night: Enjoy a sophisticated dinner at one of the high-end restaurants in Juhu or Vile Parle. This area is known for its nightlife and celebrity sightings. If you have energy left, visit a rooftop bar to enjoy the city lights and a cocktail before heading back to your accommodation.

Day 4 Local Traditions and Spiritual Sites

🏙️ Morning: Start your final day at Mahalaxmi Dhobi Ghat, the worlds largest open-air laundry. Watch the dhobis at work in a fascinating display of organized chaos that has existed for over a century. Afterward, visit the nearby Mahalaxmi Temple, dedicated to the goddess of wealth, to experience the spiritual side of the city.

☀️ Afternoon: Walk along the causeway to reach the Haji Ali Dargah, a stunning mosque and tomb located on an islet off the coast. Ensure you check the tide timings, as the path is only accessible during low tide. Later, visit Mani Bhavan, the former residence of Mahatma Gandhi, which now serves as a museum dedicated to his life and work.

🌆 Evening: Spend your final evening doing some last-minute shopping at High Street Phoenix or Palladium Mall in Lower Parel. This area, once dominated by textile mills, is now a hub for luxury brands and modern entertainment. It is a great place to pick up souvenirs or enjoy a final cup of Indian coffee.

🌙 Night: Have a farewell dinner featuring traditional Maharashtrian cuisine to wrap up your trip. Depending on your flight schedule, head toward the Chhatrapati Shivaji Maharaj International Airport. Reflect on the vibrant energy and diverse experiences of the City of Dreams as you prepare for your departure.

✈️ Travel Tips: • Use the local train network to travel long distances but strictly avoid peak rush hours in the morning and evening to avoid extreme crowds. • Download ride-sharing apps like Uber or Ola for reliable transport, or use the iconic black and yellow Kaali-Peeli taxis for shorter distances. • Carry a reusable water bottle and stay hydrated, as the humidity in Mumbai can be quite draining even during the cooler winter months.

🍛 Food Recommendations: • Vada Pav which is the ultimate Mumbai street food consisting of a spicy potato fritter inside a soft bread bun with chutneys. • Pav Bhaji which is a thick vegetable curry served with soft buttered bread rolls and a dash of fresh lemon and onions. • Misal Pav which is a spicy sprout-based curry topped with crunchy farsan and served with bread for a flavorful breakfast or snack.

🏨 Hotel/Stay Suggestions: • The Taj Mahal Palace for a luxury heritage experience with world-class service right next to the Gateway of India. • Abode Bombay for a boutique and stylish stay featuring vintage decor in the heart of the historic Colaba district. • The Sahil Hotel for a comfortable mid-range option with excellent connectivity to major tourist sites and the main railway stations.

📄 Download Itinerary as PDF

Conclusion

The AI Travel Itinerary Generator successfully demonstrates how Generative Artificial Intelligence can simplify and automate travel planning. The system allows users to enter basic travel details such as destination, number of days, and nights, and generates a structured, personalized itinerary that includes daily activities, travel tips, food recommendations, and hotel suggestions.

By integrating Google Gemini AI, Streamlit, and PDF generation, the application provides an intelligent and user-friendly solution that reduces manual effort and saves time in trip planning. The system also enhances usability by offering a downloadable PDF itinerary with destination images, making it convenient for travelers to use during their journey.

Overall, the project proves that AI-driven automation can significantly improve decision-making and planning processes in the travel domain. It provides an efficient, scalable, and practical solution for personalized travel assistance.

Future Scope

- The AI Travel Itinerary Generator can be further enhanced with several advanced features to improve functionality and user experience:

- User Preference Personalization: Allow users to select budget range, travel style, interests, and transportation preferences.

- Real-Time Data Integration: Include live weather updates, traffic information, and event recommendations.

- Booking Integration: Enable direct hotel, flight, and activity booking from the application.

- Multi-language Support: Provide itinerary generation in multiple languages for global users.

- Mobile Application Development: Develop Android or iOS versions for better accessibility.

- Voice-Based Interaction: Allow users to plan trips using voice commands.

- Collaborative Trip Planning: Enable multiple users to plan and edit a trip together.

- Machine Learning Recommendations: Improve suggestions based on past travel history and user feedback.

With these enhancements, the system can evolve into a complete intelligent travel assistant capable of providing fully automated and smart travel management.

Appendix
source code(app.py)

```python
import streamlit as st
import google.generativeai as genai
from fpdf import FPDF
import os
import re
import requests

# ------------------------- GEMINI CONFIG ----------------
genai.configure(api_key=st.secrets["GEMINI_API_KEY"])
UNSPLASH_KEY = st.secrets["UNSPLASH_ACCESS_KEY"]

model = genai.GenerativeModel(
    "gemini-flash-latest",
    generation_config={
        "temperature": 0.4,
        "top_p": 0.9,
        "top_k": 40,
        "max_output_tokens": 4600,
    }
)

st.set_page_config(page_title="AI Travel Planner", page_icon="", layout="wide")

#  HIDE STREAMLIT HEADER (Removes GitHub + Fork buttons)
st.markdown("""
    <style>
        header {visibility: hidden;}
        footer {visibility: hidden;}
    </style>
""", unsafe_allow_html=True)

# ------------------------- MAIN LAYOUT -------------------------
left_col, right_col = st.columns([1.1, 2])

with left_col:
    st.image("travel.jpg", width=420)

with right_col:
    st.image("logo.png", width=100)

    st.markdown(
        "<h1 style='margin:0;padding:0;'>AI Travel Itinerary Generator </h1>",
        unsafe_allow_html=True
    )

    st.markdown(
        "<p style='margin-top:4px;margin-bottom:12px;font-size:17px;color:#00BFFF;font-style:italic;'>Pack Your Bags — We'll Plan the Rest.</p>",
        unsafe_allow_html=True
    )

    destination = st.text_input("Enter Destination")
    days_input = st.text_input("Enter number of days")
    nights_input = st.text_input("Enter number of nights")
    generate_btn = st.button("Generate Itinerary")

# ------------------------- PROMPT -------------------------
def build_prompt(dest, days, nights):
```

```
    return f"""
Create a detailed {days}-day {nights}-night travel itinerary for {dest}.

STRICT FORMAT RULES:

Day 1 Title

🌅 Morning:
Paragraph.

🌞 Afternoon:
Paragraph.

🌆 Evening:
Paragraph.

🌙 Night:
Paragraph.

Repeat for all {days} days.

Then provide:

💡 Travel Tips:
• Tip 1
• Tip 2
• Tip 3

🍽 Food Recommendations:
• Dish 1
• Dish 2
• Dish 3

🏨 Hotel/Stay Suggestions:
• Hotel 1
• Hotel 2
• Hotel 3

IMPORTANT:
No markdown symbols like * or #
No tables
Use emojis exactly as shown
"""

# ------------------------- TEXT CLEANING ----------------
def clean_text(text):
    text = re.sub(r"[#*]", "", text)
    replacements = {"–": "-", "—": "-", "'": "'", """: '"', """: '"', "•": "-"}
    for k, v in replacements.items():
        text = text.replace(k, v)
    return text.encode("latin-1", "ignore").decode("latin-1").strip()

# ------------------------- CUSTOM PDF CLASS -------------------------
class PDF(FPDF):
    def footer(self):
        self.set_y(-10)
        self.set_font("Arial", size=9)
        self.cell(0, 5, f"Page {self.page_no()}", align="R")

# ------------------------- UNSPLASH IMAGE FETCH ----------------
def get_unsplash_image(dest):
```

```python
        try:
            url =
f"https://api.unsplash.com/photos/random?query={dest}&orientation=landscape&client_id={UNSPLAS
H_KEY}"
            res = requests.get(url, timeout=10)
            if res.status_code == 200:
                img_url = res.json()["urls"]["regular"]
                img_data = requests.get(img_url, timeout=10).content
                path = "/tmp/cover.jpg"
                with open(path, "wb") as f:
                    f.write(img_data)
                return path
            else:
                return None
        except:
            return None


# ------------------------- PDF CREATION ----------------
def create_pdf(text, destination):
    text = clean_text(text)
    pdf = PDF()
    pdf.set_auto_page_break(auto=True, margin=12)

    pdf.add_page()
    img_path = get_unsplash_image(destination)

    if img_path and os.path.exists(img_path):
        pdf.image(img_path, x=0, y=0, w=210, h=297)
        os.remove(img_path)
        pdf.set_text_color(255, 255, 255)
    else:
        pdf.set_fill_color(20, 40, 70)
        pdf.rect(0, 0, 210, 297, 'F')
        pdf.set_text_color(255, 255, 255)

    pdf.set_font("Arial", "B", 26)
    pdf.ln(120)
    pdf.cell(0, 15, f"{destination.upper()} TRAVEL ITINERARY", align="C", ln=True)

    pdf.set_font("Arial", "", 14)
    pdf.cell(0, 10, "Smart Planning. Elevated Travel.", align="C", ln=True)

    pdf.add_page()
    pdf.set_text_color(0, 0, 0)

    pdf.set_font("Arial", "B", 18)
    pdf.cell(0, 10, f"{destination} Travel Itinerary", ln=True, align="C")
    pdf.ln(4)

    pdf.set_font("Arial", size=11)

    for line in text.split("\n"):
        stripped = line.strip()

        if stripped.lower().startswith(("▯", "▯", "▯", "travel tips", "food recommendations",
"hotel/stay suggestions")):
            pdf.ln(5)
            pdf.set_font("Arial", "B", 14)
            pdf.multi_cell(0, 8, stripped)
            pdf.ln(2)
            pdf.set_font("Arial", size=11)
```

```python
        elif stripped.startswith("-"):
            pdf.multi_cell(0, 6, stripped)
            pdf.ln(1)

        elif stripped.lower().startswith("day"):
            pdf.ln(4)
            pdf.set_font("Arial", "B", 13)
            pdf.multi_cell(0, 7, stripped)
            pdf.ln(1)
            pdf.set_font("Arial", size=11)

        else:
            pdf.multi_cell(0, 6, stripped)

    pdf_path = "travel_itinerary.pdf"
    pdf.output(pdf_path)
    return pdf_path

# ------------------------- GENERATE ITINERARY ----------------
if generate_btn:
    if not destination or not days_input or not nights_input:
        st.warning("⚠ Please fill all the fields before generating itinerary.")
    elif not days_input.isdigit() or not nights_input.isdigit():
        st.warning("⚠ Days and Nights must be valid numbers.")
    elif int(days_input) > 15:
        st.warning("⚠ Maximum trip length allowed is 15 days.")
    elif int(nights_input) >= int(days_input):
        st.warning("⚠ Nights must be less than number of days.")
    else:
        days = int(days_input)
        nights = int(nights_input)

        with st.spinner("Generating your travel plan... ⏳"):
            prompt = build_prompt(destination, days, nights)
            response = model.generate_content(prompt)
            st.session_state.itinerary = response.text
            st.session_state.destination = destination

# ------------------------- DISPLAY RESULT -------------------------
if "itinerary" in st.session_state:
    st.markdown("## 📋 Generated Itinerary")
    st.markdown(st.session_state.itinerary)

    pdf_path = create_pdf(st.session_state.itinerary, st.session_state.destination)
    with open(pdf_path, "rb") as f:
        st.download_button(
            "📥 Download Itinerary as PDF",
            f,
            file_name="travel_itinerary.pdf",
            mime="application/pdf"
        )
```

Links

https://github.com/Rajesh81061/ai-travel-planner.git

https://github.com/Rajesh81061/ai-travel-planner/blob/main/Project%20Demonstration/Demo%20Video.mp4