**IBM HACKATHON PROJECT**

**TRAVEL PLANNER AI AGENT**

**Presented By:Soumya Roy**

**Name-Soumya Roy**

**Department-Computer Science And Engineering**

**College Name:Gurunanak Institute Of Technology**

**----------------------------------------------------------------------------------Technologies Whatever I Have Used To Create This Agentic AI Model:**

* **IBM Cloud Lite Services**
* **Reliable cloud infrastructure** for hosting and scaling AI applications.  
   Enables seamless API integration and scalable deployment of travel assistant services across regions. Automatically handles backend load, ensures high availability, and reduces infrastructure management overhead.
* **IBM Granite Model**
* **Advanced pre-trained large language model** optimized for enterprise-grade NLP.  
   Provides deep language understanding, allowing personalized recommendations, contextual dialogue, and accurate interpretation of user preferences in natural language interactions.

### **Natural Language Processing (NLP)**

* **Conversational AI layer** for interpreting and generating human-like language.  
   Allows users to interact naturally, asking complex travel questions like “What’s the best 3-day itinerary for Rome under $500?” while the AI understands intent, entities, and context.

### **Retrieval-Augmented Generation (RAG)**

* **Combines external data retrieval with generative responses** for up-to-date information.  
   Enables the travel planner to access real-time travel blogs, airline updates, and local tourism boards, enriching the assistant’s answers with fresh, relevant content beyond pre-trained knowledge.

### **Real-Time Data Integration**

* **Live syncing with APIs and data feeds** from airlines, weather services, transit systems, and local events.  
   Ensures travelers receive accurate, up-to-the-minute updates on flight delays, transit options, weather changes, or nearby experiences—vital for making informed travel decisions.

### **Multi-Channel Deployment**

* **Omni-channel support** for web apps, mobile devices, and voice assistants.  
   Ensures users can access travel planning services on the go—whether through a smartphone app, a website chatbot, or smart speakers like Alexa or Google Assistant—offering a consistent experience across platforms.

**IBM CLOUD SERVICES USED**

### **IBM Cloud Watsonx AI Studio**

**What it is:**  
 A collaborative development environment for building, training, validating, and deploying AI models.

**How it helps a Travel Planner AI:**

* Lets you **build custom machine learning or NLP models** using travel-related data (e.g., itineraries, weather, user preferences).
* Allows you to **fine-tune foundation models** like Granite on travel-specific use cases.
* Provides tools to label data (e.g., classify activities as "beach", "cultural", "adventure").

📌 *Example use:* Fine-tune a model to personalize itineraries for users based on their history and interests.

* **2. IBM Cloud Watsonx AI Runtime**

**What it is:**  
 The environment where AI models are deployed and served for inference (i.e., used to make predictions or generate content).

**How it helps a Travel Planner AI:**

* You can **deploy your trained models** (from AI Studio) to production.
* The Travel AI can now generate responses like “Best 3-day trip in Rome” or “What to pack for Japan in winter” by calling your deployed model.
* Supports RESTful APIs for easy integration with websites, apps, or chatbots.

📌 *Example use:* A user asks, “Plan a 7-day trip to Japan.” The model, deployed via Watsonx AI Runtime, generates a detailed itinerary.

### **3. IBM Cloud Agent Lab**

**What it is:**  
 An experimental or prototype environment for building and testing AI agents that use tools, reasoning, and dynamic workflows.

**How it helps a Travel Planner AI:**

* Create **agent workflows** that mimic how a human travel agent might operate.
* Agents can **reason**, use tools (like search or maps APIs), and call different models.
* You can **chain tasks**: find flights ➝ check visa requirements ➝ build itinerary ➝ book hotels.

📌 *Example use:* A Travel Planner agent can:

1. Understand a user's request.
2. Call a flight API.
3. Suggest hotels based on price and reviews.
4. Generate a final trip plan.

### **4 IBM Granite Foundation Model**

**What it is:**  
 A large language model (like ChatGPT) developed by IBM, trained on diverse enterprise data and optimized for enterprise tasks.

**How it helps a Travel Planner AI:**

* Acts as the **core language-understanding engine**.
* Can understand complex travel queries, generate fluent natural language, and summarize or explain options.
* Can be fine-tuned in AI Studio with travel-specific tone or preferences.

📌 *Example use:*  
 A user says: “I want a relaxing trip to Southeast Asia with some nature and culture but not too much walking.”  
 Granite helps interpret the vague request and generate a thoughtful plan based on the preferences.

**What it is:**  
 A large language model (like ChatGPT) developed by IBM, trained on diverse enterprise data and optimized for enterprise tasks.

**How it helps a Travel Planner AI:**

* Acts as the **core language-understanding engine**.
* Can understand complex travel queries, generate fluent natural language, and summarize or explain options.
* Can be fine-tuned in AI Studio with travel-specific tone or preferences.

📌 *Example use:*  
 A user says: “I want a relaxing trip to Southeast Asia with some nature and culture but not too much walking.”  
 Granite helps interpret the vague request and generate a thoughtful plan based on the preferences.

### **✅ Putting It All Together: Example Flow**

1. **User Input:** “Plan a romantic 5-day getaway in Italy with wine tasting and scenic views.”
2. **Watsonx Agent Lab:** Routes the request to the right tools (Granite model + external API for hotels/wineries).
3. **Granite Model (fine-tuned in AI Studio):** Understands the tone and travel style.
4. **Watsonx AI Runtime:** Generates the day-by-day itinerary.
5. **Agent Lab Workflow:** Adds links to book tours, check weather, etc.
6. **Output:** An interactive travel plan is returned to the user.