**AI Powered Travel Booking**

# Introduction

You are tasked with creating a travel booking application powered completely by AI agent. Its purpose is to help users plan and book their vacation in the state of Telangana.

Users will provide, by typing in a natural language, their overall interest (e.g. adventure vacation, religious tourism, forest stays, quiet getaways, children study tour etc.) and the application should suggest a vacation plan based on their budget, number of days, time of the year etc.

Following are the high level requirements:

* The application should have a chat interface where user can interact with the system by entering their requirements and the AI agent will then provide them recommendation for their vacation plan.
* **Optional**: If user wants to know more details about the place or hotel etc., it will show them pictures and videos.
* Given budget, and all other requirements, it will be able to recommend them a vacation plan.
* Once user is ready, it will be able to make all bookings for them.
* It will run on-prem and will not have GPU so a quantized model (like Mistral) should be used on commodity hardware.

# Learning Path

It is expected that you will perform research and learning through as many Internet based resources as you can find. While some general guidelines are given below, put in efforts to learn as much as you can so that you can better architect and complete the application with minimal to no guidance.

**Core Skills to Learn:**

* LangChain fundamentals (chains, agents, memory)
* RAG implementation
* Ollama model management
* FastAPI backend development
* Basic frontend development

**Recommended Resources:**

* LangChain documentation
* Ollama GitHub repo
* FastAPI official tutorial
* RAG best practices blogs

# Roadmap

## Phase 1: Core Architecture Planning

Finalize the technology stack and core architecture for the application. This step will require you to research and come up with the proper documentation on how you will implement the application. A starting point is given here which you can build upon.

* **System Components**
  + Frontend: Simple UI with chat interface
  + Backend API: FastAPI or Flask
  + AI Agent Core: LangChain + LLM
  + Knowledge Base: For RAG (collect data on travel info, hotels, destinations from Internet)
  + Booking Integration APIs: For actual reservations – This is optional
* **Technology Stack**
  + LLM Options:
    - Mistral 7B (quantized)
  + Framework: LangChain + Ollama (for local LLM serving)
  + Vector DB: ChromaDB (lightweight) or FAISS
  + Backend: FastAPI (Python)
  + Frontend: Optional: Angular / React / Vue.js / Simple command lineon Ubuntu Terminal is also Ok

Ensure that all the technology stack components are installed before development can be carried out.

Read and understand this link: <https://python.langchain.com/docs/tutorials/rag/>

You can also take help from other Internet tutorials for RAG using LangChain + Ollama.

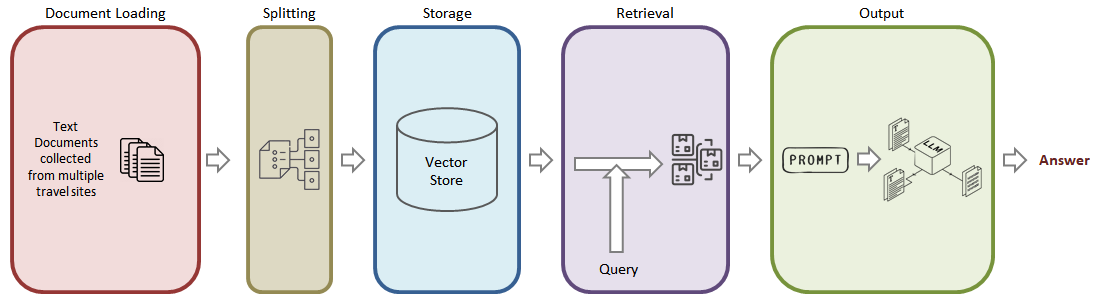


Figure : Reference Model for the Application

At the end of this step, you are required to submit either a Word document or a PowerPoint presentation detailing your approach to complete the application.

## Phase 2: Development Steps

1. Set Up Local LLM Infrastructure (Ollama with Mistral 7B quantized model)
2. Optional Step: Create Basic Chat Interface (or you can continue using Ubuntu Terminal for chat)
3. Build Travel Knowledge Base for RAG
   1. Collect Data on Tourist attractions in Telangana:
      1. Copy articles from popular travel websites including user comments and experiences shared by their users
   2. Process Data
      1. Use Langchain to split documents and create vector store
      2. Refer this link for understanding each step: <https://python.langchain.com/docs/tutorials/rag/\>
      3. A sample Python code is given below for this step
4. Implement RAG Pipeline which will result in an application that takes a user question, searches for documents relevant to that question, passes the retrieved documents and initial question to the Mistral 7B quantized model, and returns an answer.

## Phase 3: Advanced Features

Once you have a basic application running, you can attempt advanced features. These are optional enhancement and should only be attempted if the basic app is functioning properly.

* Itinerary Generation
* Add Multi-Modal Capabilities to show image/videos of the hotels, tourist places, and nearby attractions to the user if they so ask

End of Document