DBMS Term Project Report

SSH AP Daemon

Srinjoy Das	**22CS30054**
Priyanshu Gaurav	**22CS10083**
Aditya Kumar Bharti	**22CS30007**
Heyramb Agrawal	**22CS30030**
Sagnibha Roy	**22CS10064**

14th April, 2025

Contents

1. Introduction …………………………………………………………………… 3
2. Objective ………………………………………………………………………3
3. Methodology……………………………………………………………………4
   3.1 Techstack ……………………………………………………………………4
   3.2 ER-Diagram……………………………………………………………………5
   3.3 LLM-SQL ……………………………………………………………………5
   3.4 Agentic Framework utilizing LLM Agents having SQL Tools………………...6
   3.5 LLM SQL Pipeline using Agents having SQL Tools…………………………..6
   3.6 RAG based Approach ………………………………………………………7
   3.7 Travel Planning using LLM-SQL and Multi Agent Systems…………………..7
   3.8 Multi Agent Architecture forTravel Planning………………………………..8
4. Results……………………………………………………………………………9
   4.1 Comparison of both methods for different queries……………………………..9
   4.2 Token Analysis for the LLM SQL part using SQL Tooling…………………...10
5. Conclusion……………………………………………………………………12
6. References……………………………………………………………………12

1. Introduction

This document outlines the key stakeholders, technology stack, and functionality, providing a comprehensive roadmap for our development process.The travel planner website is designed as a robust, multifunctional platform aiming to streamline travel arrangements by integrating hotel booking, itinerary planning, and transportation services in one centralized interface. The system is built around four key user roles: Customer, Hotel Administrator, System Administrator, and Driver. Each role has specifically tailored features to cater to its unique responsibilities and requirements, thereby ensuring a smooth and efficient user experience.

2. Objective

Seamless User Experience for Customers:

- Provide an integrated platform for hotel searches, bookings, and itinerary planning and crafting.
- Empower users with interactive functionalities such as viewing and managing booking histories (upcoming, past, and canceled bookings).
- Deliver personalized travel recommendations through an AI-powered assistant that performs intelligent searches and itinerary suggestions based on user preferences and existing database.

Efficient Hotel Management:

- Enable hotel administrators to manage room inventories, track bookings, and analyze revenue efficiently.
- Provide access to customer reviews and feedback for continuous service improvement.
- Streamline operations with robust analytics and reporting tools to optimize hotel performance.

Comprehensive System Oversight:

- Allow system administrators to manage the entire website, ensuring data integrity, system security, and overall functionality.
- Facilitate easy scaling and integration of new features as the platform grows.
- Implement monitoring and management dashboards to quickly resolve issues and maintain high uptime.
- Allow registering and removal of customers, hotels and transport facilities as well as monitoring.

Empowered Transportation Services:

- Integrate driver functionalities to efficiently handle service requests and manage journeys.

- Provide detailed income and performance analysis for drivers, allowing them to optimize their service delivery.
- Ensure a streamlined process for accepting, tracking, and completing transportation requests, thereby enhancing overall connectivity between travel services.

Enhanced AI Assistance:

- Integrate a conversational AI assistant capable of planning itineraries based on user inputs and real-time travel data.
- Allow the AI assistant to fetch dynamic travel information and suggest attractions, ensuring that the recommendations are up-to-date.
- Enhance customer engagement and satisfaction by providing quick and accurate responses to travel-related inquiries

Security and Authentication:

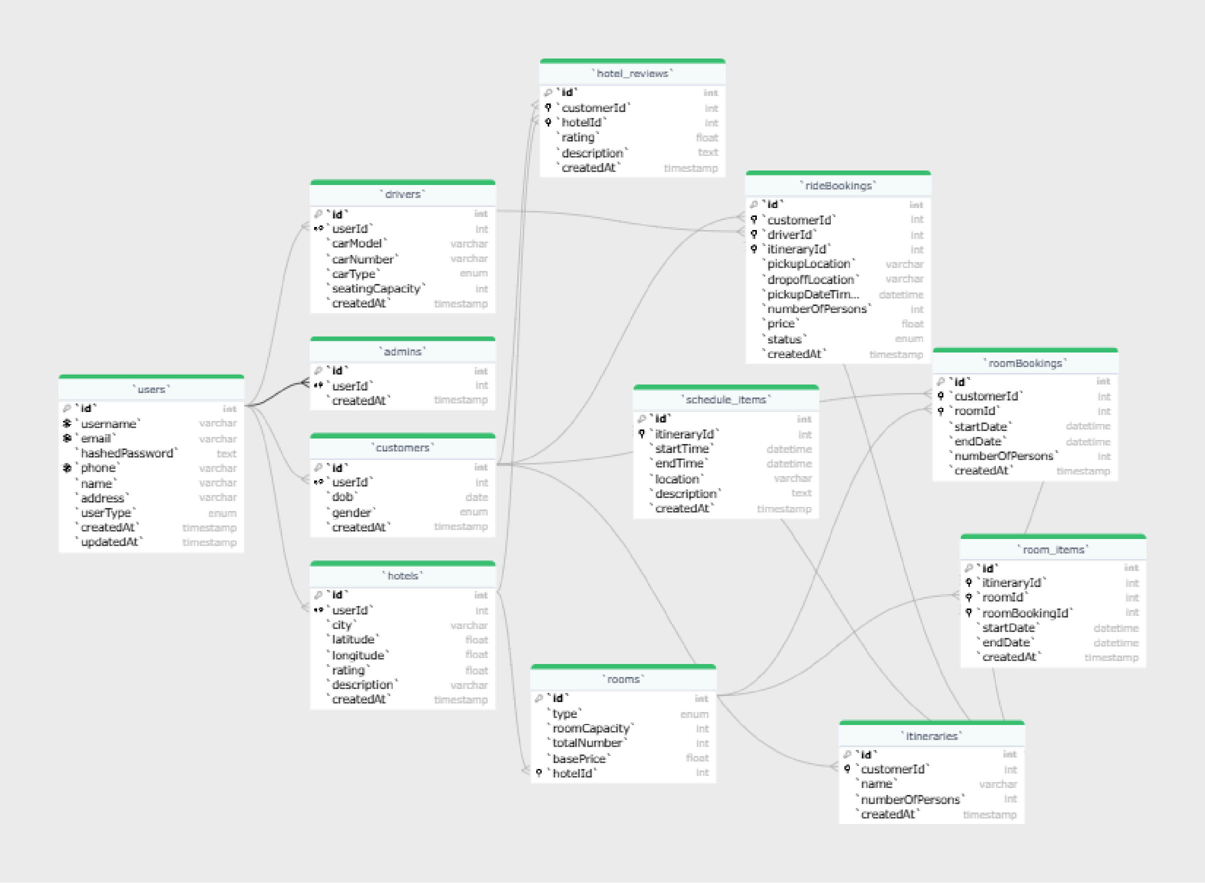
- Used JSON Web Tokens (JWT) stored in cookies for authentication.
- Implemented secure cookie attributes: HttpOnly, Secure, and SameSite to mitigate XSS (Cross-site scripting) attacks.
- Applied robust role-based access control (RBAC) so that each user type accesses only their designated data

3. Methodology

3.1 Techstack

- Frontend using React.js coupled with Tailwind CSS and shadcn-ui, to ensure an engaging and responsive interface, hosted on netlify.
- Backend developed in FastAPI to deliver fast, reliable, and scalable server-side performance.
- PostgreSQL database hosted on Azure for secure, scalable, and high-performance data management across all user roles.
- Used mockaroo to populate our database for testing and, demo purposes.
- Agentic Framework : Agno, google-genai
- RAG : vanna-ai and google-genai

3.2 ER-Diagram



3.3 LLM-SQL

This approach compares two distinct approaches to transform natural language queries into SQL database queries: 1) an agentic framework utilizing LLM Agents having SQL Tools using Agno, and 2) a RAG-based (Retrieval-Augmented Generation) system using with ChromaDB, Google Gemini and, all-miniLM-v2 as embedding model. Both approaches aim to solve the same fundamental problem—enabling non-technical users to query databases using natural language—but they differ significantly in their architecture, processing methodologies, and performance characteristics.

Converting natural language queries to accurate SQL queries presents significant challenges, including understanding user intent, mapping to database schema, handling ambiguity, and generating syntactically correct SQL. The two approaches analyzed here represent different paradigms for tackling these challenges: one using an agent-based system with specialized tools and the other using vector-based retrieval to augment LLM generation.

3.4 Agentic Framework utilizing LLM Agents having SQL Tools

The agentic framework approach implements a multi-agent system that breaks down the query translation process into specialized components:

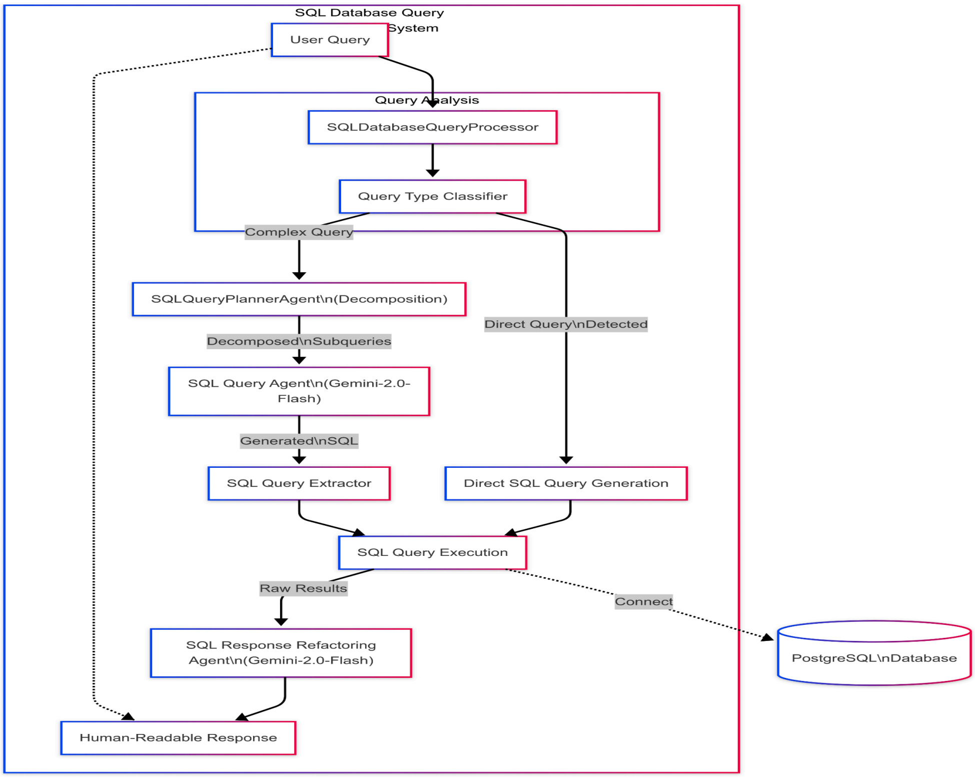
- **Query Understanding and Planning:** The system first analyzes the natural language query for intent and semantic structure, using a SQLQueryPlannerAgent to decompose complex queries into simpler sub-queries when needed.

- **Adaptive Query Generation:** When direct pattern matching isn't applicable, the system delegates to a specialized SQL generation agent that has been configured with database schema knowledge.

- **Fallback Mechanisms:** A hierarchical fallback system allows progression from direct pattern-matched queries to agent-based generation to generic template-based queries as a last resort.

- **Post-Processing:** Query results undergo post-processing through a refactoring agent to transform raw database outputs into human-readable responses.

3.5 LLM SQL Pipeline using Agents having SQL Tools



3.6 RAG based Approach

The RAG-based approach combines vector storage of SQL examples with generative AI to produce SQL queries:

- **Knowledge Base Creation:** The system indexes a collection of SQL query examples along with whole database information in a vector database (ChromaDB), creating embeddings that capture the semantic meaning of both queries and their corresponding database operations.

- **Schema Interpretation:** Database schema information is processed to understand table structures, relationships, and constraints, which informs query generation.

- **Retrieval and Similarity Matching**: When a natural language query arrives, the system retrieves the most semantically similar SQL examples from its vector store.

- **Context-Enriched Generation:** The LLM (Google Gemini) receives both the user query and the retrieved similar SQL examples as context to generate a new, contextually appropriate SQL query.

- **Execution and Response Generation:** The system executes the generated SQL query and uses the LLM again to transform results into natural language responses.

3.7 Travel Planning using LLM-SQL and Multi Agent Systems

The travel planning system is structured as a modular, agent-oriented architecture that ensures secure, efficient, and context-aware travel planning. Upon receiving a User Query, the system engages two critical components: the LocationExtractor, responsible for identifying relevant geographical entities, and the GuardrailsAgent, which performs a Safety Check to validate and sanitize the query.

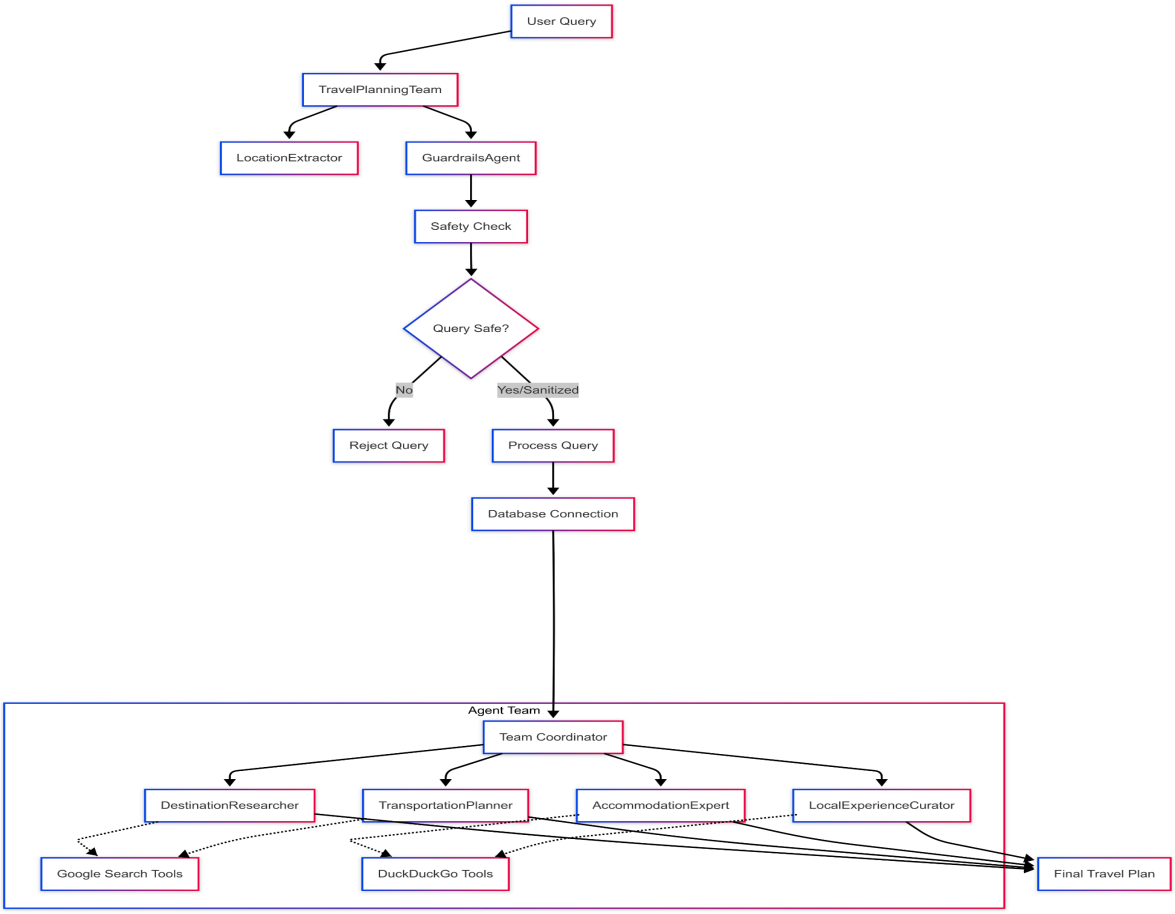
If the query passes the safety validation, it is forwarded to the data processing pipeline. Here, a SQL Tool is employed to establish a Database Connection and execute relevant SQL Queries. The extracted insights are then structured and utilized to drive the next phase of the system.

At this stage, a coordinated Agent Team—led by the Team Coordinator (Gemini-2.0-Flash)—is activated to generate a tailored travel itinerary. This team comprises the following specialized agents:

- **DestinationResearcher (Gemini-2.0-Flash)**: Aggregates destination-specific insights using Google Search tools.
- **TransportationPlanner (Gemini-2.0-Flash):** Constructs transportation options and logistics using intelligent search capabilities.
- **AccommodationExpert (Gemini-2.0-Flash):** Recommends lodging alternatives based on user preferences through curated web searches.
- **LocalExperienceCurator (Gemini-2.0-Flash):** Identifies culturally immersive experiences and local activities via DuckDuckGo tools.

Each agent contributes domain-specific intelligence, which is synthesized by the coordinator into a Final Travel Plan, delivering a seamless and personalized travel solution to the user.

3.8 Multi Agent Architecture forTravel Planning



4. Results

4.1 Comparison of both methods for different queries

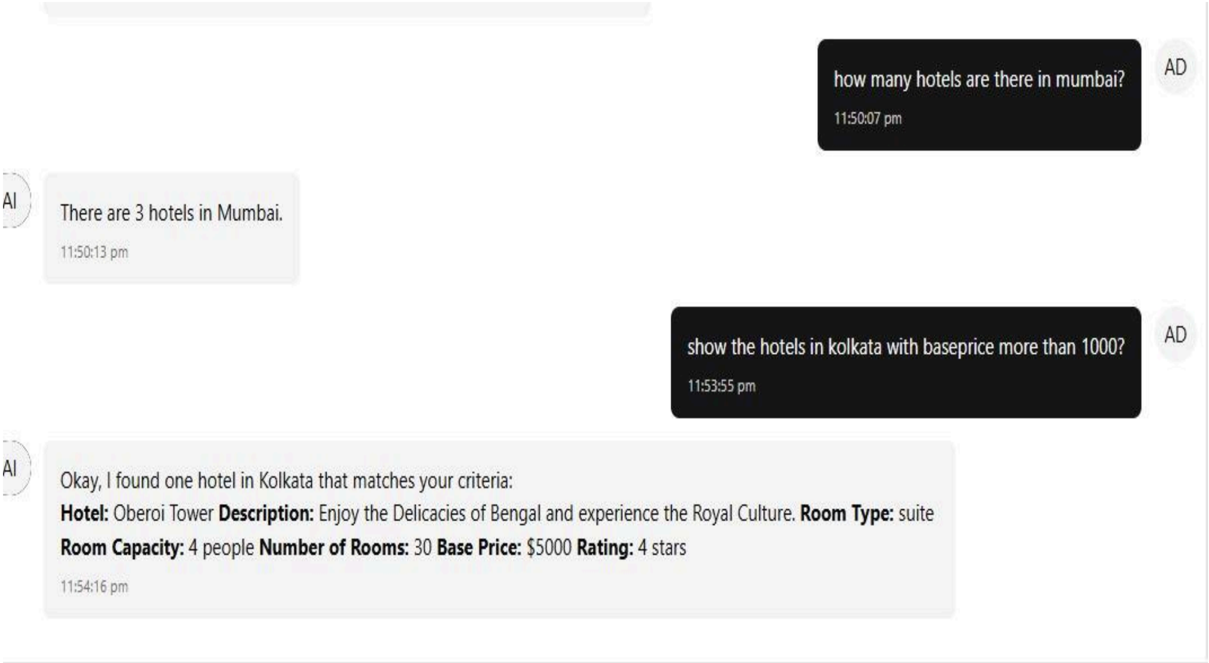
Question	SQL Tooling(Gemini)	SQL Tooling(Llama-3-70B)	SQL Tooling(Llama-3-8B)	RAG based approach
How many hotels are in the database?	The database contains 5 hotels. They are raj, Taj, Rosewood, Upper-house, and madurai elite.\n	I couldn't find any information matching your query in the database.	I couldn't find any information matching your query in the database.	5
What are the hotels in Mumbai?	The hotels in Mumbai are Taj, Rosewood and Upper-house.	The hotels are Taj, Rosewood and Upper-house.	The hotels are Taj, Rosewood and Upper-house.	Not able to answer
What is the highest rating of any hotel in the database	The highest rating of any hotel in the database is 5.0.\n	I couldn't find any information matching your query in the database.	5.0	5.0
What are the hotels in Chennai?	The hotel 'madurai elite' is located in Chennai with a rating of 5.0. It is located on MG street and is known as a great hotel.\n	The hotels in Chennai are ‘madurai elite’.	The hotels in Chennai are ‘madurai elite’.	Not able to answer
How many hotels are there in Chennai?	There is one hotel in Chennai.	One hotel is present in Chennai.	There is one hotel in Chennai.	1
Number of persons in itinerary for Summer Vacation 2025	The itinerary 'Summer Vacation 2025' has 5 person(s).	Not able to answer.	The vacation has 5 persons.	5
Number of scheduled	There is one scheduled item in	One scheduled item is there.	Not able to answer.	Not able to answer

items in summer vacation 2025 itinerary.	the Summer Vacation 2025 itinerary.\n			
What are the average room prices in Mumbai?	I cannot answer the question about the average room prices in Mumbai with the data provided. The hotel information provides the base price of a suite but does not include any average prices for rooms in Mumbai.\n"	Not able to answer.	Not able to answer.	Not able to answer

4.2 Token Analysis for the LLM SQL part using SQL Tooling

Question	**Input token**	**Output token**
How many hotels are in the database?	**335**	**21**
Here are the hotels in Mumbai?	**287**	**16**
What is the highest rating of any hotel in the database?	**411**	**15**
What are the hotels in Chennai?	**432**	**38**
How many hotels are there in Chennai?	**234**	**13**
Number of persons in itinerary for Summer Vacation 2025	**303**	**25**
Number of scheduled items in summer vacation 2025 itinerary.	**291**	**29**
What are the average room prices in Mumbai?	**322**	**17**

Some Screenshots of our demo of llm-sql

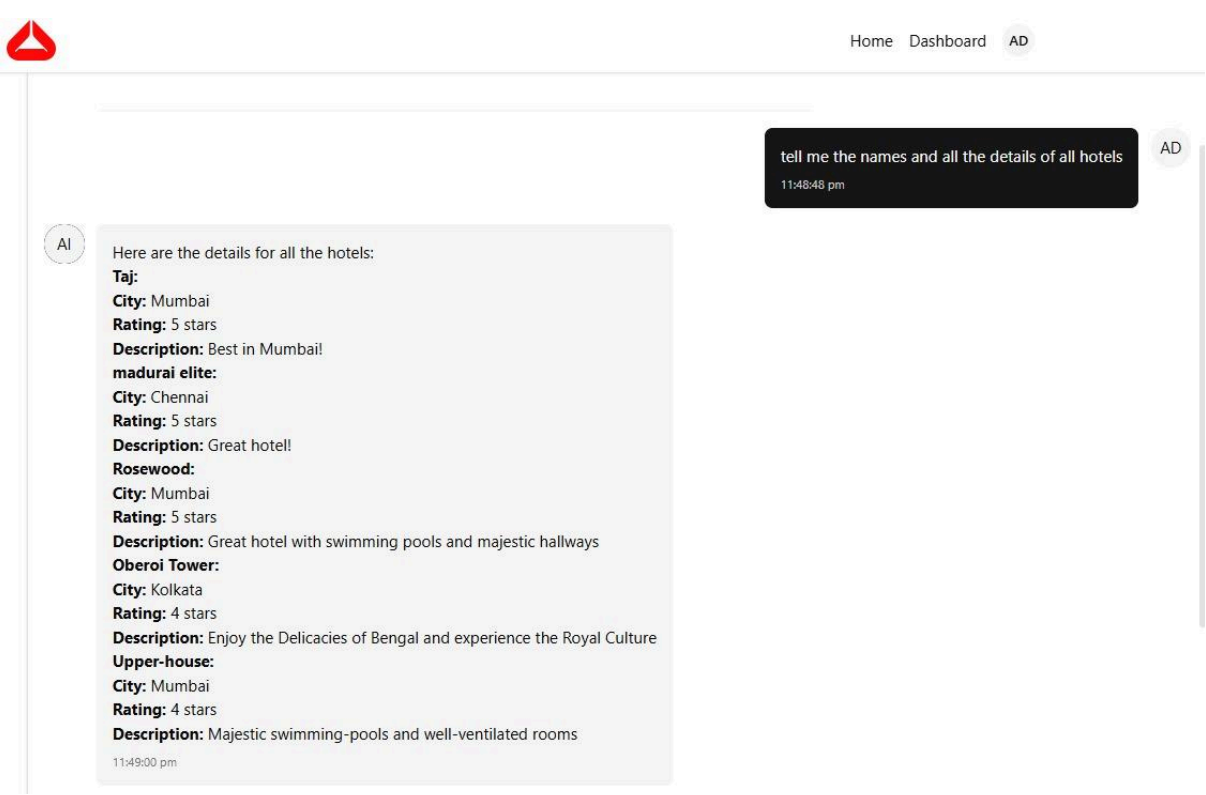


how many hotels are there in mumbai?
11:50:07 pm

There are 3 hotels in Mumbai.
11:50:13 pm

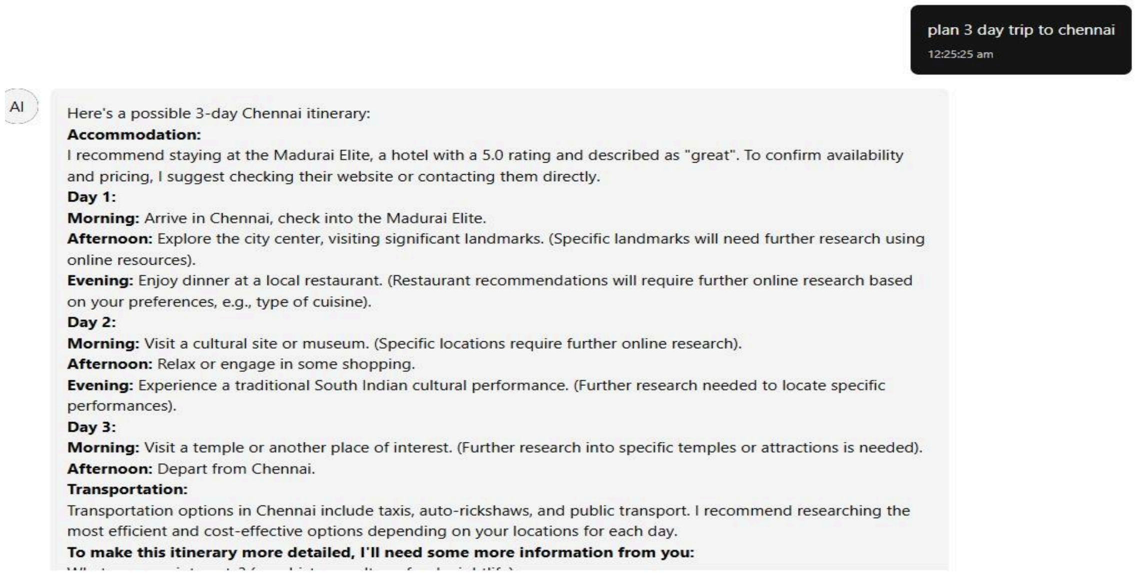
show the hotels in kolkata with baseprice more than 1000?
11:53:55 pm

Okay, I found one hotel in Kolkata that matches your criteria:
Hotel: Oberoi Tower **Description:** Enjoy the Delicacies of Bengal and experience the Royal Culture. **Room Type:** suite **Room Capacity:** 4 people **Number of Rooms:** 30 **Base Price:** $5000 **Rating:** 4 stars
11:54:16 pm



tell me the names and all the details of all hotels
11:48:48 pm

Here are the details for all the hotels:
Taj:
City: Mumbai
Rating: 5 stars
Description: Best in Mumbai!
madurai elite:
City: Chennai
Rating: 5 stars
Description: Great hotel!
Rosewood:
City: Mumbai
Rating: 5 stars
Description: Great hotel with swimming pools and majestic hallways
Oberoi Tower:
City: Kolkata
Rating: 4 stars
Description: Enjoy the Delicacies of Bengal and experience the Royal Culture
Upper-house:
City: Mumbai
Rating: 4 stars
Description: Majestic swimming-pools and well-ventilated rooms
11:49:00 pm



plan 3 day trip to chennai
12:25:25 am

AI

Here's a possible 3-day Chennai itinerary:

Accommodation:

I recommend staying at the Madurai Elite, a hotel with a 5.0 rating and described as "great". To confirm availability and pricing, I suggest checking their website or contacting them directly.

Day 1:

Morning: Arrive in Chennai, check into the Madurai Elite.

Afternoon: Explore the city center, visiting significant landmarks. (Specific landmarks will need further research using online resources).

Evening: Enjoy dinner at a local restaurant. (Restaurant recommendations will require further online research based on your preferences, e.g., type of cuisine).

Day 2:

Morning: Visit a cultural site or museum. (Specific locations require further online research).

Afternoon: Relax or engage in some shopping.

Evening: Experience a traditional South Indian cultural performance. (Further research needed to locate specific performances).

Day 3:

Morning: Visit a temple or another place of interest. (Further research into specific temples or attractions is needed).

Afternoon: Depart from Chennai.

Transportation:

Transportation options in Chennai include taxis, auto-rickshaws, and public transport. I recommend researching the most efficient and cost-effective options depending on your locations for each day.

To make this itinerary more detailed, I'll need some more information from you:

5. Conclusion

The travel planner website is a forward-thinking platform that blends an innovative tech stack with a user-centric approach to redefine travel experiences. Leveraging React.js, FastAPI, and PostgreSQL on Azure, the system is both robust and secure, underpinned by comprehensive testing and stringent security measures—including JWT authentication with secure cookies and strict role-based access control. Its scalable architecture not only meets current demands but also accommodates future enhancements, positioning the platform for continuous growth and adaptability in the dynamic travel industry

6. References

1. Agno Framework : https://docs.agno.com/introduction
2. Langchain GoogleGenerativeAI : https://python.langchain.com/docs/integrations/chat/google_generative_ai/
3. Generating SQL for Postgres using Google Gemini, ChromaDB : https://vanna.ai/docs/postgres-gemini-chromadb/
4. React Documentation : https://react.dev/learn
5. Vite Documentation : https://vite.dev/
6. FastAPI Documentation : https://fastapi.tiangolo.com/
7. ShadCN : https://ui.shadcn.com/docs
8. Tailwind CSS Documentation : https://tailwindcss.com/
9. Netlify Documentation : https://docs.netlify.com/
10. Agno SQL Tooling : https://github.com/agno-agi/agno/blob/main/cookbook/tools/sql_tools.py
11. SQL ToolKit Langchain : https://python.langchain.com/docs/integrations/tools/sql_database/
12. Google AI Studio : Chat | Google AI Studio