IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Travease-Travel Experience Agent

Thejaswini M[1], Yashas Raj U[1], Jeevan B[1], Yashaswini AR[2]

[1] *Students, Department of CSE – AI & ML, MIT Mysore, Mysuru, India*

[2]*Assistant Professor, Department of CSE – AI & ML, MIT Mysore , Mysuru, India*

Abstract: Contemporary travel platforms frequently depend on rigid interfaces and manual filtering processes, which hinder effective interaction and decision-making support. This document introduces TravEase, a conversational travel management system powered by Agentic AI that enables users to access and oversee structured travel information using natural language inquiries. Developed with Google's Gemini 2.0 Flash model, the system employs a tool-oriented Agentic AI strategy wherein the agent interprets user intentions and seamlessly activates backend functions for data retrieval, analytics, and auditing purposes. TravEase incorporates a FastAPI backend, an SQLite database, and real-time response streaming to guarantee precise, data-supported outputs. The system showcases enhanced usability, decreased manual workload, and dependable conversational decision-making assistance.

Index Terms - Conversational AI, Agentic Systems, Travel Management, ReAct Framework, Decision Support.

I. INTRODUCTION

Travel management systems play a key role in tourism and business travel, where users expect fast and accurate information. Despite the rise of digital travel systems (DTSs), the majority still use static interfaces with manual filtering and provide a relatively slow and unintuitive user experience, requiring users to navigate multiple pages and repeat the same search parameters multiple times to compare travel options. Most travel sites do not provide smart assistance, do not understand natural language, and do not provide analyzes of price changes, popular categories or recently-added services. These restrictions may confuse the user and influence their travel decisions.

To address these challenges, this paper presents TravEase, an Agentic AI (agAI)-based conversational travel management system that allows travel data to be queried and processed in natural language. TravEase employs an agent-driven AI framework to automate data extraction from travel systems and exploit data analytics to provide conversational decision support.

II. LITERATURE REVIEW

Recent research indicates a growing integration of artificial intelligence within travel and tourism systems to enhance personalization, refine recommendation accuracy, and enrich user experience. Techniques such as machine learning and natural language processing have been extensively utilized to evaluate traveler preferences and streamline information retrieval. Nevertheless, numerous travel platforms continue to depend on fixed interfaces and lack intelligent conversational features. Studies on conversational agents underscore their value in assisting users; however, the majority of these systems are based on rules and are incapable of undertaking advanced reasoning, analytics, or backend tasks. Other research focuses on travel data analysis and package management but fails to enable natural-language interaction or real-time auditing. The literature reviewed highlights a deficiency in systems that merge conversational intelligence with organized backend processes. TravEase fills this void by implementing an Agentic AI approach that integrates natural-language interaction, analytics, and auditing into a cohesive framework.

III. SYSTEM METHODOLGY

TravEase is designed utilizing a modular and tiered approach to guarantee scalability, dependability, and a clear delineation of roles. The system employs an Agentic AI framework, wherein the AI agent functions as a reasoning mechanism that interprets user inquiries and identifies suitable backend tools for execution. Rather than producing responses directly, the agent assesses user intent, extracts necessary parameters, and calls specific functions to carry out CRUD operations, analytics, and auditing activities. The backend logic is constructed with FastAPI and a SQLite database, facilitating efficient data storage and access. Each backend process is encapsulated within precisely defined tools to uphold consistency and minimize errors. Real-time conversational engagement is enabled through Server-Sent Events (SSE), allowing responses to be delivered progressively to the user. This approach guarantees accurate, data-supported responses while enhancing system maintainability and improving user experience.

IV. SYSTEM ARCHITECTURE

The TravEase system is constructed using a tiered architecture aimed at ensuring modularity, scalability, and effective communication among its components. This structure includes a chat-based user interface, an Agentic AI layer, a business logic layer, a repository layer, and a SQLite database. Users engage with the system via a conversational platform, sending inquiries that are relayed to the AI agent for analysis. The Agentic AI layer interprets user intentions and dynamically selects the appropriate backend tools to perform necessary tasks such as data retrieval, analysis, or auditing. The tools layer encompasses essential business logic and interfaces with the repository layer, which oversees all interactions with the database. The data retrieved from the database is processed by the agent and transmitted back to the user interface in real time. The accompanying Figure 1 illustrates the organized architecture that guarantees dependable, data-driven responses and facilitates future improvements to the system.

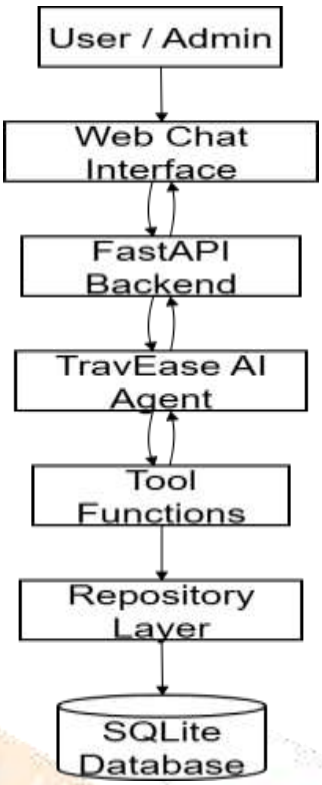

Figure 1 : System Architecture Diagram

V. DATA MODEL DESIGN

TravEase employs an organized relational data framework to effectively store and oversee travel-related data. The main entity, Travel Combo, encapsulates details like package title, description, category, base cost, linked agency, and timestamps generated by the system. These attributes facilitate crucial actions including creation, alteration, filtering, and analytical assessment of travel packages. Additionally, an Agency entity is incorporated to signify travel agencies, gathering information such as agency name, operational region, and rating. The link between travel combos and agencies permits region-based analysis and cross-entity inquiries. This structured data model ensures data integrity, promotes efficient CRUD operations, and supports dependable analytics and auditing features within the system.

VI. IMPLEMENTATION

The TravEase system employs a modular backend architecture created with Python 3.11, which prioritizes both maintainability and scalability. The backend utilizes FastAPI to offer high-performance RESTful API endpoints and facilitates asynchronous request processing. Input validation and data serialization are handled through Pydantic models, ensuring that all data submitted by users adheres to established schemas prior to processing.For persistent data storage, a lightweight SQLite relational database is utilized. Connectivity to the database and the execution of queries are managed by a specialized repository layer that abstracts direct SQL operations and reinforces the principle of separation of concerns. This repository layer enables efficient CRUD operations, category- based filtering, price-based inquiries, and timestamp-driven auditing. System-generated timestamps are automatically kept to accurately monitor the creation and modification of records.

The Agentic AI layer integrates Google's Gemini 2.0 Flash model with the help of an Agent Development Kit and operates within a tool-based execution framework. The agent conducts intent analysis on user inquiries, identifies relevant parameters, and dynamically engages backend tools instead of producing free-form responses. This methodology significantly minimizes inaccuracies and ensures that all outputs are

derived from verified database operations. Real-time conversational interaction is facilitated through Server-Sent Events, allowing for the progressive streaming of agent responses to the frontend. Collectively, these implementation decisions ensure accurate and reliable conversational travel management.

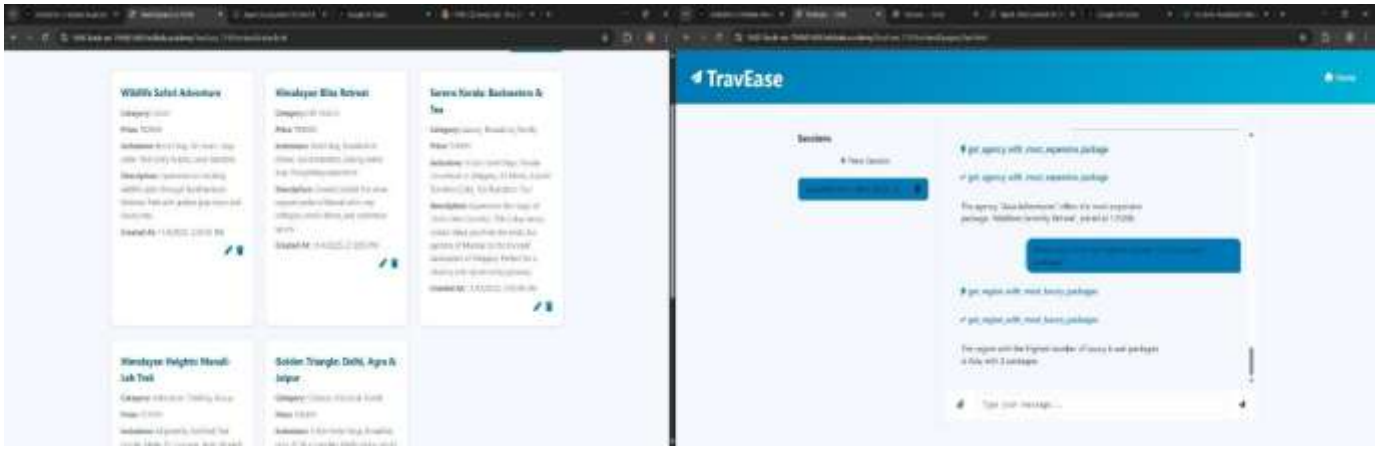

Figure 2 : TravEase Travel Packages – User Dashboard

VII. RESULT ANALYSIS

The TravEase system underwent a series of functional and interaction-centric assessments to evaluate its efficacy, dependability, and conversational precision. It successfully interpreted a diverse array of natural-language inquiries pertaining to travel package management, analytics, and auditing. The Agentic AI framework consistently discerned user intentions and activated the necessary backend tools, showcasing strong reasoning and decision-making abilities.

Response times remained minimal thanks to the efficient FastAPI backend and streamlined SQLite-based data retrieval. CRUD tasks such as creating, altering, obtaining, and removing packages were performed accurately, with system-generated timestamps guaranteeing dependable tracking of recent modifications. Analytical inquiries, including categorization-based filters and price evaluations, yielded consistent and verifiable outcomes directly from stored data.

The implementation of a tool-driven execution model significantly minimized erroneous responses, as all outputs were based on confirmed database functions instead of unstructured text generation. Real-time response delivery through Server-Sent Events enhanced the user experience by offering incremental feedback throughout the interaction. Overall, the findings suggest that TravEase provides reliable, accurate, and effective conversational travel management, affirming the success of the proposed Agentic AI framework.

VIII. CONCLUSION

This article introduced TravEase, a conversational travel management system driven by Agentic AI, aimed at addressing the shortcomings of conventional, menu-based travel platforms. By incorporating natural language communication alongside a tool- oriented backend structure, the system facilitates dependable execution of CRUD operations, analytics, and auditing functions, while ensuring that all outputs are based on confirmed database actions. The implementation of an Agentic AI framework rooted in the Reason-Act model markedly enhances intent comprehension, diminishes erroneous outputs, and improves the precision of responses. Experimental assessments illustrated that TravEase offers prompt, reliable, and data-

supported conversational exchanges, thereby enhancing user experience and decision-making efficiency. Its modular and scalable architecture permits the effortless inclusion of future improvements, such as recommendation systems, live travel APIs, multilingual capabilities, and various input methods. In summary, TravEase exemplifies the value of merging conversational AI with organized backend processes for advanced travel management solutions.

REFERENCES

[1] C. Li, X. Wang, and Y. Chen, "AI-Driven Travel Recommendation Systems: A Comprehensive Review," *IEEE Access*, vol. 12, pp. 34521– 34538, 2024.

[2] S. Ahmed and R. Joseph, "Conversational Agents for Smart Tourism Platforms," in *Proceedings of the IEEE International Conference on Intelligent Systems*, pp. 112–118, 2023.

[3] A. Santos, M. Ferreira, and J. Costa, "Intelligent Travel Package Management Using Machine Learning Techniques," *Journal of Intelligent Information Systems*, Springer, vol. 58, no. 2, pp. 341–360, 2022.

[4] T. Nakamura and K. Lee, "Human-Centered Tourism Analytics and Personalization Using AI," *IEEE Transactions on Computational Social Systems*, vol. 11, no. 1, pp. 85–96, 2024.

[5] Y. Liu, P. Zhou, and H. Zhang, "Conversational AI Systems: Architecture, Applications, and Challenges," *IEEE Internet Computing*, vol. 27, no. 3, pp. 44–53, 2023.

[6] D. Silver et al., "ReAct: Synergizing Reasoning and Acting in Language Models," *arXiv preprint arXiv:2210.03629*, 2022.

[7] Google Research, "Gemini: A Family of Highly Capable Multimodal Models," *Technical Report*, Google DeepMind, 2023.

[8] S. Kim and J. Park, "Tool-Augmented Large Language Models for Reliable Decision Support," *IEEE Software*, vol. 41, no. 2, pp. 60–68, 2024.

[9] S. Ramírez, L. Torres, and M. Gómez, "Design of RESTful APIs Using FastAPI for High-Performance Applications," *IEEE Access*, vol. 11,

pp. 78901–78912, 2023.

[10] R. Fielding, "Architectural Styles and the Design of Network-Based Software Architectures," Ph.D. dissertation, Univ. of California, Irvine,

2000.