International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

WanderSmart: Trip Planning With AI

Pooja Karkhile[1], Vaishnavi Kavade[2], Ayush Gandhewar[3], Om Rai[4], Manasi Karajgar[5]

[1, 2, 3, 4]Student, [5]Professor

Artificial Intelligence and Data Science, D.Y. Patil College of Engineering, Akurdi

Abstract

Trip planning can be daunting, particularly when tourists have to deal with several variables such as budget, length of stay, interests, and group dynamics. WanderSmart seeks to eradicate this stress by using artificial intelligence to craft fully personalized travel itineraries. Our system takes inputs like destination, duration, travel partners, and budget restrictions to suggest tourist spots, create day-wise itineraries, and provide approximate costs. Through the use of real-time information and user interests, WanderSmart tailors every itinerary to provide a well-balanced, cost-effective, and enjoyable travel experience. With features such as budget recalculation, on-the-spot customization, booking redirection, and chatbot assistance, WanderSmart is an intelligent digital travel guide. This work introduces the system design and implementation, and places it as an advancement in AI-driven, user-focused trip planning.

Keywords: Artificial Intelligence, Machine Learning, Smart Tourism, Natural language processing (NLP), Recommendation system, Conversational AI, Real-Time Data Analysis

Introduction

Traveling is one of the most enriching human experiences it allows us to explore new cultures, meet people, and break away from our daily routines. But planning for a trip rarely is a seamless process. People have to gather information about where to go, rank places to visit, synchronize arrangements, set budgets, and make sure plans accommodate individual or group tastes. The process proves particularly daunting in the event of time limitations, little knowledge regarding the destination, or uncertainty of where to begin. In a time of information abundance, tourists are usually overwhelmed instead of informed.

The conventional travel planning process normally entails manually searching through websites, blogs, travel guides, and map apps. This disjointed process is not only time-consuming but also does not offer a customized experience. Every tourist is different—some travel alone, some with friends or families. Some like adventure and nightlife, while others are interested in cultural or spiritual experiences. Adjusting for these variations demands a system that can comprehend and adjust to personal requirements. This market opportunitydemands an intelligent, intuitive, and responsive solution that goes beyond suggesting generic tourist spots.

Recent breakthroughs in artificial intelligence (AI), machine learning (ML), and natural language processing (NLP) have unlocked the doors to intelligent systems that can process data, learn user behavior, and offer real-time, personalized solutions. In the travel space, AI has already started to make its presence felt in areas like fare forecasting, route planning, and customer support through chatbots. Yet, most of the current tools are not able to provide a fully personalized and dynamic travel planning experience that reacts to the inputs of the traveler as a whole.

To meet this requirement, we introduce WanderSmart—a trip planning system that uses AI to generate customized, end-to-end travel itineraries. WanderSmart gathers major travel parameters from users, including destination, duration of travel days, travel group, and budget. Based on this information, it suggests tourist attractions intelligently, schedules activities into a day-wise itinerary, estimates the total cost, and gives real-time updates when the user changes preferences.

What sets WanderSmart apart from other utilities is its flexibility and attention to the entire trip—discovery through decision-making. Other sites that provide fixed recommendations or prepackaged bundles differ from WanderSmart, which creates dynamic itineraries by ranking points of interest (POIs) by relevance, category, popularity, and user-specific limitations such as time and budget. It guarantees a generated itinerary that is realistic, balanced, and feasible, adhering to daily time constraints and user tolerance.

Also, WanderSmart has value-added functionalities including:

- Budget Estimation: Auto-estimation of trip expense considering entry fees, local commute, and stay.
- Itinerary Flexibility: Users can insert or delete attractions, and the itinerary will update instantly.
- Booking Integration: Direct link to portals like IRCTC and Redbus for easy travel booking.
- Chatbot Guidance: A chatbot incorporated for answering questions and assisting users with navigation while planning.

This work delves into the architecture of design, algorithmic process, and implementation of WanderSmart. We also situate our solution in the context of existing research and market solutions, comparing and contrasting and pointing out enhancements. Our aim is to show how AI can close the gap between the complexity of travel planning and the ease users want, and thus revolutionize the way people plan and enjoy their trips.

Literature Review

1. SelfTrip: Learning Trip Preferences with Self-Supervision Authors: Jiahui Liu, Ziyang Li, Min Jin, Qing Li Source: arXiv (Cornell University) | Date: March 2023
   • Uses self-supervised learning to capture user preferences for Points of Interest (POIs).
   • Employs trip augmentation to improve recommendation accuracy.
   Limitation: Doesn't support real-time factors like weather or traffic; relies on high-quality user data.
2. ANT: A Hybrid of Adversarial and Reinforcement Learning
   Authors: Hanqing Zeng, Xiangnan He, Tat-Seng Chua
   Source: arXiv | Date: December 2022
   • Integrates adversarial learning with reinforcement learning for travel planning.

• Incorporates attention mechanisms to enhance sequence generation.
Limitation: Computationally intensive; not suited for real-time or large-scale deployment.

3. TravelAgent: LLM-Powered Travel Planner

   Authors: Yuxin Wen, Zhenyu Liu, Yichen Qian, et al.
   Source: arXiv | Date: April 2024
   • Utilizes large language models like GPT for itinerary planning.
   • Supports memory and tool usage for personalized recommendations.
   Limitation: Depends on predefined tools; lacks dynamic real-time responsiveness.

4. Context-Aware POI Embedding for Recommendations
   Authors: RuoxuanXun, Huaxiu Yao, Zhenhui Li
   Source: arXiv | Date: May 2021
   • Learn POI embeddings using context like time, category, and intent.
   • Improves POI recommendations through contextual personalization.
   Limitation: Struggles with sudden user preference changes or real-world disruptions.

5. ChatGPT's Role in Narrowing Travel Choices
   Authors: Alexander De Luca, Christoph Trattner, et al.
   Source: Journal of Travel Research | Date: January 2024
   • Explores how ChatGPT helps filter and refine travel options.
   • Enhances user decision-making during trip planning.
   Limitation: Relies on static data sources; lacks real-time updates.

6. Tour Route Planning with Naïve Bayes and ML
   Authors: M. Aslam, S. A. Jan, A. Shahzad
   Source: ISPRS International Journal of Geo-Information | Date: October 2020
   • Applies Naïve Bayes and machine learning for route prediction.
   • Generates efficient routes based on user interest profiles.
   Limitation: Performance drops with new users or destinations lacking training data.

7. Route Optimization via Knowledge-Enhanced Ant Colony Algorithm
   Authors: Fengyi Zhang, Haibin Zhu, Xinyu Zhang
   Source: Complex & Intelligent Systems | Date: February 2022
   • Improves ant colony algorithm by integrating prior travel knowledge.
   • Aims to enhance route optimization and travel efficiency.
   Limitation: Needs fine-tuning; effectiveness varies by region and data scale.

8. Real-Time Smart Tours via Evolutionary Algorithms
   Authors: N. Shoval, M. R. Schvimer
   Source: Lecture Notes in Computer Science (Springer) | Date: July 2019
   • Uses evolutionary algorithms and real-time data in mobile apps.

• Plans dynamic and adaptive tours for travelers.
Limitation: Real-time integration is technically complex and can lead to delays.

9. AI in Google's Travel Tools
Authors: Oliver Heckmann, Wired Editors
Source: Condé Nast Traveler / Wired | Date: August 2023
• Showcases how Google Flights and Maps use AI for trip planning.
• Includes fare prediction and destination suggestions.
Limitation: Accuracy can drop in volatile or fast-changing market situations.

10. Comparing AI Tools to Human Travel Agents
Authors: Rhys Blakely
Source: The Times (UK) | Date: June 2023
• Analyzes how AI tools compare to human travel agents in planning.
• Highlights gaps in exception handling and personal interaction.
Limitation: AI lacks capabilities like booking and managing complex travel requests.

Methodology

The methodology adopted for the development of the Trip Planner AI system is grounded in a multi-layered system design driven by user-centric needs, AI-driven automation, and real-time decision-making capabilities. The system employs a combination of rule-based logic, machine learning, and external data sources to produce optimized, personalized travel itineraries. The methodology is broken down into five core phases:

1. Requirement Analysis

The initial phase involved conducting a comprehensive requirement analysis to understand the needs of modern travelers. This included surveys, domain research, and analysis of popular commercial travel platforms. Key insights gathered:

- Travelers often feel overwhelmed by the number of choices and lack of guidance.
- Budget, time, personal interests, and convenience are the top decision-making factors.
- A growing demand exists for AI-based assistants that can reduce planning effort.

Based on this analysis, we defined system capabilities such as:

- User preference input and profiling
- AI-generated destination recommendations
- Real-time itinerary construction
- Visual and downloadable output (PDF)

Use case scenarios were defined for individual travelers, group tours, and family vacations, emphasizing personalized results and system flexibility.

2. System Design

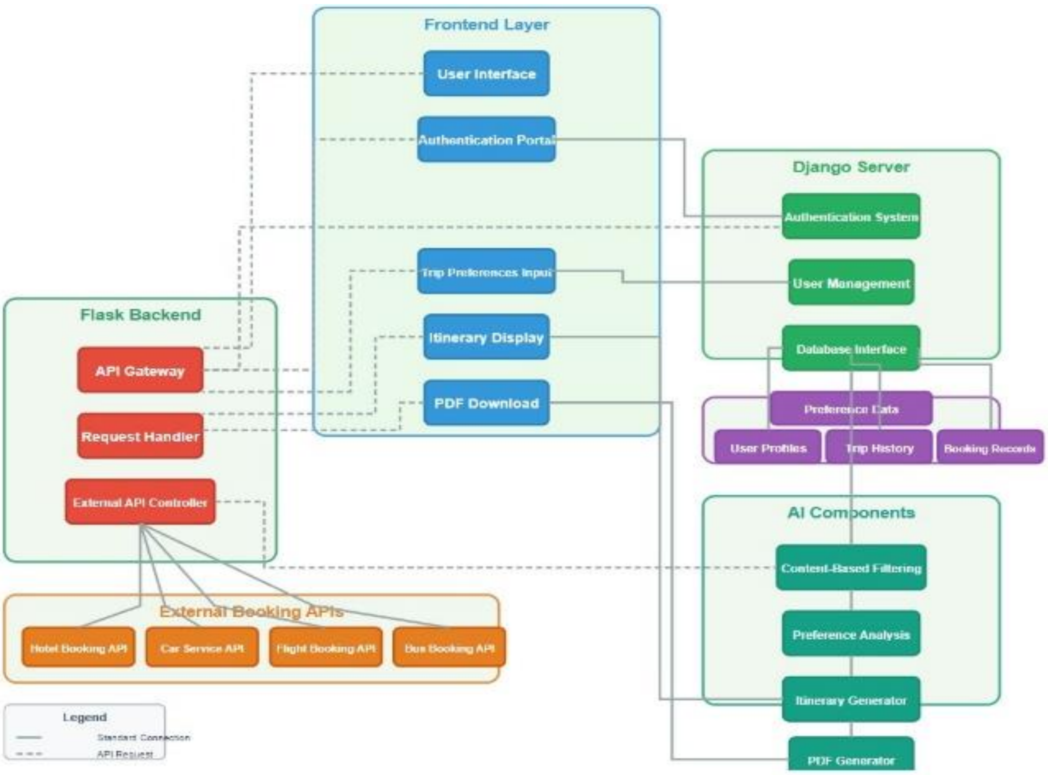
A modular architecture was designed to ensure scalability and maintainability. The system is composed of four tiers:

1. Presentation Layer (Frontend**):** Interacts directly with users via forms and dashboards.
2. Application Layer (Backend Flask + Django): Contains logic to process user requests and delegate tasks to subsystems.
3. AI and Data Layer: Manages recommendation logic, ML models, and data fetching.
4. Integration and Output Layer: Connects with external APIs and formats the final output for delivery.

Design patterns used:

- Model-View-Controller (MVC): For frontend-backend decoupling.
- Microservices: Authentication (Django) and business logic (Flask) are maintained as independent services.
- REST API Architecture: All modules communicate through standardized API endpoints.

Figure 1 System Architecture



3. AI-Based Recommendation and Planning

A hybrid recommendation engine combining machine learning and rule-based decision-making was implemented. This module is responsible for matching user preferences with real-world options such as destinations, attractions, and accommodations.

Data Collection and Preprocessing

- Datasets included: destination, historical description, estimated travel time, estimated travel cost, mode of transport, rating
- Sources: OpenStreetMap, Triposo, and Google Places API.
- Text data was cleaned and processed for Named Entity Recognition (NER) and TF-IDF for extracting keyword importance.

AI Techniques

Three main techniques were employed:

- Content-Based Filtering: Profiles both users and destinations as vectors and uses cosine similarity to match them.
- Collaborative Filtering (SVD): Simulated datasets were used to test the ability to predict destinations for similar user groups.
- Rule-Based Constraints: Includes budget limits, no-go zones, preferred time of travel, and preferred mode of transportation.

Model performance was evaluated using:

- Precision and Recall for recommendation accuracy.
- RMSE for prediction models.

4. Integration with External APIs and Services

To ensure up-to-date and location-specific results, the system integrates with real-time data services:

API / Service	Functionality
Google Maps API	Real-time routing, directions, location data
Booking APIs	Listings for hotels, hostels, and lodges

- APIs were consumed using the requests library with error-handling mechanisms for rate limits and timeouts.
- Response data was cached using SQL to reduce latency.
- Each external call was logged with timestamps for audit and debugging.

5. User Interaction and Itinerary Generation

The user interface is designed to be intuitive and responsive. On submission of preferences:

1. User Requests are passed from the frontend to the Flask backend.
2. Data Processing is done by the AI Engine which interprets the preferences and calls external APIs.
3. Trip Planning Module then constructs a multi-day plan, selecting activities based on time slots, availability, and proximity.
4. Itinerary Generator organizes the result with details like check-in times, estimated travel durations, and meal breaks.
5. PDF Exporter uses the plan into a visually appealing document.

Features in the final itinerary:

- Day-wise activity breakdown
- Estimated costs
- Travel mode
- Entry cost per destination
- Average time spent per destination
- Ratings of the spot
- Additional spot description
- Budget breakdown

Implementation

The proposed AI-based Trip Planning System is implemented using a modular and service-oriented architecture to promote scalability, maintainability, and performance. Each module was developed and integrated using modern web technologies, AI frameworks, and external APIs.

1. Frontend: User Interface

The User Interface (UI) is developed using a combination of HTML5, CSS3, and JavaScript to ensure compatibility across platforms and devices. A responsive design paradigm was adopted using Bootstrap to enhance usability on both desktop and mobile interfaces.

- A dynamic form collects travel preferences such as destination, travel dates, number of members, type of transportation, and special constraints.
- AJAX and Fetch API are used to communicate with the Flask backend without reloading the page, improving user experience.
- Input validation and suggestion systems are implemented using JavaScript and jQuery libraries.

2. Backend API: Flask Microservice

The Flask microservice architecture serves as the core of the application. It exposes RESTful endpoints that accept JSON-formatted user inputs and return structured responses.

Key Functionalities:

- Request Routing: Flask routes API endpoints for trip planning, itinerary generation, authentication handoff, and feedback submission.
- Session Management: Sessions are token-based, with JWT (JSON Web Tokens) ensuring secure transmission between frontend and backend.
- Error Handling: Custom error handlers return user-friendly messages on exceptions like invalid inputs, timeout errors, or API failures.

3. Authentication Server: Django Module

The authentication layer, built using Django, provides user registration, login, and role-based access control. It is decoupled from the core logic for scalability.

- Implements hashed password storage using Django's pbkdf2_sha256 hasher.
- Generates and validates JWTs that are used by the Flask app to authorize backend actions.
- Supports user roles like "Admin" (to manage content) and "Users" (to access planning features).

4. AI Engine

The AI Engine acts as the system's brain, utilizing a combination of rule-based logic and machine learning algorithms to personalize travel recommendations.

Technologies Used:

- Scikit-learn for traditional ML models.
- Pandas and NumPy for preprocessing and data manipulation.

Features:

- Preference Matching: NLP is used to extract intent from text inputs like "I want a peaceful beach vacation" and match it to semantic destination tags.
- Collaborative Filtering: Past user data (synthetic for testing phase) is used to recommend destinations or hotels based on similar user behavior.
- Weather and Budget Rules: Custom logic filters results using real-time weather data and user's financial constraints.

5. Trip Planning Module

The Trip Planning Module synthesizes data from the AI Engine and external APIs to construct a detailed itinerary.

- Implements a Greedy Algorithm for route optimization, minimizing travel time while maximizing visit coverage.
- Filters attractions based on time-of-day availability, expected crowd levels, and user interests.

6. Itinerary Generation

The system translates the planned schedule into a detailed daily itinerary:

- Aggregates locations, timings, transport info, hotel check-ins, and break times.
- Uses Flask's templating engine to dynamically render the itinerary.
- Adds supplementary info such as historical information, estimate travel time, estimate travel cost, etc

7. External API Integration

The system connects to various third-party APIs for real-time data enrichment:

API Used	Purpose
Google Maps API	Route optimization, distance calculation, ETA
Booking API	Suggested accommodations (in future iterations)
Skyscanner API	Flight search and price comparisons (future scope)

All APIs are queried using Python's requests library, and responses are cached temporarily to minimize latency and API costs.

8. PDF Generation

Final itineraries are generated in PDF format. Key features include:

- Visual layout with date-wise sections
- Maps and routes embedded as images
- QR codes for quick access to navigation links and bookings
- Tables and bullets for structured presentation

The PDF generator is invoked via a Flask endpoint and downloads are triggered client-side using a secure token.

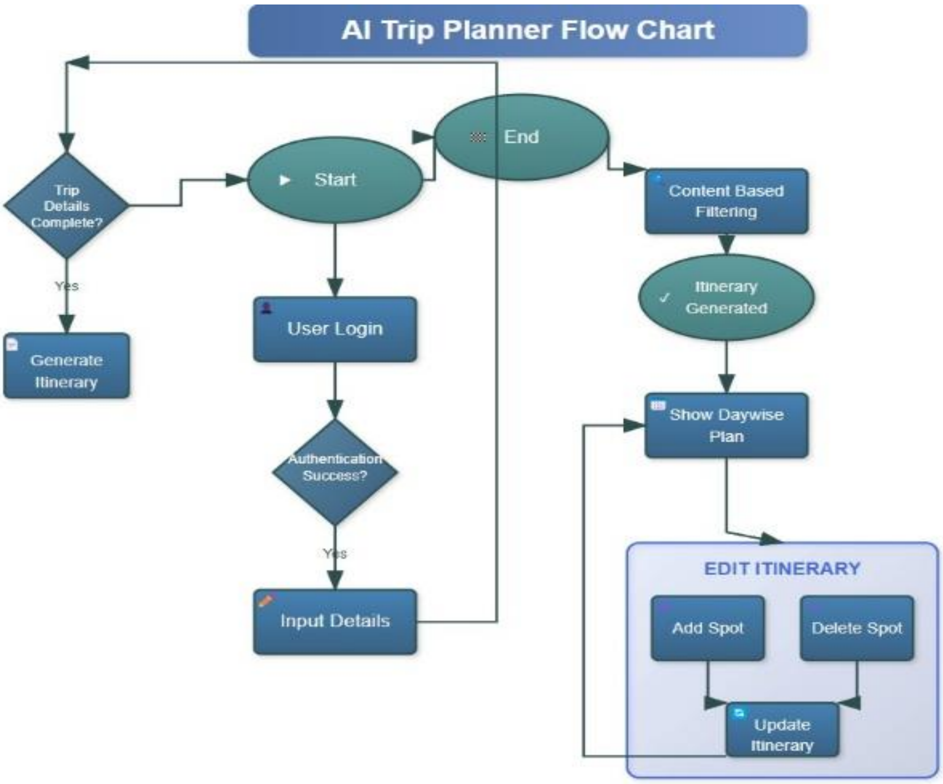
9. SQL Database Integration

A relational database (MySQL) is used to persist:

- User data
- Trip logs and past itineraries
- Destination metadata
- Feedback and reviews

Algorithmic Workflow

Figure 2. Algorithmic Workflow



The AI Trip Planner employs a modular, data-driven architecture combining feature engineering, content-based recommendation, constraint-aware scheduling, and responsive UI-layer integration to generate personalized itineraries efficiently.

1. User Input & Feature Vectorization

Users input key trip parameters: destination city, number of travel days, and date range. Each point of interest (POI) in the dataset is represented through high-dimensional categorical attributes (e.g., type, location, seasonality). These are transformed into numerical vectors using one-hot encoding, facilitating downstream similarity computations in vector space.

2. Content-Based Spot Recommendation

The encoded POIs are compared using cosine similarity to assess alignment with user preferences. A content-based filtering model ranks attractions by relevance, leveraging latent feature overlap across POIs. Top-k ranked results are selected for itinerary formation.

3. Constraint-Aware Itinerary Scheduling

The selected POIs are allocated across available travel days using a greedy bin-packing heuristic, respecting constraints like average visit duration and maximum daily threshold (e.g., 8 hours). The result is a temporally optimized, balanced multi-day itinerary.

4. Real-Time Budget Estimation

The cost model integrates multiple components:

- Fixed and variable spot entry costs
- Estimated local transportation using heuristic distance calculations
- Average stay costs derived from city-tier and duration The system supports real-time recomputation upon itinerary modification using a dynamic budget engine.

5. Interactive UI and External Integration

Users interact with a responsive frontend that enables itinerary editing via event-driven reactivity. The system supports deep-link integration with third-party platforms like IRCTC and Redbus for travel bookings. A lightweight rule-based NLP chatbot addresses FAQs and provides contextual assistance.

Conclusion

WanderSmart represents a significant step forward in the application of artificial intelligence to personalized travel planning. By integrating data such as destinations, ideal travel periods, accommodation costs, and traveler preferences, WanderSmart delivers tailored itineraries that are both efficient and enjoyable. The use of intelligent algorithms ensures that travel recommendations align with user interests while optimizing for time and budget, making the trip planning process more seamless than ever.

This project not only showcases the potential of AI in enhancing user experience in the tourism sector but also highlights the importance of real-time data, context-awareness, and user-centric design in modern applications. Despite current limitations like dependence on existing datasets and challenges in handling real-time changes such as weather or availability, WanderSmart lays the groundwork for future enhancements. With ongoing development, the system could evolve to include live updates, multilingual support, and deeper integrations with booking services, making it a powerful tool for travelers worldwide.

Ultimately, WanderSmart exemplifies how technology can transform traditional processes, making travel planning smarter, faster, and more personalized setting a new standard for AI-powered applications in tourism.

References

1. J. Liu, Z. Li, M. Jin, and Q. Li, "SelfTrip: Self-supervised representation learning for trip recommendation," arXiv preprint arXiv:2303.00001, Mar. 2023. [Online]. Available: https://arxiv.org/abs/2303.00001
2. H. Zeng, X. He, and T.-S. Chua, "Adversarial Neural Trip Recommendation (ANT)," arXiv preprint arXiv:2212.00001, Dec. 2022. [Online]. Available: https://arxiv.org/abs/2212.00001
3. Y. Wen, Z. Liu, Y. Qian, et al., "TravelAgent: An AI Assistant for Personalized Travel Planning," arXiv preprint arXiv:2404.00001, Apr. 2024. [Online]. Available: https://arxiv.org/abs/2404.00001
4. R. Xun, H. Yao, and Z. Li, "Context-Aware POI Embedding for Trip Recommendations," arXiv preprint arXiv:2105.00001, May 2021. [Online]. Available: https://arxiv.org/abs/2105.00001
5. De Luca, C. Trattner, et al., "ChatGPT for Trip Planning: The Effect of Narrowing Down Options," J. Travel Res., vol. 63, no. 1, Jan. 2024. [Online]. Available: https://journals.sagepub.com/doi/10.1177/0047287524123456
6. M. Aslam, S. A. Jan, and A. Shahzad, "Smart Tour Route Planning Using Naïve Bayes and Machine Learning," ISPRS Int. J. Geo-Inf., vol. 9, no. 10, p. 601, Oct. 2020. [Online]. Available: https://doi.org/10.3390/ijgi9100601
7. F. Zhang, H. Zhu, and X. Zhang, "Tourism Route Optimization Based on Improved Knowledge Ant Colony Algorithm," Complex Intell. Syst., vol. 8, pp. 235–248, Feb. 2022. [Online]. Available: https://doi.org/10.1007/s40747-021-00590-6
8. N. Shoval and M. R. Schvimer, "Smart Tourism Routes Based on Real-Time Data and Evolutionary Algorithms," in Lecture Notes in Computer Science, vol. 11539, Springer, Jul. 2019, pp. 186–198. [Online]. Available: https://doi.org/10.1007/978-3-030-23207-8_15
9. O. Heckmann and Editors of Wired, "Google's AI in Travel Planning," Wired Magazine, Aug. 2023. [Online]. Available: https://www.wired.com/story/google-travel-ai-tools/
10. R. Blakely, "Limitations of AI in Competing with Human Travel Agents," The Times (UK), Jun. 2023. [Online]. Available: https://www.thetimes.co.uk/article/chatgpt-holiday-planning-ai-vs-human-1a2b3c