ISSN: 1672 - 6553

JOURNAL OF DYNAMICS AND CONTROL

VOLUME 9 ISSUE 8: 74 - 88

WANDER WISE: IMPLEMENTATION OF AN AI-POWERED PERSONALIZED TRAVEL ITINERARY PLANNER

Vanishri Sataraddi[1], Divyashree S[2], Sowmyashree S[3], Kavitha B[4]

[1,4]Assistant Professor, Department of CSE,
[2,3]Assistant Professor, Department of ISE,
RNS Institute of Technology, Bangalore, India

DOI: https://doi.org/10.71058/jodac.v9i8007

WANDER WISE: IMPLEMENTATION OF AN AI-POWERED PERSONALIZED TRAVEL ITINERARY PLANNER

Vanishri Sataraddi*[1], Divyashree S[2], Sowmyashree S[3], Kavitha B[4]

[1,4,]Assistant Professor, Department of CSE, [2,3]Assistant Professor, Department of ISE, RNS Institute of Technology, Bangalore, India

[1]vanishrisataraddi@gmail.com, [2]*divyashrees2008@gmail.com,* [3]*sowmyashree532@gmail.com,* [4]*kavitha.sonu94@gmail.com*

Abstract—This paper presents a comparative survey of au- tomated travel itinerary planning systems, focusing on systems that use artificial intelligence and machine learning to personalize travel experiences. By examining three recent papers, we identify core features, strengths, and limitations across various trip planning technologies, including traditional POI-based itinerary planners, generative AI tools like ChatGPT, and community- based travel management apps. This survey highlights the poten- tial and limitations of current systems, shedding light on future research directions.

Index Terms—Travel itinerary, AI, itinerary planning, tourism technology, generative AI.

I. INTRODUCTION

Computerized travel agenda arranging has changed the travel industry area by offering adaptable, continuous itinerary items [13] in light of client inclinations. Different frameworks mean to address the necessities of explorers for customized trip arranging. This paper looks at three particular methodologies: POI-based trip arranging frameworks, ChatGPT-driven schedule age, and a movement journal application zeroing in on local area and joint effort.

Every framework gives extraordinary highlights that take care of various voyager requests and use cases. In any case, holes in setting mindful customization, dynamic rethinking, and thorough information usage remain. This review assesses these frameworks in light of their capacity to convey proficient and customized travel agendas.

II. LITERATURE SURVEY

Two distinct but interrelated studies contribute significantly to the evolving landscape of travel itinerary[13] creation and recommendation systems. The first study delves into the application of Artificial Intelligence (AI) and Machine Learning (ML) techniques within the domain of personalized recommendations. It reviews various approaches that leverage user preferences and contextual information to optimize recommendations. Despite the advancements, a noticeable gap exists in exploring sophisticated neural network models and their potential to improve the personalization and accuracy of travel recommendations.

Conversely, the second study focuses on Reinforcement Learning (RL) frameworks applied to recommendation systems. It highlights how RL-based approaches, such as the Markov Decision Process (MDP), can dynamically adapt to user behavior, optimizing recommendations based on evolving user preferences and feedback. These methodologies address challenges such as information overload and decision fatigue, particularly within complex systems like travel planning platforms.

III. REVIEW OF ITINERARY PLANNING APPROACHES

A. ChatGPT-Driven Itinerary Generation

Katerina Volchek and Stanislav Ivanov explore ChatGPT's use as a travel itinerary planner. ChatGPT, leveraging generative AI, generates itineraries based on natural language prompts. This approach has benefits:

- **Ease of Use**: Tourists can request itineraries by specifying simple parameters.
- **Accessibility**: ChatGPT's web interface and mobile compatibility provide high accessibility.
- **Preliminary Planning**: ChatGPT[2] generates general ideas, suitable for initial inspiration.

B. Design of Exploring Automated Systems for Planning Travel Itineraries

Peilin Chen's work on computer-based intelligence driven travel schedule arranging frameworks presents an organized way to deal with making customized and proficient itinerary items [13]. The framework uses man-made brainpower to conquer the constraints of customary arranging appa- ratuses, which frequently need continuous updates and customization. The framework is planned with various

DOI: https://doi.org/10.71058/jodac.v9i8007

modules that cooperate to give an extensive travel arranging experience[1]. Striking highlights include:

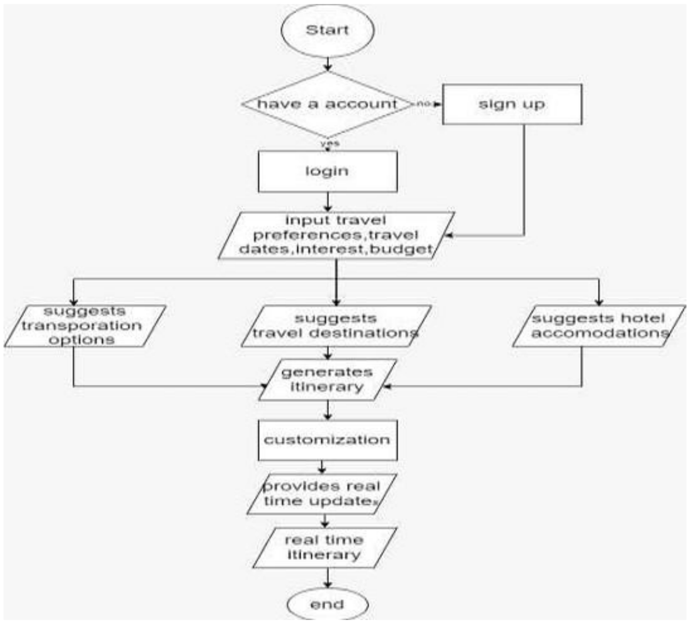

Fig. 1. FLOWCHART

- **User Preferences**: The system uses association rule mining, a technique inspired by shopping basket analysis, to analyze user behaviors and preferences, helping to make personalized travel recommendations.
- **Trip Planning Modules**: Includes various submodules such as finished product routes (pre-planned itineraries), self-service planning (allowing users to customize every aspect of their trip), and real-time route search using vertical search technology to gather updated travel information from major websites.
- **Intelligent Processing**: Uses a fuzzy weight algorithm to rate and recommend itineraries based on user preferences and mined association rules[14], delivering optimized travel plans.
- **System Performance**: Demonstrated stability and scalability through multi-threaded performance testing, effectively handling multiple concurrent users while balancing thread use for optimal performance.

A key limitation identified in the paper is the system's reliance on server-based database updates, which could affect the accuracy of real-time information, especially under high traffic loads. Future iterations could enhance real-time data handling and optimize system performance under extreme loads.

C. Personalized Itinerary Planning Systems

Senjuti Basu Roy, Gautam Das, Sihem Amer-Yahia, and Cong Yu's work on intelligent schedule arranging presents a framework that empowers voyagers to make custom schedules in view of criticism driven proposals of Focal points (POIs). Their framework tends to enter difficulties in customized schedule development, which regularly includes choosing significant POIs and enhancing course request inside a given time spending plan. Not at all like customary static schedules, this intuitive methodology powerfully adjusts proposals in view of client input, bringing about a profoundly customized travel insight. Prominent highlights include:

- **User Preferences**: Users rate each suggested POI as "yes" (interested) or "no" (not interested). The system leverages these preferences to filter and update future recommendations, allowing users to refine their itineraries gradually.
- **POI Selection**: The system uses a probabilistic model to suggest POIs that balance user preferences with practicality[11], such as proximity to other attractions and the user's time budget. This iterative approach allows users to see only the most relevant POIs without being overwhelmed by options.
- **Routing and Navigation**: The system calculates optimized routes, arranging the selected POIs in an order that fits within the user's time constraints. This process includes starting points, typically the user's accommodation, and stops when the route satisfies the time budget and user preferences.
- **Advanced Features**: To handle the complexity of personalized itinerary construction, the authors introduce heuristic algorithms that simplify the NP-complete problem of selecting and scoring itineraries. The scoring model adjusts based on user feedback, while a pruning strategy improves efficiency by prioritizing POIs that maximize expected itinerary [6] quality.

A limitation identified in the study is the reliance on predefined POI ratings and fixed time budgets, which could restrict the system's ability to adapt to real-time changes such as unforeseen travel delays or shifts in user preferences. Despite this, the system provides a robust foundation for enhancing the personalization of travel itineraries[12] through continuous user engagement.

D. Machine Learning-Based Travel Itinerary Systems

The exploration by Mrs. S. Sangeetha Mariammal and her group presents a Brilliant Travel Right hand that utilizes a half and half AI way to deal with give fitted proposals to vacationer locations and facilities in light of client inclinations. Planned explicitly for India, this framework pulls information from stages like TripAdvisor, Holidify, and Yatra to accumulate data on attractions, inns, and client audits. By incorporating AI calculations, it offers a customized, easy to use travel arranging experience with the accompanying eminent elements:

- **User Preferences**: The system gathers input on user-specific preferences such as types of attractions,

DOI: https://doi.org/10.71058/jodac.v9i8007

hotel budgets, and desired accommodations (e.g., free Wi-Fi, parking). This data helps shape itinerary recommendations aligned with the user's travel style and expectations.

– **POI Selection**: K-Means clustering is applied to group nearby POIs and filter options based on user location and interest. The K-Nearest Neighbor (KNN) algorithm refines hotel and attraction sugges- tions, enhancing the system's ability to recommend relevant travel options based on geographic proxim- ity and user interests.
– **Routing and Navigation**: The 2-Opt algorithm is used to optimize travel routes between POIs, mini- mizing travel distances and time while maximizing coverage of relevant attractions. Each itinerary is visualized on an interactive map using the Python Folium library, displaying POIs and the recom- mended travel path for each day[1].
– **Advanced Features**: This system includes an image recognition module that uses the VGG-16 deep learn- ing model to identify attractions from user-uploaded photos, enhancing the platform's functionality with transfer learning. A sentiment analysis component leverages logistic regression to analyze user reviews, adjusting future recommendations based on positive or negative feedback.

While successful in giving tweaked travel arranging, the ongoing framework doesn't yet offer unique continuous schedule changes, possibly restricting its adaptability because of abrupt changes or inclinations. Future bearings for this examination incorporate commercializing the stage, adding local area based highlights, for example, client produced touring websites, and creating devices for constant flexibility.

E. POI-Based Trip Planning Systems

Chang et al's. work on the Programmed Travel Schedule Arranging Framework (ATIPS)[10] presents a model that consequently creates venture out agendas custom-made to client inclinations. The framework permits clients to enter travel time, flight, and objective subtleties, and it plans schedules around famous objections while adjusting to individual interests. Prominent highlights include:

– **User Preferences**: Learns and adjusts itineraries based on user-defined preferences and prior travel history[1], ensuring personalized recommendations.
– **POI Selection**: Recommends attractions based on relevance, popularity, user preferences, and available amenities such as dining and accommodations.
– **Routing and Navigation**: Uses a greedy algorithm to create optimized routes by selecting each point of interest sequentially, aiming for the highest-rated attraction along the path.
– **Advanced Features**: Considers constraints like bud- get, travel distance, and time available for each stop, making it suitable for multi-day trips.
– **Feedback Mechanism**: Updates user preferences dynamically based on feedback from previous trips, which refines future itinerary[7] suggestions.

A notable limitation of ATIPS[10] is its lack of real-time adaptability; it does not adjust itineraries dynamically for unexpected changes such as sudden weather shifts or delays in travel time. [10]

F. Survey of Modern Itinerary Planning Applications

Jaiswal's survey on travel itinerary planning systems reviews current applications, highlighting the integration of modern technologies and personalization features. Key elements of these applications include:

– **User Personalization**: Many systems adjust itineraries based on user preferences, past travel data, and AI-based learning, creating highly relevant travel plans.
– **Real-Time Updates**: Proposed features include in- corporating live data such as weather forecasts, event schedules, and traffic updates to allow for responsive itinerary[7] adjustments.
– **Comprehensive Services**: Uses AI and machine learning for refined itinerary recommendations and chatbot-based user support, making itinerary plan- ning more interactive.
– **Feature Gaps**: Highlights the lack of real-time adaptability in many platforms, with room for im- provement in dynamically updating itineraries based on sudden changes.

The survey identifies limitations such as incomplete real-time data integration, leaving gaps in user experience when itinerary updates are needed for unexpected events or new preferences. [11]

G. Travel Route Optimization Based on Tourist Prefer- ences

Wangi et al. present a personalized travel route system that leverages a modified greedy algorithm to optimize travel itineraries from start to finish, maximizing effi- ciency and alignment with tourist preferences. The sys- tem is primarily intended for tourists exploring specific regions. Key features include:

– **User Preferences**: Customizes routes based on user-defined preferences and allotted travel time, provid- ing a tailored experience with flexibility in attrac- tions.
– **Optimized Routing**: Integrates Google Maps to dis- play routes visually, offering detailed information on travel time and distances between points of interest.
– **Personalization and Flexibility**: Allows users to select starting and ending points and accommodates the choice of attractions along the way.
– **Advanced Functionality**: Offers time-based route adjustments, with the system adapting route options to fit within the user's specified time frame.

DOI: https://doi.org/10.71058/jodac.v9i8007

A primary limitation noted is that while the routes are initially optimized, the system lacks real-time adaptability and does not dynamically adjust the route for unexpected changes, such as delays or new preferences encountered mid-route. [12]

H. TRIP-PAL: Travel Planning with Guarantees by Combining Large Language Models and Automated Planners

Tomas de la Rosa and colleagues' work on TRIP-PAL[4] in- troduces a hybrid model for travel planning that integrates Large Language Models (LLMs), like GPT-4, with auto- mated planners to ensure travel itineraries meet complex constraints and optimize user satisfaction. Recognizing the limitations of using LLMs alone for structured and constraint-bound travel planning, the authors developed a method that leverages both LLMs and automated planners to produce feasible, high-quality plans. Notable features include:

– **LLM-Driven Data Extraction**: The LLM gathers POIs, user ratings, and travel time estimations based on user input, ensuring that TRIP-PAL[4] starts with a rich data foundation formatted into structures that the planner can interpret.

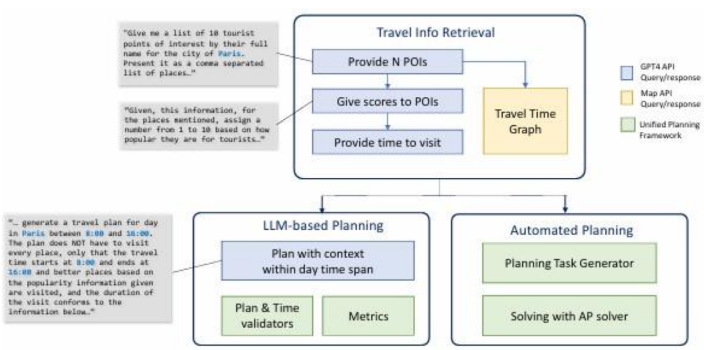

Fig. 2. LLM(TRIP-PAL)

– **Automated Planning**: Uses algorithms like A* with the LMCUT heuristic to construct plans that guarantee validity by adhering to user constraints, such as time limits, travel preferences, and specific POI requirements.
– **Oversubscription Planning**: Selects an optimal subset of POIs when time constraints prevent visiting all destinations, allowing users to visit the highest-priority locations within limited time.

The primary limitation noted in the study is the system's dependency on the initial quality of data parsed by the LLM. Since LLMs can occasionally generate inconsistent data, TRIP-PAL's[8] effectiveness could vary depending on the accuracy of the extracted information. Improvements in LLM training and data handling could further enhance TRIP-PAL's[8] performance.

I. A Review of Trip Planning Systems

WMK Tizani's 1992 review of trip planning systems (TPS) explores the early requirements and potential of real-time, multi-modal travel information systems. This paper examines both public and private transportation modes and advocates for integrated systems that would better serve travelers, transport operators, and policymakers. The review highlights the importance of integrating static and dynamic data, such as timetables and real-time traffic information, to improve trip planning[9] and reduce congestion. Key insights include:

– **User Needs**: Tizani emphasizes that travelers need comprehensive information that spans multiple transport modes, including public transit, personal vehicles, and alternative routes.
– **System Integration**: The paper advocates for a TPS that combines static data (such as timetables and routes) with dynamic data (including real-time traffic conditions, incidents, and delays) to provide optimal route choices and in-journey updates.
– **Potential Benefits**: The integrated system could increase public transport utilization and aid in traffic management, offering benefits to travelers in the form of better route options and reduced uncertainty, while also supporting public transport operators by increasing ridership.
– **Technology Limitations**: The review discusses the feasibility of implementing such a TPS given the state of early telecommunications and information technology of the time, highlighting challenges like the need for national support and standardization.

A huge impediment in Tizani's work is its emphasis on obsolete innovations[13], for example, Videotex, which restricts its immediate application in the present comput- erized setting. By the by, the paper's bits of knowledge into the significance of constant, multi-modular data frameworks stay applicable, highlighting the continuous requirement for incorporated, information driven trip arranging arrangements[10].

– **User Data Extraction**: Utilizes LLMs to gather Points of Interest (POIs), travel times, and preferences based on user input.
– **Constraint-Based Optimization**: Uses automated planners to sequence trips that satisfy user-defined constraints and maximize satisfaction.
– **Enhanced Utility**: Ensures optimized itineraries that provide higher user satisfaction and adaptability to real-world constraints.

This hybrid approach significantly outperformed LLMs alone, especially in complex travel scenarios. A noted challenge was ensuring scalability as the number of POIs increases [4].

DOI: https://doi.org/10.71058/jodac.v9i8007

J. Intelligent Travel Planning Insights Using Machine Learning

Karthik B. and Vignesh S. explore a machine learning-based travel planning system that uses collaborative filtering and genetic algorithms (GAs) to deliver tailored, data-driven itineraries. Their work highlights the challenges of traditional planning, such as time-consuming manual comparisons, and proposes a solution that learns user preferences to improve itinerary personalization. Notable features include:

- **Collaborative Filtering**: Analyzes user profiles to recommend destinations based on similarities with other travelers.
- **Genetic Algorithm Optimization**: Uses GAs to optimize travel plans for factors like time, budget, and user satisfaction.
- **Iterative Improvement**: Continuously refines recommendations through user feedback for improved accuracy.

This approach's potential to create highly personalized and adaptable travel experiences is promising, though it currently lacks real-time adjustments for changing user preferences [5].

K. Travel Itinerary Planning Using TSP and K Means Clustering

Kartika Nur Kholidah and partners fostered a movement schedule arranging application utilizing the Mobile Sales rep Issue (TSP) and k-implies grouping to improve travel courses regarding distance and time. The framework permits clients to choose wanted POIs and gatherings them into day to day groups to make an effective multi-day agenda. Key parts of their methodology include:

- **POI Clustering**: Uses k-means clustering to group nearby destinations into day-based clusters.
- **Route Optimization**: Solves TSP within each cluster to create a minimized route that reduces travel time.
- **Interactive Mapping**: Provides a map-based itinerary display, allowing users to see route efficiency and modify their plans.

While effective for small-to-medium scale itineraries, the application requires improvements in handling additional constraints and enhancing processing speed for larger POI sets [6].

However, limitations arise due to ChatGPT's reliance on pre-2021 data, meaning it lacks real-time updates on new POIs or temporary events. Furthermore, the absence of personalized scheduling, cost considerations, and context-based adjustments limits its effectiveness for detailed trip management [2].

L. The "Travel Diary" Community-Based Itinerary App

The Travel Diary app, as discussed by A. Padma Priya, combines AI-driven itinerary suggestions with a focus on social engagement and community support. The app's key features include:

- **AI-Powered Customization**: Provides personalized itineraries tailored to user interests.

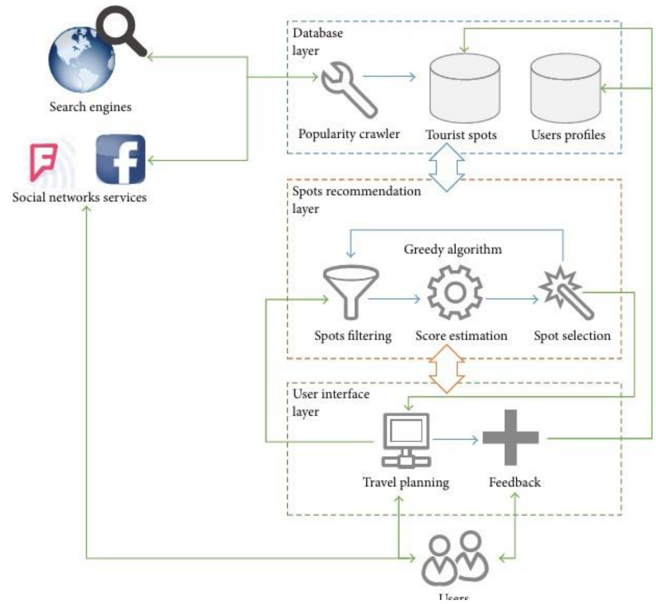

Fig. 3. AI-driven Data

- **Collaboration Tools**: Users can plan trips with friends in real-time.
- **Social Sharing**: Travel Diary enables users to share experiences, review destinations, and follow like-minded travelers.

While Travel Diary offers extensive user engagement, it requires active internet access and lacks reactive trip planning features. It also does not yet integrate contextual updates [3].

IV. Comparative Analysis

TABLE I
Comparison of Itinerary Planning Systems

Feature	POI-Based Systems	ChatGPT-Based Planning	Travel Diary
Customization	High	Moderate	High
Real-Time Adaptability	Limited	None	Limited
Community Engagement	Minimal	None	High
Dynamic Re-Planning	Lacking	None	Lacking
Cost & Time Management	Moderate	None	Moderate
Ease of Use	Moderate	High	High
Device Accessibility	Web-based	Web and mobile	Mobile app

Table I thinks about the qualities and shortcomings of each methodology in a few areas of interest, including flexibility, local area elements, and cost administration.

DOI: https://doi.org/10.71058/jodac.v9i8007

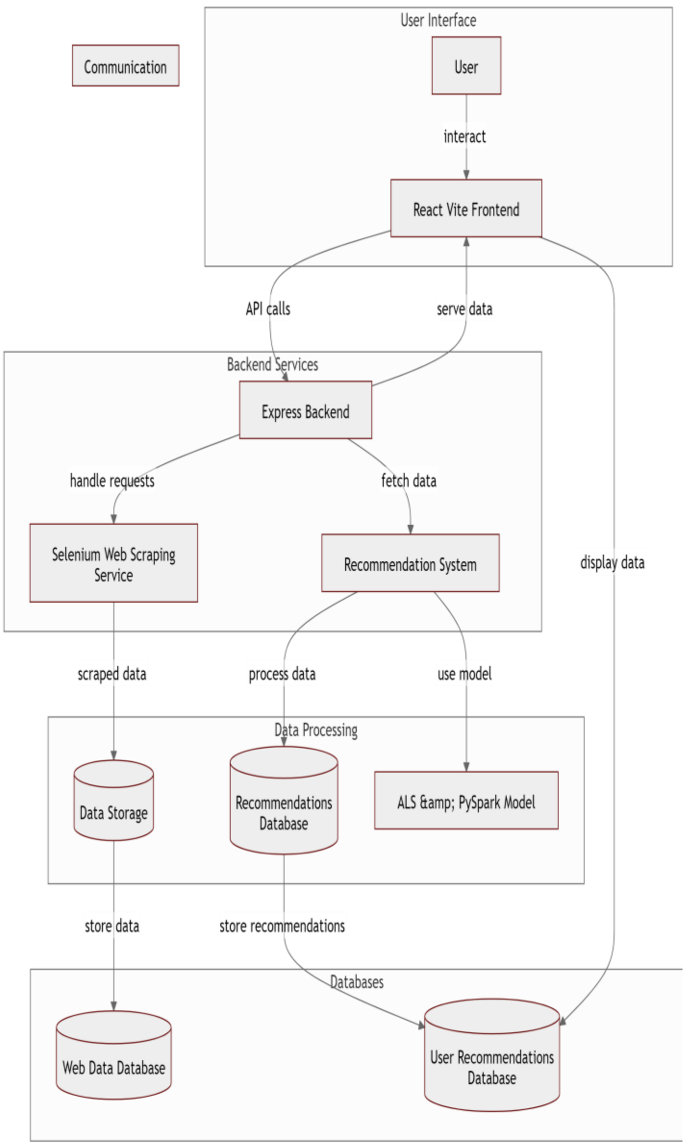

Fig. 4. System Architecture

V. System Architecture

A. Overview

The system architecture for the **Wander Wise Project** highlights the flow of data and interaction between the user, backend services, recommendation systems, and data storage modules. The design ensures modularity, scalability, and efficiency in generating personalized travel recommendations.

B. Explanation of Components

1) **User Interface:** The user interacts with the system via the React Vite Frontend, a highly responsive and intuitive interface designed to enhance the user experience. This frontend enables users to seamlessly input their preferences, view tailored recommendations, and provide valuable feedback.
The React Vite framework ensures fast loading times, smooth transitions, and a modular design, making it easier to scale and maintain the application. Key features of the frontend include interactive forms for capturing user preferences, dynamic recommendation displays that adapt to user input, and a feedback system to continuously refine the suggestions offered by the platform.
Communication between the frontend and backend is managed through well-structured and secure API calls. These APIs are designed to handle user queries efficiently, ensure real-time data synchronization, and maintain a seamless flow of information between the user interface and the backend logic.
By leveraging modern web technologies like React and Vite, the platform delivers an engaging and efficient user experience, driving both usability and satisfaction.

2) **Backend Services:** The Express Backend serves as the central processing hub, orchestrating various critical operations to ensure the system runs smoothly and efficiently. It is responsible for handling incoming user requests, coordinating with multiple services, and managing the flow of data throughout the application.
One of its primary functions is to facilitate data scraping, where it gathers up-to-date and relevant information from various external sources. This data is then cleaned, structured, and stored for subsequent processing. The backend also performs intensive data processing tasks, such as filtering, categorizing, and enriching the scraped data to make it suitable for generating high-quality recommendations.
Additionally, the Express Backend interacts seamlessly with the recommendation system, feeding it with the necessary input and retrieving tailored outputs for users. This involves implementing sophisticated algorithms to personalize results based on user preferences and feedback, ensuring accuracy and relevance.
Built on the Express framework, the backend is designed for scalability and reliability, with features such as robust error handling, middleware for seamless API integration, and optimized routes for faster performance. It also incorporates secure communication protocols to protect user data and maintain privacy across the entire platform.

3) **Selenium Web Scraping Service:** The Selenium Web Scraping Service plays a crucial role in gathering real-time travel-related data from a variety of external sources, such as accommodation platforms, activity listings, and travel deal websites. By automating web interactions, Selenium efficiently navigates through complex web pages, extracts valuable information, and ensures the data collected is accurate and up-to-date.
This service is designed to handle dynamic content, such as JavaScript-rendered pages, making it an ideal solution for scraping modern, interactive websites. It retrieves a wide range of data, including hotel availability, pricing, user reviews, popular attractions, events, and travel

DOI: https://doi.org/10.71058/jodac.v9i8007

itineraries.

Once the data is collected, the service processes and organizes it into a structured format. The cleaned and standardized data is then stored in the Web Data Database, which serves as a centralized repository. This database is optimized for quick access and integration with other components, such as the recommendation system and analytics engine.

The Selenium Web Scraping Service incorporates er- ror handling mechanisms to manage challenges like CAPTCHA verification, rate-limiting, and changes in website structure. Regular monitoring ensures the scraping scripts remain functional and up-to-date with any modifications in the target websites.

By leveraging Selenium's capabilities, the platform can continuously provide users with fresh, relevant, and comprehensive travel data, enhancing the quality and reliability of the recommendations offered.

4) **Recommendation System:** Powered by advanced algorithms such as **ALS (Alternating Least Squares)**, **PySpark**, and **LLMs (Large Language Models)**, the recommendation system is the core component that generates personalized travel suggestions for users.

The system leverages **ALS**, a matrix factorization technique, to identify hidden patterns in user preferences and travel-related data. By analyzing historical interactions and feedback, it can predict user interests with high accuracy. **PySpark**, a distributed computing framework, ensures scalability by processing large datasets efficiently, enabling the system to handle real-time queries even as the user base grows.

In addition to these traditional techniques, the integration of **LLMs** brings the power of natural language understanding and contextual awareness to the recommendation system. LLMs analyze textual data such as user reviews, travel blogs, and descriptions to extract meaningful insights and enhance the recommendations. The system continuously retrieves and updates data from the **Recommendations Database**, which stores processed user preferences, behavioral data, and travel-related information. By combining collaborative filtering, content-based filtering, and contextual insights, the system generates highly personalized itineraries, accommodation options, and activity suggestions tailored to individual needs.

Furthermore, the recommendation system incorporates feedback loops to refine its predictions over time, ensuring that the suggestions become increasingly relevant as users interact with the platform. This dynamic and data-driven approach guarantees an engaging and satisfying user experience.

5) **Data Processing:** The **Data Processing** module serves as a critical intermediary component, bridging the gap between raw scraped data and actionable insights used by the recommendation system. Its primary function is to clean, validate, transform, and enrich the data collected from external sources before feeding it into the downstream components.

This module starts by performing data cleaning, removing duplicates, handling missing values, and standardizing formats to ensure consistency and reliability. It then applies validation rules to filter out inaccurate or irrelevant information, guaranteeing high-quality inputs for the recommendation system.

After cleaning and validation, the module transforms the data into a structured format suitable for analysis. This includes converting unstructured web-scraped data into tabular or JSON formats, categorizing data points (e.g., accommodations, activities, user reviews), and enriching it with additional attributes, such as geolocation data or sentiment analysis from textual reviews.

The processed data is then integrated into the **Recommendations Database**, where it becomes accessible for real-time queries by the recommendation system. The module also interacts with other storage systems, such as the **Web Data Database**, ensuring that raw and processed datasets are properly archived for future use.

In addition to its core tasks, the data processing module incorporates scalable pipelines powered by frameworks like **Apache Spark** and **Pandas**, enabling efficient handling of large volumes of data. By automating these processes, it ensures that the platform remains responsive and capable of delivering up-to-date and relevant recommendations to users.

6) **Databases:** The platform utilizes a well-structured database architecture to ensure efficient storage, retrieval, and management of diverse datasets critical for its operation. These databases work together to support seamless functionality and scalability.

- **Web Data Database:** This database serves as the repository for raw data collected through the **Selenium Web Scraping Service**. It organizes unprocessed data such as accommodation details, activity options, travel deals, user reviews, and event information. The data is stored in its raw form for traceability and future reprocessing needs. Indexing mechanisms are employed to enable quick access to specific datasets, and backups are maintained to ensure data integrity.
- **Recommendations Database:** The **Recommendations Database** stores processed and structured data that has been curated by the **Data Processing** module. This includes enriched travel-related information, categorized data points, and features extracted for use in the recommendation system. Optimized for fast querying, this database supports efficient real-time interaction with the **Recommendation System**, ensuring that users receive personalized and relevant suggestions with minimal latency.
- **User Recommendations Database:** This database focuses on user-specific data, maintaining personal-

DOI: https://doi.org/10.71058/jodac.v9i8007

ized recommendations, user feedback, and interaction histories. Feedback loops leverage this data to refine recommendation algorithms and improve accuracy over time. It also tracks user preferences and engagement metrics, enabling adaptive personalization. The database is designed with a privacy-first approach, ensuring secure storage and compliance with data protection regulations.

7) **Feedback Loop:** The **Feedback Loop** is a critical component of the system, designed to enhance the accuracy and personalization of recommendations by incorporating user feedback into the learning process. This iterative mechanism ensures that the system evolves and adapts to user preferences over time, delivering increasingly relevant and satisfying results.

   When users interact with the platform—such as by selecting recommendations, providing ratings, or submitting explicit feedback—their actions are captured and stored in the **User Recommendations Database**. This feedback is then analyzed to identify patterns, preferences, and areas for improvement.

   Advanced techniques, such as sentiment analysis, implicit feedback modeling, and collaborative filtering, are employed to process both explicit and implicit feedback. For example, a user's preference for certain travel activities or destinations can be inferred from their engagement patterns, even if explicit ratings are not provided.

   The processed feedback is used to update the underlying algorithms, such as **ALS** and **LLMs**, enabling them to refine their predictions and adapt to new trends. The system also uses feedback to detect and correct potential biases or inaccuracies in the recommendation models, ensuring fairness and diversity in the suggestions.

   This continuous feedback integration not only improves individual user experiences but also enhances the overall quality of the platform. By leveraging a dynamic and user-centered approach, the feedback loop helps maintain the platform's relevance, accuracy, and competitiveness in the travel industry.

This architecture integrates technologies such as **React Vite**, **Express.js**, **Selenium**, and machine learning models, ensuring a seamless workflow and robust travel itinerary generation.

C. System Workflow

The sequence diagram illustrates the process of generating and delivering personalized recommendations in the system. Below is a step-by-step explanation of each component and interaction:

- **User**: The end user interacts with the application.
- **React Vite**: Serves as the front-end framework, enabling users to access the application interface.
- **Express Server**: Acts as the backend, orchestrating requests and processes between components.
- **Data Storage**: Stores the scraped and processed data for further use.

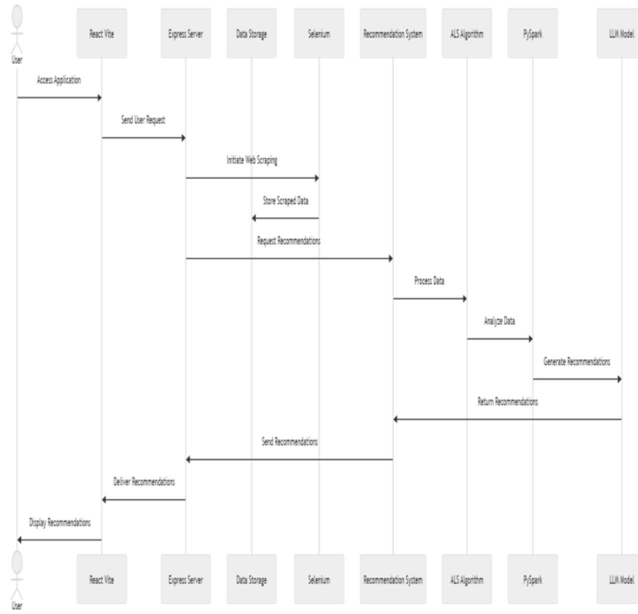

Fig. 5. Sequence Diagram

- **Selenium**: A web scraper used to collect external data required for recommendations.
- **Recommendation System**: Manages the orchestration of recommendation generation.
- **ALS Algorithm**: Implements collaborative filtering to provide personalized suggestions.
- **PySpark**: Handles large-scale data analysis and processing.
- **LLM Model**: Utilizes AI to generate detailed and personalized recommendations.

Workflow

1) The **User** accesses the application through the **React Vite** front end.
2) **React Vite** sends the user request to the **Express Server**.
3) The **Express Server** initiates web scraping through **Selenium**.
4) **Selenium** collects data and stores it in **Data Storage**.
5) The **Express Server** requests recommendations from the **Recommendation System**.
6) The **Recommendation System** processes data using:
   - The **ALS Algorithm** for collaborative filtering.
   - **PySpark** for large-scale data processing.
   - The **LLM Model** for generating recommendations.
7) The **LLM Model** returns recommendations to the **Recommendation System**.
8) The **Express Server** delivers recommendations back to **React Vite**.
9) **React Vite** displays the recommendations to the **User**.

DOI: https://doi.org/10.71058/jodac.v9i8007

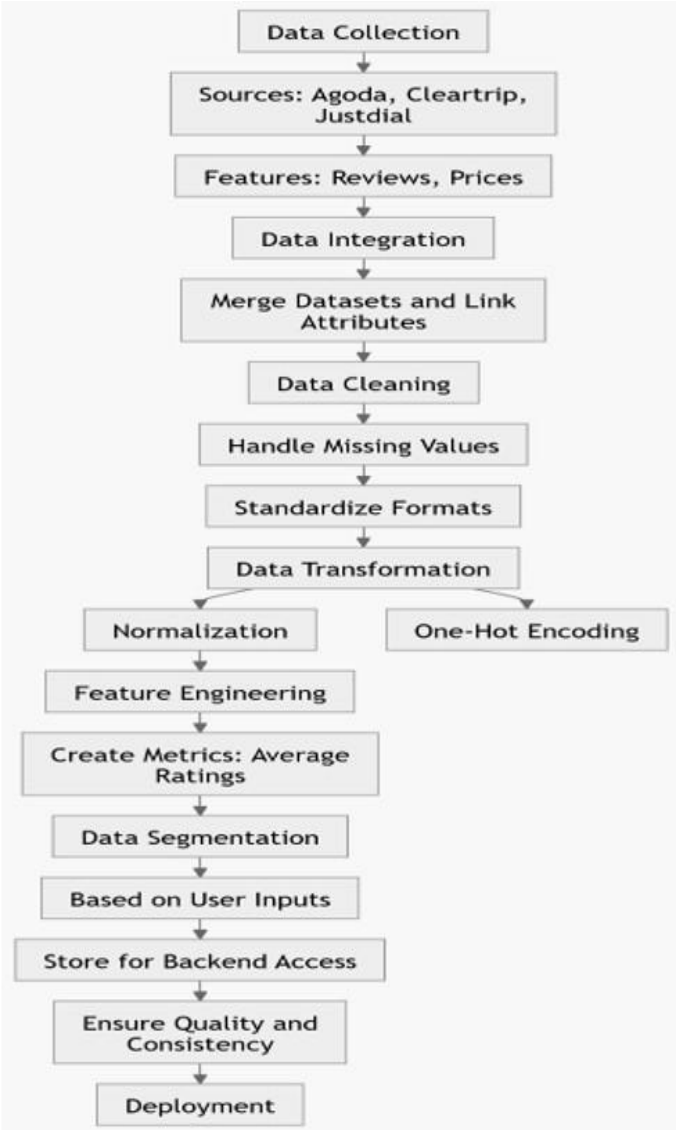

Fig. 6. Flow Diagram of Preprocessing

VI. Implementation Details

Explanation

The Figure 3 represents the preprocessing flow for the Wander Wise project. Here is the step-by-step explanation of each component:

1) **Data Collection**: Collect data from multiple sources, including Agoda for hotels, Cleartrip for flights, and Justdial for rental vehicles. Gather features such as reviews, prices, and amenities from these platforms
2) **Data Integration**: Merge data sets from different sources to create a unified structure. Link attributes such as locations and ratings to ensure consistency and connectivity between datasets.
3) **Data Cleaning**: Handle missing values to ensure that there are no gaps in critical data points. Standardize formats such as dates, currencies, and text fields to maintain uniformity.Data Transformation: Perform normalization to scale numerical values into a consistent range for easier processing.
4) **Feature Engineering**: Create new metrics such as average ratings and cost-efficiency scores for better insights. Extract and derive relevant features to enhance the recommendation system's accuracy.
5) **Data segmentation**: Segment the data based on user input, such as destination, duration of the trip, and preferences. Prepare subsets for specific modules, such as hotels, restaurants, and attractions.
6) **Store for Backend Access**:Store the processed and segmented data in a database (e.g., Azure) for dynamic and real-time access by the application backend.
7) **Ensure Quality and Consistency**:Validate the processed data to ensure that it meets the quality standards and aligns with the system's requirements.
8) **Deployment**:The preprocessed data is ready for deployment to power the system's recommendation and planning features.This flow ensures the collected raw data is cleaned, structured, and optimized for efficient use within the Wander Wise application.

A. Data Collection and Integration

Hotel Data (Scraped from Agoda)

- **Details Scraped:** The system collects a comprehensive set of data points from Agoda to ensure the most relevant and useful information is available for users. The details include:
  – **Hotel Names:** Accurate identification of hotels for clear and reliable recommendations.
  – **Locations:** Geographic data, including addresses and proximity to key landmarks or attractions.
  – **Prices Per Night:** Current pricing for various room options, ensuring budget-friendly suggestions.
  – **Star Ratings:** Official ratings to indicate the quality and service standards of each hotel.
  – **User Reviews:** Aggregated customer feedback, including ratings and comments, to assess hotel satisfaction and reliability.
  – **Available Amenities:** A detailed list of amenities such as Wi-Fi, complimentary breakfast, parking, pools, and fitness centers to match user preferences.
  – **Booking Links:** Direct links to Agoda booking pages for seamless user redirection and faster reser- vation processes.
- **Purpose:** The scraped hotel data serves as a foundational component for delivering highly personalized recommendations to users. The system utilizes this data to:
  – Match hotels to user-defined preferences, including budget, required amenities, and travel group size.
  – Prioritize recommendations based on proximity to attractions, events, or other user-specified locations.

DOI: https://doi.org/10.71058/jodac.v9i8007

- Enable real-time filtering and sorting of hotels based on criteria like star ratings, price ranges, or user reviews.
- Provide users with diverse options by showcasing hotels across various price tiers and quality levels.

By leveraging detailed hotel data, the platform ensures users receive accurate, relevant, and convenient accommodation options that enhance their overall travel experience.

Restaurant Data (Scraped Using Google Maps API)

- **Details Scraped:** The platform utilizes the **Google Maps API** to gather a rich dataset of restaurant information, ensuring users receive comprehensive dining recommendations. The details scraped include:
  - **Restaurant Names:** Accurate and complete names for easy identification and reference.
  - **Cuisine Types:** Categories such as Italian, Chinese, Indian, Vegan, and more to cater to diverse culinary preferences.
  - **Average Costs:** Approximate pricing information per meal or person, helping users make budget- conscious choices.
  - **Ratings:** Aggregated star ratings based on user feedback to assess the quality and popularity of each restaurant.
  - **User Reviews:** Detailed comments and feedback from previous diners, offering insights into service, ambiance, and food quality.
  - **Geolocations:** Latitude and longitude coordinates for accurate mapping and proximity-based filtering.
  - **Operating Hours:** Opening and closing times to ensure availability during the user's preferred dining hours.
  - **Contact Details:** Phone numbers and websites for easy communication and reservation.
- **Purpose:** The restaurant data serves multiple purposes aimed at enhancing user convenience and dining experiences. Key objectives include:
  - Providing an interactive map-based interface where users can explore nearby dining options visually.
  - Allowing users to filter and sort restaurants based on criteria like cuisine type, budget, ratings, and proximity.
  - Catering to specific dietary preferences or restrictions, such as vegetarian, vegan, or gluten-free options.
  - Enabling seamless trip planning by integrating dining suggestions with nearby attractions or accommodations.
  - Offering real-time updates on restaurant availability, such as changes in operating hours or temporary closures.
  - Delivering a personalized dining experience by factoring in user preferences and historical feedback.

By utilizing the Google Maps API, the platform ensures accurate and real-time restaurant data, empowering users to discover and enjoy the best dining options wherever they go.

Bus Data (Scraped from Goibibo)

- **Details Scraped:** The platform gathers detailed and structured data from **Goibibo** to provide users with a variety of bus travel options. The information scraped includes:
  - **Bus Names:** Names or operators of buses to help users identify their preferred service providers.
  - **Types of Buses:** Categories such as sleeper, semi-sleeper, luxury, and non-AC buses to cater to varying comfort and travel preferences.
  - **Ticket Prices:** Pricing details for different seating or berth options, enabling users to choose based on their budget.
  - **Amenities:** Information on available amenities such as Wi-Fi, air conditioning, charging points, blankets, and recliner seats for a comfortable journey.
  - **Departure and Arrival Times:** Precise timings to help users plan their trips efficiently.
  - **Boarding and Drop-off Points:** Specific locations for boarding and alighting to ensure convenience and proximity to user preferences.
  - **User Ratings and Reviews:** Feedback from past passengers, providing insights into service quality, punctuality, and overall satisfaction.
  - **Journey Duration:** Estimated travel times to help users plan their schedules effectively.
- **Purpose:** The bus data serves as a crucial component for recommending suitable intercity travel options that align with user preferences. The primary objectives include:
  - Offering a curated list of buses tailored to user-defined requirements, such as budget, amenities, and travel time.
  - Allowing users to filter options based on criteria like type of bus, operator reputation, and departure/arrival timings.
  - Enhancing travel convenience by displaying nearby boarding and drop-off points in relation to the user's current location or destination.
  - Integrating with other travel components, such as accommodations and attractions, to provide end-to-end travel planning.
  - Providing a personalized experience by prioritizing recommendations based on historical preferences and feedback.
  - Ensuring real-time updates on seat availability, price changes, and potential delays, empowering users to make informed decisions.

By leveraging data from Goibibo, the platform delivers reliable and comprehensive bus travel options, ensuring users have a seamless and comfortable intercity travel experience.

DOI: https://doi.org/10.71058/jodac.v9i8007

Flight Data (Scraped from Cleartrip)

- **Details Scraped:** The platform collects a comprehensive range of flight data from **Cleartrip** to provide users with optimal flight options based on their travel preferences. The data scraped includes:
  - **Flight Names:** Identification of the specific flight services, including flight numbers, to ensure accurate recommendations.
  - **Airlines:** Information on the airline providers, allowing users to choose based on preferred airlines, loyalty programs, or brand reputation.
  - **Departure and Arrival Times:** Exact times for both departure and arrival, helping users manage their schedules and plan trips effectively.
  - **Ticket Prices:** Pricing details for different seat classes, providing users with transparent options that suit their budget.
  - **Layovers:** Information on layover durations and locations, helping users decide on direct or connecting flights based on convenience and time efficiency.
  - **Seat Class Options:** Availability of different seat classes (e.g., economy, business, first class) to meet users' comfort preferences.
  - **Baggage Allowance:** Details on carry-on and checked baggage allowances to aid in decision-making, especially for users with specific luggage requirements.
  - **Flight Duration:** Total flight duration, including layovers, to help users select the most time-efficient flights.
- **Purpose:** The flight data serves to deliver highly personalized, cost-effective, and efficient flight options to users. The main goals include:
  - Offering flight recommendations that align with user-defined travel dates, budget constraints, and destination preferences.
  - Allowing users to filter and sort flights by factors such as price, flight duration, departure time, and airline.
  - Helping users optimize their travel plans by providing alternatives for flights with shorter layovers, better timing, or reduced costs.
  - Providing clear, up-to-date information about baggage policies and seat options to avoid confusion during the booking process.
  - Enabling users to compare different flight offerings from multiple airlines to choose the most suitable options.
  - Integrating with other travel components like accommodations and transport options to offer a seamless end-to-end travel planning experience.

  By leveraging data from Cleartrip, the platform ensures that users receive reliable and comprehensive flight options, empowering them to make informed decisions and enjoy a smooth travel experience.

Rental Vehicle Data (Scraped from Justdial)

- **Details Scraped:** The platform gathers detailed rental vehicle data from **Justdial** to provide users with convenient options for local or intercity travel. The data scraped includes:
  - **Vehicle Types:** A range of vehicle categories including cars (compact, sedan, SUVs), bikes (scooters, motorcycles), and specialty vehicles (luxury cars, minibuses) to cater to different travel needs.
  - **Rental Prices:** Pricing information for various vehicle types and rental durations (daily, weekly, monthly), allowing users to choose vehicles based on their budget.
  - **Availability:** Real-time information on the availability of vehicles, ensuring that users can access up-to-date options for their desired rental period.
  - **Pickup and Drop-off Locations:** Detailed location information, including city, neighborhood, or specific points (airports, train stations), for user convenience when planning the start and end of their rental journey.
  - **User Ratings:** Reviews and ratings from past customers, helping users assess the quality of the rental service, vehicle condition, and overall satisfaction.
  - **Vehicle Features:** Specific details about vehicle amenities, such as air conditioning, GPS, child seats, and fuel options (petrol/diesel/electric), allowing users to choose vehicles that meet their needs.
  - **Insurance and Deposit Details:** Information on rental insurance options, damage coverage, and security deposit requirements for added transparency and trustworthiness.
- **Purpose:** The rental vehicle data serves to deliver flexible and convenient travel options by recommending suitable vehicles for various travel scenarios. The primary goals include:
  - Offering personalized recommendations based on user-defined preferences, such as budget, vehicle type, and rental duration.
  - Allowing users to filter and sort vehicles by criteria like price, vehicle type, availability, and ratings for optimized decision-making.
  - Ensuring convenience by providing real-time availability and options for nearby pickup and drop-off locations, especially for users traveling to or from key transportation hubs.
  - Facilitating intercity and local travel planning by providing options for one-way rentals, long-distance travel, and short-term usage.
  - Empowering users with clear, transparent pricing information, including insurance coverage and deposit details, to avoid unexpected costs.
  - Integrating with other travel components, such as accommodations and transport, to provide a seamless travel experience across different segments.

DOI: https://doi.org/10.71058/jodac.v9i8007

By using data scraped from Justdial, the platform ensures that users receive a variety of reliable and well-priced rental vehicle options, enhancing flexibility, comfort, and convenience throughout their travels.

These datasets, obtained through extensive web scraping, enable **Wander Wise** to offer comprehensive, real-time travel planning. This ensures that users receive accurate recommendations across multiple travel domains, enhancing their overall experience.

B. Algorithms

*1) **Algorithm 1: Flight Recommendation System**:* This algorithm processes flight data to recommend flights based on user-defined weights for price, duration, and non-stop preferences. It also filters flights by a specified departure time range.

1) Load JSON data containing flight details into a pandas DataFrame.
2) Clean and preprocess the data:
   - Parse the price column to remove currency symbols and commas, converting it to an integer.
   - Convert the flight duration to total minutes.
   - Add binary encoding for stops: non-stop = 1, otherwise = 0.
3) Normalize the features (price, duration, non-stop) for scoring.
4) Define user-defined weights for price, duration, and non-stop preferences.
5) Compute a weighted score for each flight using the formula:
   score = weight_price * (1 - price_norm) + weight_duration * (1 - duration_norm) + weight_non_stop * non_stop_norm.
6) Convert departure times to datetime for filtering.
7) Filter flights within the user-specified time range (e.g., 6:00 AM to 12:00 PM).
8) Sort the filtered flights by the computed score in descending order.
9) Save the recommended flights as a JSON file and print the top recommendations.

*2) **Algorithm 2: Hotel Recommendation System Using ALS**:* This algorithm uses collaborative filtering with ALS (Alternating Least Squares) to recommend hotels to users based on their ratings and selected features.

1) Initialize a Spark session.
2) Load the hotel reviews and features datasets from JSON files into Spark DataFrames.
3) Preprocess the data:
   - Cast the Rating column to float.
   - Use StringIndexer to create numeric indices for UserID and Hotel_Name.
4) Join the review and feature datasets on the Hotel_Name column.
5) Filter hotels based on user-selected features (e.g., Internet services, Elevator).
6) Train the ALS model:
   - Use UserID_index as the user column.
   - Use Hotel_index as the item column.
   - Use Rating as the rating column.
7) Generate top N hotel recommendations for each user using the trained ALS model.
8) Explode the recommendations into a readable format and map indices back to user and hotel names.
9) Calculate the average predicted rating for each hotel and sort by the predicted rating in descending order.
10) Save the top hotel recommendations as a JSON file and print them to the console.

C. Frontend and Backend Integration

The integration of the frontend and backend for WanderWise ensures seamless interaction between users and the recommendation system. Below are the details of the integration process:

Frontend: React Vite

- **Frontend Framework**: React Vite is used for building the user interface, ensuring fast rendering and a smooth user experience.
- **API Integration**: Axios is utilized for sending HTTP requests to the backend.
- **User Interface**: Interactive components enable users to input preferences, view recommendations, and filter results dynamically.
- **State Management**: React Context API or Redux handles application state efficiently.

Backend: Express Server

- **Server Framework**: Express is used to build the backend server, providing RESTful API endpoints for communication with the frontend.
- **Data Processing**: Handles user requests and forwards them to the recommendation system.
- **Middleware**: Implements middleware for authentication, input validation, and error handling.
- **CORS Handling**: Ensures secure cross-origin resource sharing between the frontend and backend.

Integration Workflow

1) The frontend sends user preferences and input data to the backend via HTTP requests.
2) The backend processes the requests, queries the database or recommendation engine, and applies business logic.
3) Processed data and recommendations are sent back to the frontend.
4) The frontend dynamically updates the UI to display results to the user.

Benefits of Integration

- **Improved User Experience**: Fast, responsive UI with real-time updates.
- **Scalability**: Modular design ensures easy addition of new features.

DOI: https://doi.org/10.71058/jodac.v9i8007

- **Robust Communication**: Secure and efficient data exchange between the frontend and backend.

VII. Deployment

A. Infrastructure

The system is deployed on AWS, utilizing services such as EC2 for hosting and S3 for data storage. Docker containers ensure scalability and reliability.

B. Continuous Integration and Delivery

A CI/CD pipeline is established using GitHub Actions, automating testing and deployment processes.

VIII. Results

Automated itinerary planning systems present varied solutions for trip management, each with strengths that cater to specific user needs. POI-based systems excel in personalizing attractions and routes but lack adaptability. ChatGPT[2] offers a user-friendly platform for inspiration but needs validation for real-world use. Community-based apps like Travel Diary foster engagement and collaboration but lack dynamic adaptability.

IX. Future Directions and Research Opportunities

The discoveries show key regions where future examination can concentration to further develop mechanized agenda frameworks by consolidating trend setting innovations and systems:

- **Reactive Re-Planning**: Enhancing itinerary adaptability to real-time changes, such as unexpected closures or weather shifts, could improve the user experience. Implementing context-aware algorithms that dynamically adjust schedules based on real-time information would allow itineraries to stay relevant and convenient for travelers.
- **Personalized Group Itineraries**: Developing systems that can cater to diverse interests within group itineraries is essential for enhancing user satisfaction. Future research could explore collaborative filtering techniques that accommodate individual preferences within a group context, allowing systems to generate personalized itineraries for sub-groups with distinct preferences within a single itinerary plan.
- **Enhanced AI Recommendations with RBM and ALS**: To improve recommendation quality, future systems could incorporate deep learning models such as Restricted Boltzmann Machines (RBM) and Matrix Factorization techniques like Alternating Least Squares (ALS). RBM could help uncover latent features of user preferences by learning complex patterns in user behavior, thereby enabling more accurate predictions of interest in POIs. Matrix Factorization with ALS, often used in recommendation systems, could efficiently handle large-scale user-POI matrices, allowing the model to learn optimal latent factors for each user and POI. This approach would enable a more precise matching of users to POIs, enhancing the relevance of itinerary recommendations.
- **Building Interactive Frontends with React and Vite**: Future schedule organizers can use Respond alongside Vite, a quick improvement construct instrument, to make responsive, powerful UIs. Respond's part based engineering joined with Vite's superior presentation construct cycle can empower designers to convey quick stacking, intelligent, and easy to use web applications, fundamental for a consistent schedule arranging experience.
- **Integrating Google Maps API for Location-Based Services**: Using the Google Guides Programming interface can improve travel arranging applications by giving continuous geolocation, steering, and planning administrations. Joining with Google Guides permits clients to see and interface with POIs, get bearings, and envision courses inside the application, making agenda arranging instinctive and intelligent.

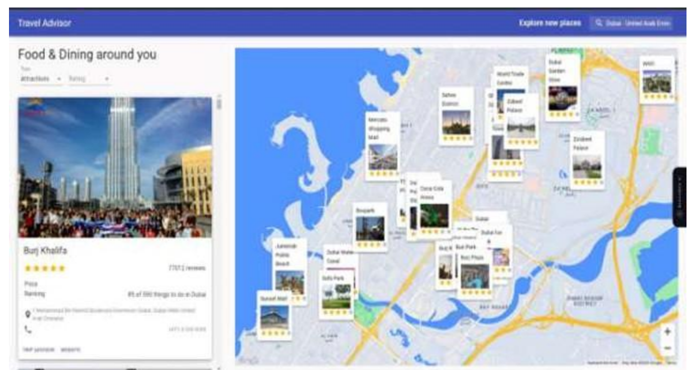

Fig. 7. MAPS

- **Django for Robust Backend Services**: For secure and scalable backend development, Django offers a powerful framework that supports efficient data handling and user authentication. Using Django, developers can manage user data, provide secure login options, and handle backend logic for itinerary generation, offering a robust server-side foundation for frontend applications.
- **Scalable Deployment with AWS**: Deploying itinerary planning applications on cloud platforms like AWS ensures scalability, reliability, and performance. By leveraging AWS services such as EC2 for hosting, S3 for storage, and RDS for database management, applications can accommodate large user bases and manage high traffic volumes, ensuring consistent availability and performance.
- **Sustainable Tourism Options**: As travelers become more eco-conscious, itinerary systems could incorporate eco-friendly options such as sustainable POIs, low-impact transportation routes, and environmentally conscious accommodation options. Leveraging AI to recommend sustainable choices that align with travelers' environmental values can help promote responsible tourism and reduce the travel industry's environmental impact.

By integrating these advancements, future itinerary planning systems could offer a more robust, adaptive, and user-centered

DOI: https://doi.org/10.71058/jodac.v9i8007

experience, meeting the evolving demands of modern travelers.

X. Conclusion

Automated itinerary planning systems present varied solutions for trip management, each with strengths that cater to specific user needs. POI-based systems excel in personalizing attractions and routes but lack adaptability. ChatGPT [2] offers a user-friendly platform for inspiration but needs validation for real-world use. Community-based apps like Travel Diary foster engagement and collaboration but lack dynamic adaptability. Integrating advanced AI and real-time data could yield more reliable, adaptable systems. The increasing demand for personalized, adaptable, and socially connected trip planning systems highlights the potential of these tools to transform travel planning.

CONFLICT OF INTEREST

On behalf of all authors, the corresponding author states that there is no conflict of interest.

- Competing Interests: Not Applicable

- Funding Information: Not Applicable

- Research Involving Human and /or Animals: Not Applicable

- Informed Consent: Not Applicable

References

[1] K. Sylejmani and A. Dika, ”A Survey on Tourist Trip Planning Systems,” *International Journal of Arts & Sciences*, 2011.

[2] K. Volchek and S. Ivanov, ”ChatGPT as a Travel Itinerary Planner,” *ENTER 2024*, Deggendorf Institute of Technology, 2024.

[3] A. P. Priya, ”An Automated Itinerary Planning and Trip Management System,” *International Journal of Creative Research Thoughts*, vol. 12, no. 5, 2024.

[4] T. de la Rosa, S. Gopalakrishnan, A. Pozanco, Z. Zeng, and D. Borrajo, ”TRIP-PAL: Travel Planning with Guarantees by Combining Large Language Models and Automated Planners,” *J.P. Morgan AI Research*, 2024.

[5] K. B. and V. S., ”Intelligent Travel Planning Insights Using Machine Learning,” *International Journal of Creative Research Thoughts*, vol. 12, no. 4, 2024.

[6] K. N. Kholidah, S. Rani, and S. N. Huda present ”The Development of a Travel Itinerary Planning Application Utilizing the Traveling Salesman Problem and K-Means Clustering Approach” (Department of Informatics, Islamic University of Indonesia, 2024).

[7] P. Chen discusses the ”Design of a Travel Itinerary Planning System Based on Artificial Intelligence” in *Journal of Physics: Conference Series*, vol. 1533, no. 3, 2020.

[8] T. dela Rosa, S. Gopalakrishnan, A. Pozanco, Z. Zeng, and D. Borrajo, ”TRIP-PAL: Travel Planning with Guarantees by Combining Large Language Models and Automated Planners,” *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

[9] W. M. K. Tizani, ”A Review of Trip Planning Systems,” *Working Paper 373, Institute of Transport Studies*, University of Leeds, 1992.

[10] H.-T. Chang, Y.-M. Chang, and M.-T. Tsai introduce ”ATIPS: An Automatic Travel Itinerary Planning System for Domestic Areas.”

[11] H. Jaiswal, ”Survey Paper on Travel Itinerary Planning Systems,” *International Journal of Advances in Engineering and Management*, vol. 5, no. 11, pp. 142–149, 2023.

[12] V. H. Wangi, J. T. Beng, and Wasino, ”Start to End: Recommended Travel Routes Based on Tourist Preference,” *IOP Conference Series: Materials Science and Engineering*, vol. 852, 012163, 2020.

[13] S. Basu Roy, G. Das, S. Amer-Yahia, and C. Yu, ”Interactive Itinerary Planning,” *Proceedings of the VLDB Endowment*, vol. 4, no. 2, pp. 175–185, 2011.

[14] S. S. Mariammal, S. B. Akshaya, M. Priyanga, S. Saran Kumar,

DOI: https://doi.org/10.71058/jodac.v9i8007

and P. Prakash, ”Smart Travel Assistant with Itinerary Planner Using Hybrid Machine Learning Approach,” *International Research Journal of Modernization in Engineering, Technology and Science*, vol. 4, no. 5, pp. 983–989, 2022.