TRANSFORMATIVE TOUR PLANNING WITH DYNAMIC ROUTE OPTMIZATION

A PROJECT REPORT

Submitted by

BANGARRU SRIJESH M 211420104034

BARATHAN K 211420104035

DHAYANAND G B 211420104062

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

APRIL 2024

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report **"TRANSFORMATIVE TOUR PLANNING WITH DYNAMIC ROUTE OPTMIZATION"** is the bonafide work of **BANGARRU SRIJSESH M (211420104034), BARATHAN K (211420104035) & DHAYANAND G B (211420104062)** who carried out the project work under my supervision.

Signature of the HOD with date	Signature of the Supervisor with date
Dr L. JABASHEELA M.E., Ph.D., **Professor and Head,** Department of Computer Science and Engineering, Panimalar Engineering College, Chennai - 123	**Dr N. PUGHAZENDI M.E., Ph.D.,** **Professor,** Department of Computer Science and Engineering, Panimalar Engineering College, Chennai - 123

Submitted for the Project Viva – Voce examination held on ____________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION BY THE STUDENT

We BANGARRU SRIJESH M [211420104034], BARATHAN K [211420104035], DHAYANAND G.B [211420104062] hereby declare that this project report titled “Transformative Tour Planning with Dynamic Route Optimization”, under the guidance of Dr N. PUGHAZENDI is the original work done by us and we have not plagiarized or submitted to any other degree in any university by us.

ACKNOWLEDGEMENT

Our profound gratitude is directed towards our esteemed Secretary and Correspondent, **Dr. P. CHINNADURAI, M.A., Ph.D**., for his benevolent words and fervent encouragement. His inspirational support proved instrumental in galvanizing our efforts, ultimately contributing significantly to the successful completion of this project.

We want to express our deep gratitude to our Directors, **Tmt. C. VIJAYARAJESWARI, Dr. C. SAKTHI KUMAR, M.E., Ph.D., and Dr. SARANYASREE SAKTHI KUMAR, B.E., M.B.A., Ph.D.,** for graciously affording us the essential resources and facilities for undertaking this project.

Our gratitude is also extended to our Principal, **Dr. K. MANI, M.E., Ph.D.,** whose facilitation proved pivotal in the successful completion of this project.

We express my heartfelt thanks to **Dr. L. JABASHEELA, M.E., Ph.D.,** Head of the Department of Computer Science and Engineering, for granting the necessary facilities that contributed to the timely and successful completion of project.

We would like to express our sincere thanks to our parents, friends, project coordinator and guide **Dr. N. PUGHAZENDI, M.E., Ph.D.,** and all the faculty members of the Department of CSE for their unwavering support for the successful completion of the project.

BARATHAN K

BANGARRU SRIJESH M

DHAYANANAD G B

ABSTRACT

Travel and tourism have a big effect on society and the economy. They facilitate cultural exchange, open doors to new job opportunities, boost local businesses, and play a pivotal role in economic development. However, making a perfect tour plan can be time-consuming, involving processes like researching destinations and accommodations, budget constraints, and intricate daily itineraries. To get over the hump this system will address the problem by offering the best tour plan within the budget limit and providing the flexibility to make changes or updates to the plan during the journey. With our vast dataset at your fingertips, you have the power to prioritize the places that intrigue you the most. It's all about tailoring your journey to your desires. This system weaves together your preferences, budgets, precious time, your residence's proximity, and the ideal mode of transport with all this our system effortlessly devises the finest possible plans. The heart of our model is developed with the distance matrix and advanced tsp algorithm, with this deep-learning recommendation algorithm the plans are materialized. The system employs the Flutter framework and firebase for providing the user interface, database services, functionality, and integration with the AI model. It utilizes the Directions Api to display routes for places as per plans using Google Maps. As travel is about making people feel extraordinary, our system places the control firmly in the hands of the traveler offering the flexibility to shape their journeys as they see fit. These meticulously crafted plans are here to elevate your every journey and make it unforgettable. Embark on a seamless journey where every detail aligns with your desires, transforming ordinary trips into extraordinary adventures. Let our innovative system be your guide, empowering you to create memories that last a lifetime. Experience the freedom to explore, discover, and create cherished moments, as our system crafts personalized itineraries tailored to your unique travel aspirations. Unleash the potential of your journey with precision planning and unparalleled flexibility at your fingertips.

TABLE CONTENTS

LIST OF TABLES

LIST OF ABBREVIATION

S.NO	ABBREVIATION	DEFINITION
1	TSP	Travelling Salesman Problem
2	UML	Unified Modeling Language
3	DFD	Data Flow Diagram
4	SDG	Sustainable Development Goals

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Our innovative travel planning system application revolutionizes the way you experience the world. Powered by advanced technology including the distance matrix and advanced TSP algorithm, Flutter framework, Firebase, python flask with Gemini model, we seamlessly weave together your preferences, budget, and time constraints to craft personalized itineraries. With the flexibility to make changes on the go and access to real-time route information through Google Maps integration, every journey becomes an unforgettable adventure. Whether you're seeking cultural immersion, local exploration, or simply relaxation, our system empowers you to shape your travels and create cherished memories with ease. Our groundbreaking travel planning system harnesses the power of cutting-edge algorithms and seamless integration to redefine the way you explore the world. From intricate daily itineraries to budget-conscious recommendations, we prioritize your desires, ensuring each journey is tailor-made to perfection. With real-time updates and unparalleled flexibility, embark on a transformative adventure where every moment is crafted to exceed expectations. Our cutting-edge travel planning system harnesses advanced algorithms, Flutter framework, firebase and Gemini model to craft personalized itineraries tailored to your preferences, budget, and time constraints. With real-time route optimization through Google Maps integration and the flexibility to make changes on the go, every journey becomes a transformative adventure, empowering you to explore the world with ease and create cherished memories. With the flexibility to make changes and updates on the fly, you remain in control of your itinerary, adapting it to unexpected opportunities or preferences as they arise. Every detail, from accommodation options to local attractions, is meticulously curated to elevate your travel experience and create lasting memories.

1.2 PROBLEM DEFINITION

Traditional travel planning processes often proven to be:

- Time-consuming
- requiring extensive research
- budget considerations and
- intricate itinerary planning

This presents a significant barrier for individuals seeking seamless and personalized travel experiences. To address this challenge, our project aims to develop an innovative travel planning system that leverages advanced algorithms and technology to streamline the planning process, providing users with tailored itineraries, real-time updates, and unparalleled flexibility, ultimately transforming ordinary trips into extraordinary adventures. the lack of integration between various travel planning tools often results in disjointed experiences and missed opportunities for exploration. Our system seeks to bridge this gap by offering a unified platform that seamlessly integrates with popular services like Google Maps and leverages cloud-based infrastructure for scalability and reliability. By empowering users to effortlessly customize their journeys while on the move, our solution aims to revolutionize the way people plan, experience, and cherish their travels. With a focus on user empowerment and enhanced convenience, our project endeavors to redefine the travel planning experience, making it accessible, efficient, and truly unforgettable for adventurers worldwide.

CHAPTER 2

LITERATURE SURVERY

Wu-Chih Hu1, et al. [1] The proposed system for logistics route planning leverages advanced technologies and algorithms to optimize transportation efficiency in response to the booming e-commerce industry. It utilizes Google Maps' "multiple destination" function to search for routes and employs a multilayer perceptron model to simulate traffic conditions. Additionally, Dijkstra's algorithm is applied to identify the optimal route based on vehicle speed. Key features include the ability to consider traffic condition prediction, calculate optimal routes with multiple destinations, and enable real-time tracking of vehicles. Experimental implementation confirms the system's feasibility and effectiveness in reducing idle driving and improving transportation efficiency, making it suitable for real-world logistics operations.

Advantages: The proposed logistics route planning system utilizes advanced algorithms and technologies to optimize transportation efficiency. By integrating Google Maps and predictive traffic models, it accurately calculates optimal routes, leading to reduced operational costs and enhanced efficiency.

Disadvantages: Integration with external services like Google Maps could introduce dependency issues, leading to disruptions if the service experiences downtime or changes its APIs.

Xiao Zhou, et al. [2] stated that a tourist sight empty vector is designed to store optimal tourist sights based on individual preferences, with temporary accommodation in downtown areas serving as clustering centers for optimal sight extraction. The algorithm employs motive iteration intervals and sub-intervals to develop a one-way shortest path algorithm, determining the shortest routes between clustering centers and tourist sights, as well as between different sights. Positive and negative motive benefit impact factors are considered, generating motive iteration values through iterative functions.

Advantages: Personalized Tour Planning will be considered as individual preferences by using temporary accommodations as clustering centers, ensuring tourists receive tailored sightseeing recommendations.

Disadvantages: Complexity of Implementation and maintaining the algorithm, particularly the motive iteration values and closed-loop structures, may require advanced technical expertise and computational resources, posing challenges for smaller tourism organizations.

H. H. Owaied, et al. [3] proposed a paper that introduces a framework for an intelligent tourism guiding system, designed as a knowledge-based system inspired by the behavior of human tour guides. The model aims to emulate the role of a human guide, providing visitors with personalized and professional service to meet their needs and enhance their tour experience. Comprising five modules including user interface, inference engine, knowledge base, dynamic database, and GIS and XRM application, the system offers comprehensive support for local, regional, and international guiding needs. It is suggested that mobile telephone companies could offer this system as a service for their customers, leveraging its capabilities to enhance tourism experiences.

Advantages: The proposed intelligent tourism guiding system is its ability to provide personalized and professional guidance to visitors. By mimicking the role of a human tour guide and leveraging a knowledge-based approach, the system can tailor information and recommendations to meet the individual needs and preferences of each visitor.

Disadvantage: A potential disadvantage of the intelligent tourism guiding system is its reliance on technology, which may lead to limitations in providing the human touch and nuanced interactions that a human tour guide can offer

Maobin Ding, et al. [4] proposed a paper that highlights the significant role of tourism in the social and economic landscape, emphasizing its potential to drive GDP growth and enhance overall satisfaction and experience for travelers. Leveraging technologies like sensor networks and artificial intelligence (AI), the tourism industry aims to provide accurate information, enrich travel experiences, and improve service quality and efficiency. Big data analysis enables businesses to understand tourist preferences, market demands, and trends, facilitating targeted marketing strategies and personalized experiences.

Advantages: Real-time data from sensor networks provide valuable insights into tourist behavior, allowing businesses to adapt and enhance their offerings accordingly. By leveraging these technologies, the tourism industry can deliver high-quality services consistently, promote sustainable growth, and stay competitive in an evolving market landscape.

Disadvantages: With the increased reliance on digital technologies and data sharing, there is a heightened risk of data breaches and cyberattacks, potentially compromising sensitive tourist information and damaging the reputation of tourism businesses.

Takashi Hasuike, et al. [5] paper introduces a personal tour planning problem that considers general tour routes and tourist satisfaction for sightseeing places based on different tourism purposes. To address the challenge of representing time-dependent models using static network models, the Time-Expanded Network (TEN) is proposed. TEN copies the set of nodes in the underlying static network for each discrete time step, transforming the problem into a classical static network flow problem. The model is formulated as a bi-objective 0-1 integer programming problem, aiming to maximize total satisfaction value while minimizing deviation from the best plan of standard tour routes.

Advantages: The proposed personal tour planning model offers several advantages. By incorporating time-dependent satisfactions and activity durations, it provides a more realistic and tailored approach to tour route optimization, enhancing the overall satisfaction of tourists The introduction of the Time-Expanded Network (TEN) simplifies the representation of time-dependent models, facilitating the application of existing combinatorial optimization and soft computing algorithms.

Disadvantages: Despite its merits, the personal tour planning model may face some limitations. The reliance on integer programming and combinatorial optimization algorithms could lead to computational complexity and scalability issues, particularly for large-scale tour planning problems or real-time applications

Feiran Huang, et al. [6] presented a Multi-task Deep Travel Route Planning framework (MDTRP) aimed at enhancing travel route planning by integrating various auxiliary information. Unlike existing methods focusing on specific planning tasks, MDTRP offers flexibility across different tasks, such as next-point recommendation, general route planning, and must-visit planning. By constructing a heterogeneous network and employing a deep model with attention mechanisms, the framework effectively incorporates Point of Interests (POI) attributes, user preferences, and historical route data.

Advantages: The Multi-task Deep Travel Route Planning framework (MDTRP) offers several advantages. By integrating rich auxiliary information such as Point of Interests (POI) attributes, user preferences, and historical route data, MDTRP enhances the effectiveness of travel route planning.

Disadvantages: The complexity of the heterogeneous network and deep model could result in increased computational overhead, particularly for large datasets or real-time planning scenarios.

Shou-Chih Lo, et al. [7] presents a novel approach for mobile tour applications, focusing on tour schedule optimization and route smoothness. The proposed scan-based algorithm efficiently arranges the visiting sequence of points of interest while considering route smoothness, enhancing the overall tour experience. Additionally, a distributed shortest path algorithm is introduced to maintain time-changing traffic data regionally, ensuring accurate and up-to-date route planning.

Advantages: The proposed scan-based algorithm considers both visiting sequence and route smoothness, resulting in more efficient and enjoyable tour schedules for tourists.

Disadvantages: Implementing scan-based and distributed shortest path algorithms may introduce computational overhead, potentially impacting the performance of mobile tour applications on resource-constrained devices.

Yuan Yao, et al. [8] addresses the challenge of multi-objective route planning in smart cities, specifically focusing on the integration of safety considerations alongside distance optimization.

Existing methods often overlook safety as a critical criterion, leading to suboptimal navigation decisions. To address this, the proposed multi-objective hyper-heuristic (MOHH) framework employs reinforcement learning mechanisms to enhance route selection. By designing low-level heuristics and leveraging parallel processing, the RL-PMOHH algorithm significantly improves computational efficiency compared to traditional methods.

Advantages: The proposed multi-objective hyper-heuristic (MOHH) framework prioritizes safety in walking route planning, providing users with routes that minimize crime risks while also optimizing for distance.

Disadvantages: Complexity in implementing and fine-tuning the MOHH framework may require advanced technical expertise in reinforcement learning and optimization algorithms, which could pose challenges for some users or researchers.

Kewad Rathod, et al. [9] highlights the increasing availability of information on the internet and the challenges tourists face in selecting suitable destinations amidst the vast options provided by web search engines and tourism sites. The project aims to address this issue by designing an intelligent AI-Based Tourist Guide system algorithm. This system seeks to eliminate the need for a middleman, such as a tourist guide, by providing a comprehensive application where users can independently plan and arrange their trips based on their individual preferences and needs.

Advantages: The AI-Based Tourist Guide system allows users to plan and arrange their trips independently, without relying on a middleman or tourist guide.

Disadvantages: The system's effectiveness depends on the accuracy and reliability of the AI algorithms and data sources, which could be susceptible to errors or biase

CHAPTER 3
SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing application stands as a comprehensive travel companion, offering a myriad of features to assist travelers in planning, booking, and experiencing their journeys. At its core, the app harnesses the collective wisdom of millions of users, providing a vast repository of reviews and ratings for accommodations, dining options, attractions, and activities worldwide. This wealth of user-generated content empowers travelers to make informed decisions tailored to their preferences and needs. Through intuitive search and filtering tools, users can efficiently discover and explore destinations, refining their options based on criteria such as price, location, and amenities.

- User Reviews and Ratings: One of the core features of the application is its extensive collection of user-generated reviews and ratings for hotels, restaurants, attractions, and activities. Users can read feedback from fellow travelers to make informed decisions about where to stay, dine, and explore.
- Search and Filtering: The app allows users to search for specific destinations, accommodations, restaurants, or attractions. Users can apply various filters such as price range, star rating, cuisine type, and more to refine their search results and find options that match their preferences.
- Booking: It enables users to book hotels, flights, restaurants, and experiences directly through the app. It partners with various booking platforms to provide seamless booking experiences for users.

3.1.1 DISADVANTAGES

There are some potential disadvantages to consider:

- Overwhelming Amount of Data: The vast amount of data on hotels, lodges, destinations, and transportation options could potentially overwhelm users, making it difficult for them to make decisions.

- Reliability of User Reviews: Aggregating user reviews is valuable, but there's always a risk of biased or inaccurate reviews that could mislead users.
- Privacy and Security Concerns: Collecting user data for bookings and reviews raises privacy concerns, and ensuring the security of this data is crucial to maintaining user trust.
- Limited Customization: While the platform offers tailored itineraries, some users may prefer more customization options to truly personalize their travel plans.
- Potential for Information Overload: Providing detailed information on destinations and activities is beneficial, but it could also overwhelm users with too much information, making it challenging for them to narrow down their choices.
- Inaccuracy of Estimated Prices: Estimated ticket prices for flights and trains may not always be accurate, leading to discrepancies between the expected and actual costs of travel.
- Competitive Market: The travel industry is highly competitive, with many similar platforms offering similar services. Standing out and attracting users in such a crowded market can be challenging.
- Dependency on Internet Connectivity: Users may face difficulties accessing the platform and making bookings if they have limited or no internet connectivity while traveling.

Addressing these disadvantages through user-friendly interfaces, transparent communication, reliable data sources, and robust privacy measures can help mitigate potential issues and enhance the overall user experience.

3.2 PROPOSED SYSTEM

Our proposed travel planning system is a comprehensive solution designed to simplify and enhance the travel experience for users. Leveraging advanced technologies and robust data integration, the system offers a seamless platform for itinerary creation, accommodation booking, and destination exploration.

- Ensuring Best time of attraction and seasonal journey for travelers.
- Crafting tour plans based on a user's singular preference.
- Best tour guide for devotional travelers.
- Customization and modification on tour plans.

It offers users a seamless platform to create personalized itineraries, select accommodations, and explore destinations. Key features include intuitive interface, personalized itinerary generation, accommodation selection, destination exploration, user reviews, transportation recommendations, basic tour planning, and real-time updates. By leveraging advanced technology and robust data integration, our system aims to simplify the travel planning process and empower users to create unforgettable experiences. Our travel planning system offers a user-friendly interface for personalized itinerary creation, accommodation booking, and destination exploration. Key features include intuitive navigation, personalized itineraries based on preferences and budget, streamlined accommodation booking, comprehensive destination information with user reviews, transportation recommendations, basic tour planning, and real-time updates. Designed to simplify and enhance the travel experience, our system aims to empower users to create unforgettable journeys with ease.

3.2.1 ADVANTAGES

The travel planning system offers several advantages for users:

- Personalized Itineraries: By leveraging user preferences and budget constraints, the system can generate customized itineraries tailored to each traveler's unique interests and needs, ensuring a more fulfilling travel experience.
- User-Friendly Interface: The intuitive interface makes it easy for users to navigate the platform, create itineraries, and book accommodations, enhancing the overall user experience and satisfaction.
- Basic Tour Planning: The system's basic tour planning functionality allows users to craft tour plans based on their preferences, enabling them to explore destinations in a structured and organized manner.
- Customization and Modification: Users have the flexibility to customize and modify their tour plans as needed, accommodating changes in preferences or unforeseen circumstances, ensuring greater flexibility and control over their travel experiences.

CHAPTER 4
SYSTEM DESIGN

4.1 IMPLEMENTATION ENVIRONMENT

The implementation of tour planning system involves the following:

Client-Side: Users interact with the application through a mobile application. The front-end interface built using flutter frameworks using dart and python presents options for inputting constraints such as budget, time availability, and transportation preferences. Users submit their preferences, triggering requests to the server for personalized tour plans.

Server-Side: The backend server, developed using Python with frameworks like Flask, receives user requests and processes them. It utilizes algorithms such as the Traveling Salesman Problem (TSP) with the Gemini Model to generate personalized tour plans based on user constraints. Distance calculations and transportation options are obtained through Google Maps API. The server communicates with the database, which uses firebase data storage, to store and retrieve user data, preferences, and generated tour plans.

4.2 INPUT DESIGN

Software Requirement:

- Operating System: Windows 10
- IDE: Visual Studio Code
- Framework: Flutter Framework
- Maps API: Google Maps Platform
- API Backend: Firebase

Hardware Requirement:

- Processor: i5 6th gen or higher/ Ryzen 5 1600 or higher
- Graphics: Integrated Graphics or any External Graphics
- RAM: 8 GB or higher
- Hard Disk: 10GB for installation
- Proper Internet Connectivity

4.3 SYSTEM ARCHITECTURE

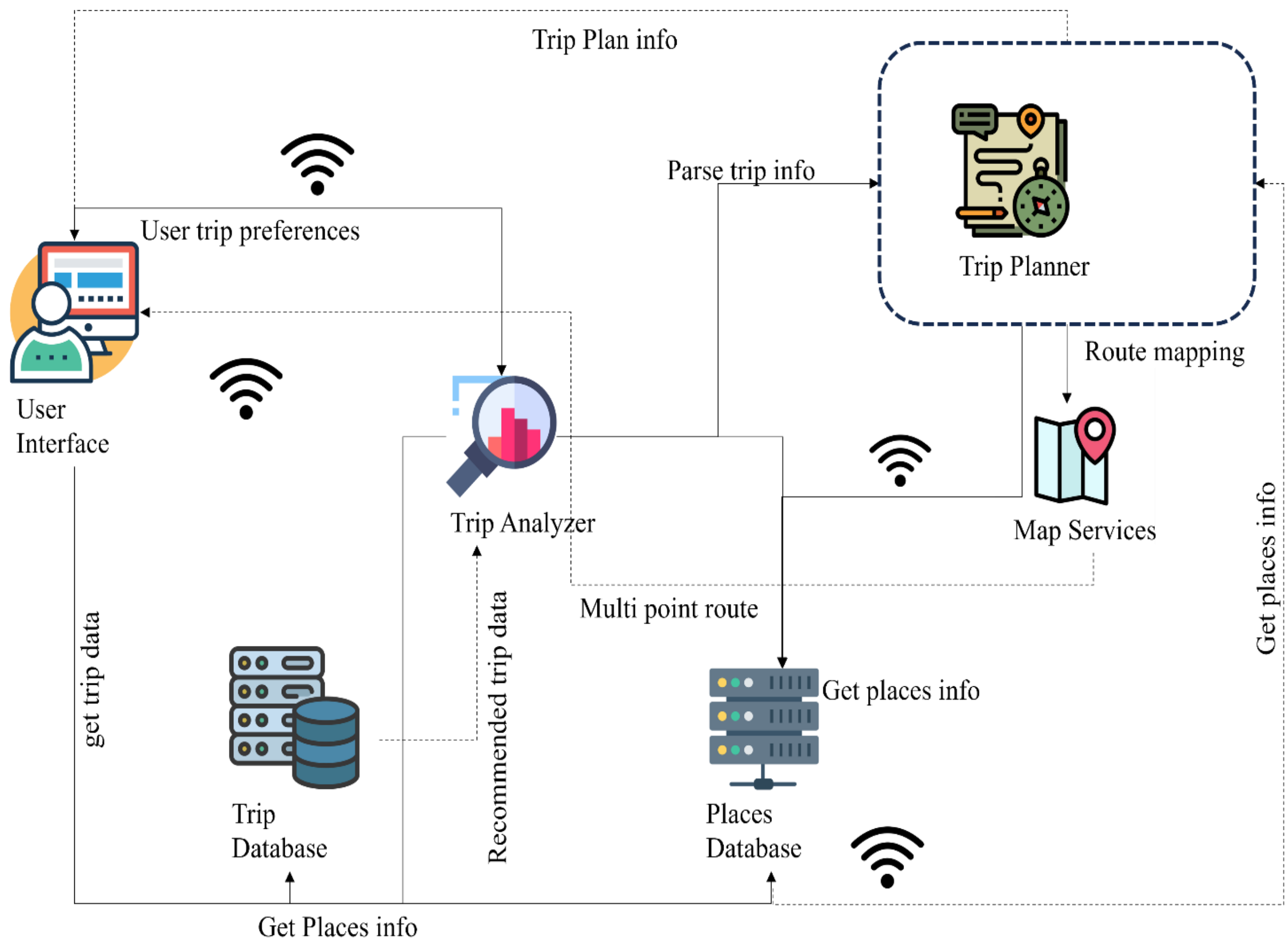


Fig 4.3 Architecture diagram for Tour Planning application

Figure 4.3 consists of several interconnected components that work seamlessly to provide users with a personalized and efficient experience. At the core lies the backend server, responsible for processing user requests, managing data, and communicating with external services. The front-end interface, whether web-based or mobile, interacts with the backend to display information and handle user inputs. A trip planner utilizes algorithms to generate custom itineraries based on user preferences and available resources.

4.4 UML DIAGRAMS

4.4.1 Use Case Diagram

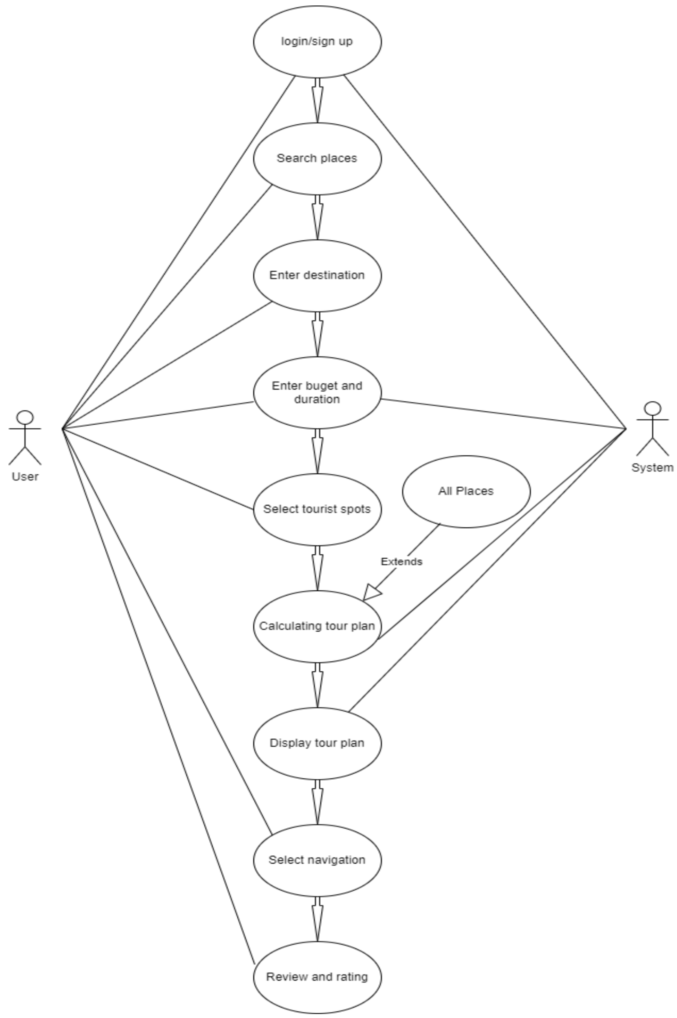


Fig 4.4.1 Use Case Diagram for Tour planning application

This use case diagram refers to activities done by user and system and their corresponding use cases. The diagram provides a comprehensive overview of the interactions and functionalities within the tour application, enhancing clarity and understanding for stakeholders.

4.4.2 Class Diagram

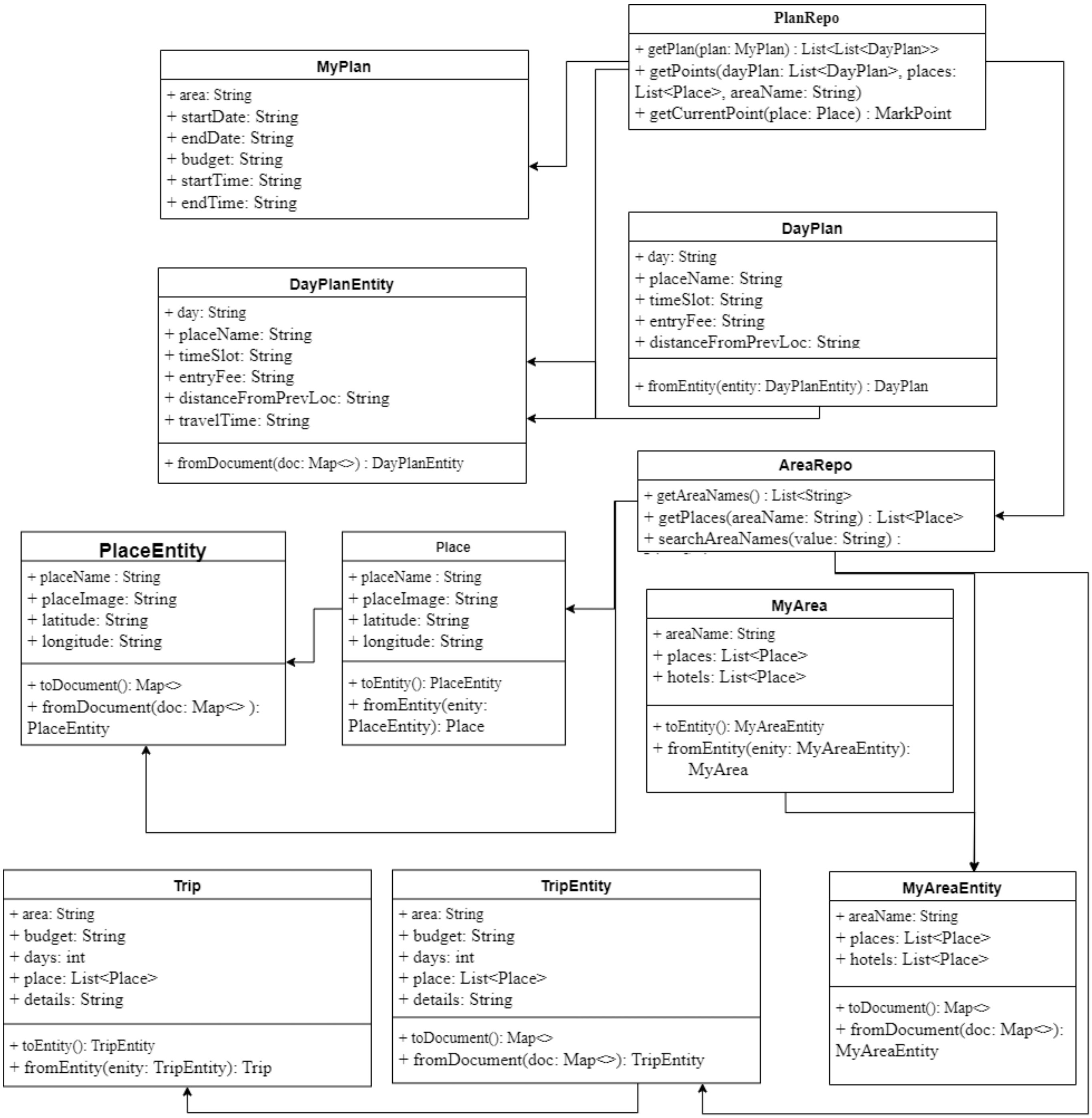


Fig 4.4.2 Class Diagram for Tour planning application

The Class diagram refers to relationships between different classes such as user repo, trip repo, plan repo and area repo.

4.4.3 Sequence Diagram

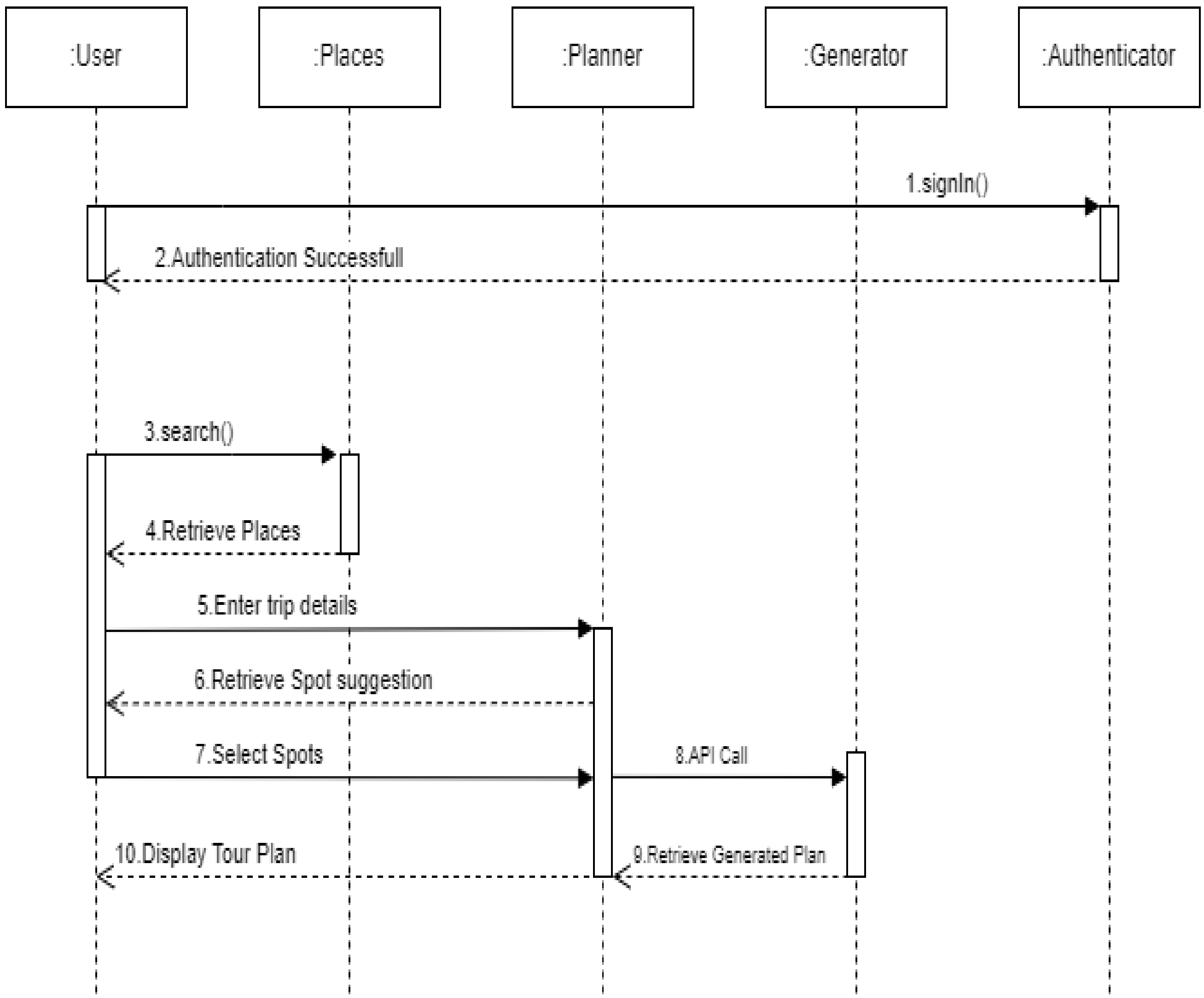


Fig 4.4.3 Sequence Diagram for Tour planning application

In the sequence diagram of our tour planning application system, the sequence of activities between the user and the system is depicted with clarity and precision. It illustrates the flow of interactions starting from the user's initiation of actions, such as searching for destinations or selecting preferences, followed by the system's responses and subsequent actions.

4.4.4 Collaboration diagram

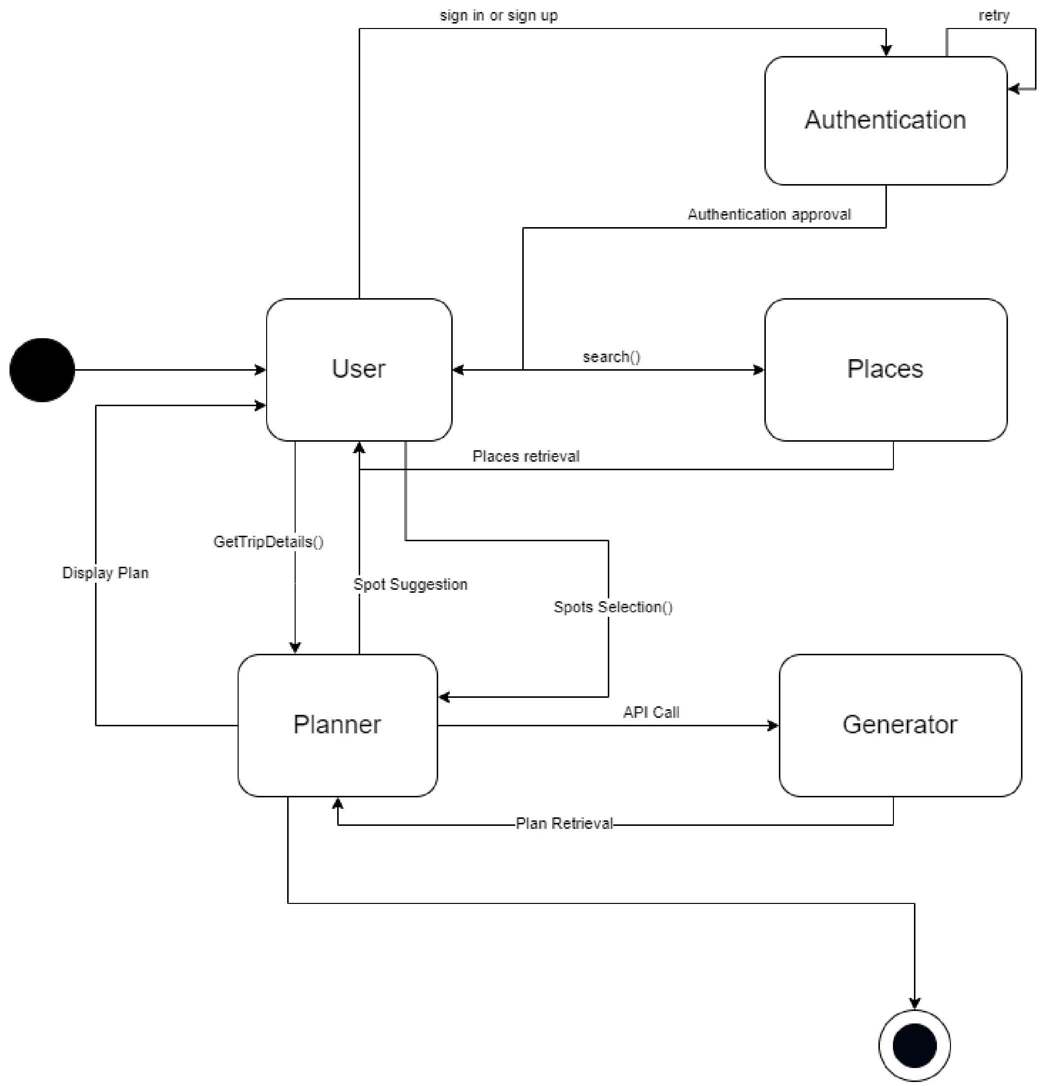


Fig 4.4.4 Collaboration diagram for Tour planning application

The collaboration diagram of tour planning system shows the sequence of activities of user interaction. In the collaboration diagram for our Tour Planning Application, the sequence of activities during user interaction is visually depicted, highlighting the collaboration between various components of the system. The diagram illustrates how different entities within the system, such as the user interface, plan generator, places etc. interact to facilitate the user's journey through the tour planning process.

4.4.5 Activity diagram

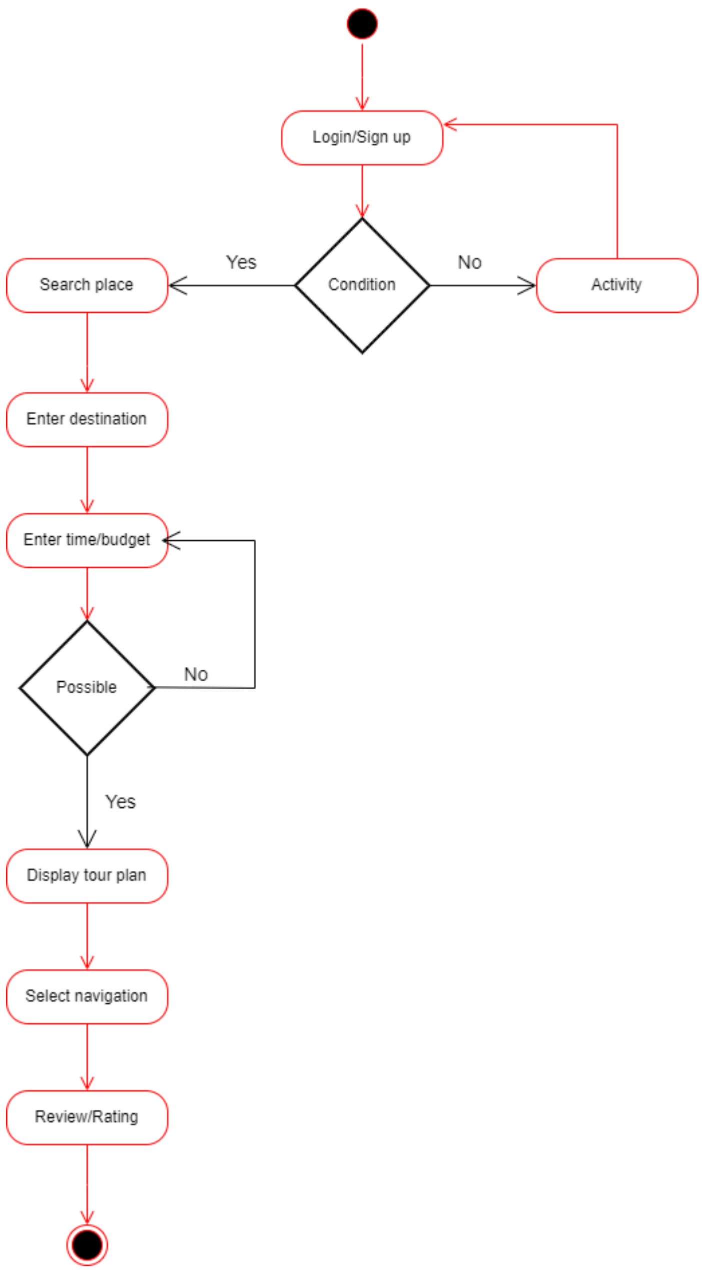


Fig 4.4.5 Activity Diagram for Tour planning application

The activity diagram of planning system shows the flow of activities in the application. The diagram also captures decision points and alternative paths, reflecting the flexibility of the application to accommodate different user choices and preferences. Overall, the activity diagram offers a structured visualization of the tour planning workflow, enabling users to understand the sequence of actions and the system's response at each step.

4.4.6 Component Diagram

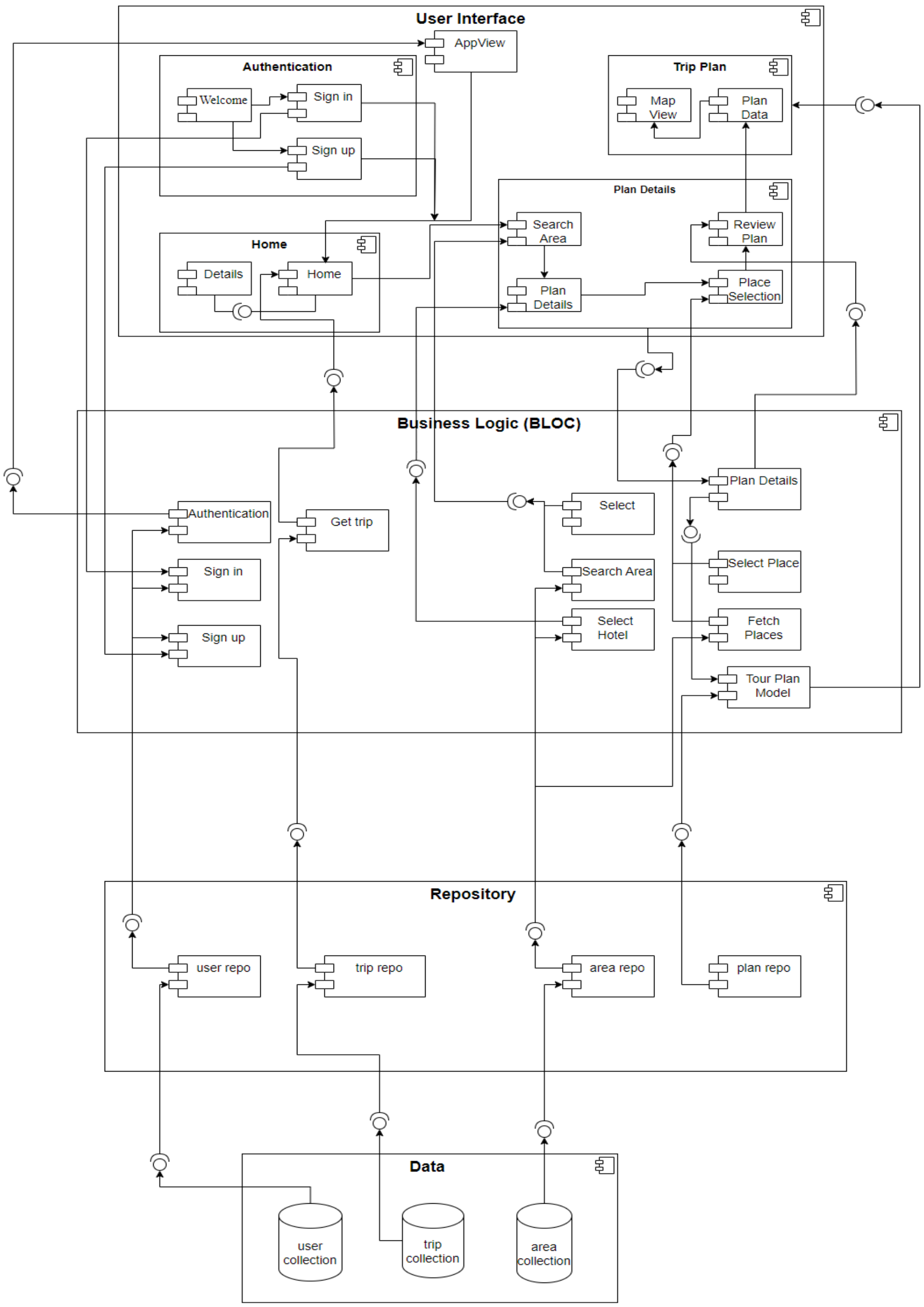


Fig 4.4.6 Component Diagram for Tour planning application

The component diagram shows the flow of activities between each component in the application. the component diagram offers a holistic perspective on the architecture of our application, elucidating the relationships and dependencies between different components and highlighting how they collaborate to deliver a seamless and comprehensive tour planning experience for users.

4.4.7 Deployment Diagram

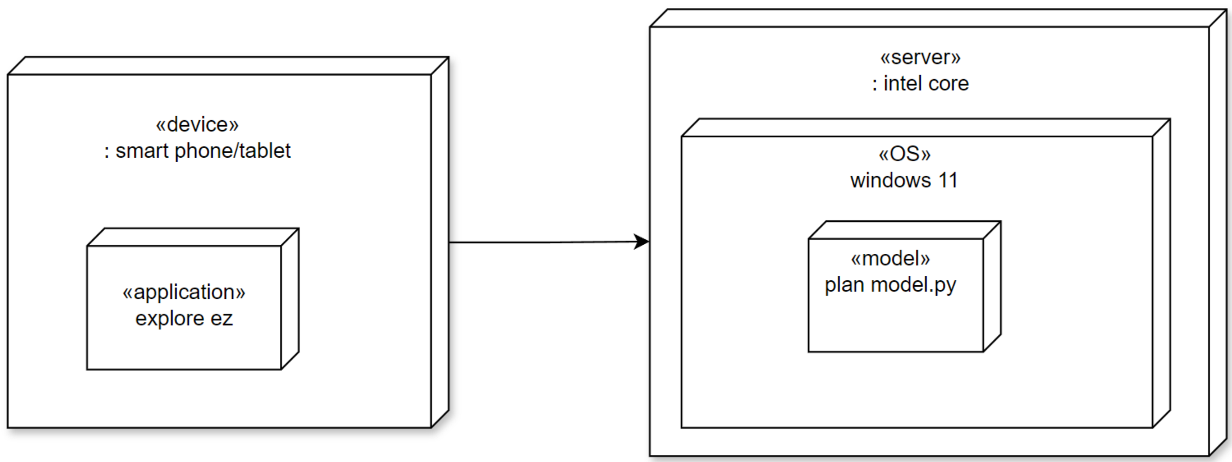


Fig 4.4.7 Deployment Diagram for Tour planning application

The deployment diagram shows the environment of the tour application is being deployed. The deployment diagram provides a visual representation of the environment in which our tour application is deployed, showcasing the distribution of components across various nodes and hardware devices. At its core, the diagram illustrates the physical infrastructure supporting our application, including servers, databases, and other resources.

4.5 DATA DICTIONARY

This is normally represented as the data about data. It is also termed as metadata some times which gives the data about the data stored in the database. It defines each data term encountered during the analysis and design of a new system. Data elements can describe files or the processes. Following are some rules, which defines the construction of data dictionary entries:

- Words should be defined to understand what they need and not the variable need by which they may be described in the program.
- Each word must be unique. We cannot have two definitions of the same client.
- Aliases or synonyms are allowed when two or more entries show the same meaning. For example, a vendor number may also be called a customer number. 10
- A self-defining word should not be decomposed. It means that the reduction of any information into subparts should be done only if it is really required, that is it is not easy to understand directly.
- Data dictionary includes information such as the number of records in a file, the frequency a process will run, security factors like pass word which the user must enter to get excess to the information

4.5.1 SIGN IN TABLE

Column name	Data type	Description	Constraint
Email	varchar	Username of the user	Not null
Password	varchar	Password of the user	Not null

4.5.1 Sign in table for Tour planning system

4.5.2 SIGN UP TABLE

Column name	Data type	Description	Constraint
Username	varchar	Full name of the user	Not null
Email	varchar	Username name of the user	Not null
Password	varchar	Password of the user	Not null

4.5.2 Sign up table for Tour planning system

4.5.3 PLACE TABLE

Column name	Data type	Description	Constraint
Place name	varchar	Name of place	Not null, unique primary key
Latitude	Double	Latitude coordinate of the place	Not null
Longitude	Double	Longitude coordinate of the place	Not null
Place image	String	URL of the image	Default

4.5.3 Place table for Tour planning system

4.5.4 TRIP TABLE

Column name	Data type	Description	Constraint
Trip	String	Name of the trip	Not null
Budget	String	Amount required	Not null
Days	Integer	Total required	Not null
Details	String	Trip description	Default

4.5.4 Trip table for Tour planning system

4.6 DATA FLOW DIAGRAM

4.6.1 LEVEL 0

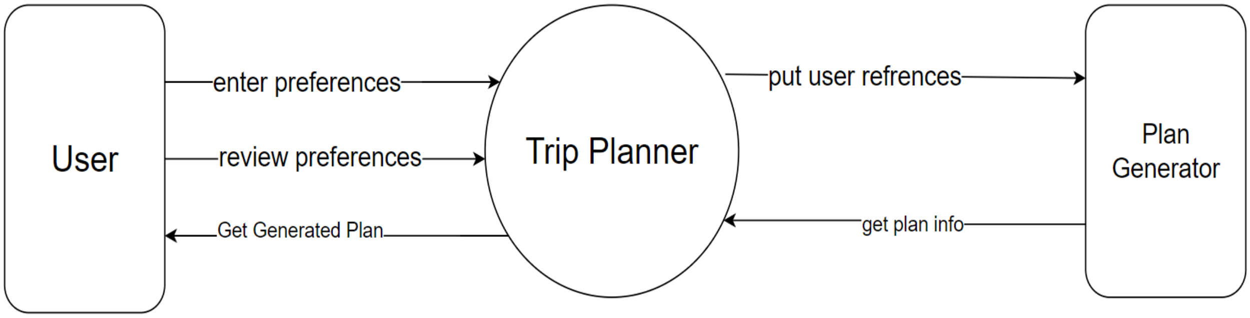


Fig 4.6.1 Dataflow diagram level 0

The zero level of data flow diagram of the Tour planning application system shows the various user programs. The zero level of the data flow diagram for the Tour planning application system encompasses a range of user programs aimed at providing a seamless experience for travelers. These programs include registration/login systems for account management, tour search and planning functionalities, user preferences, reviews etc.

4.6.2 FIRST LEVEL

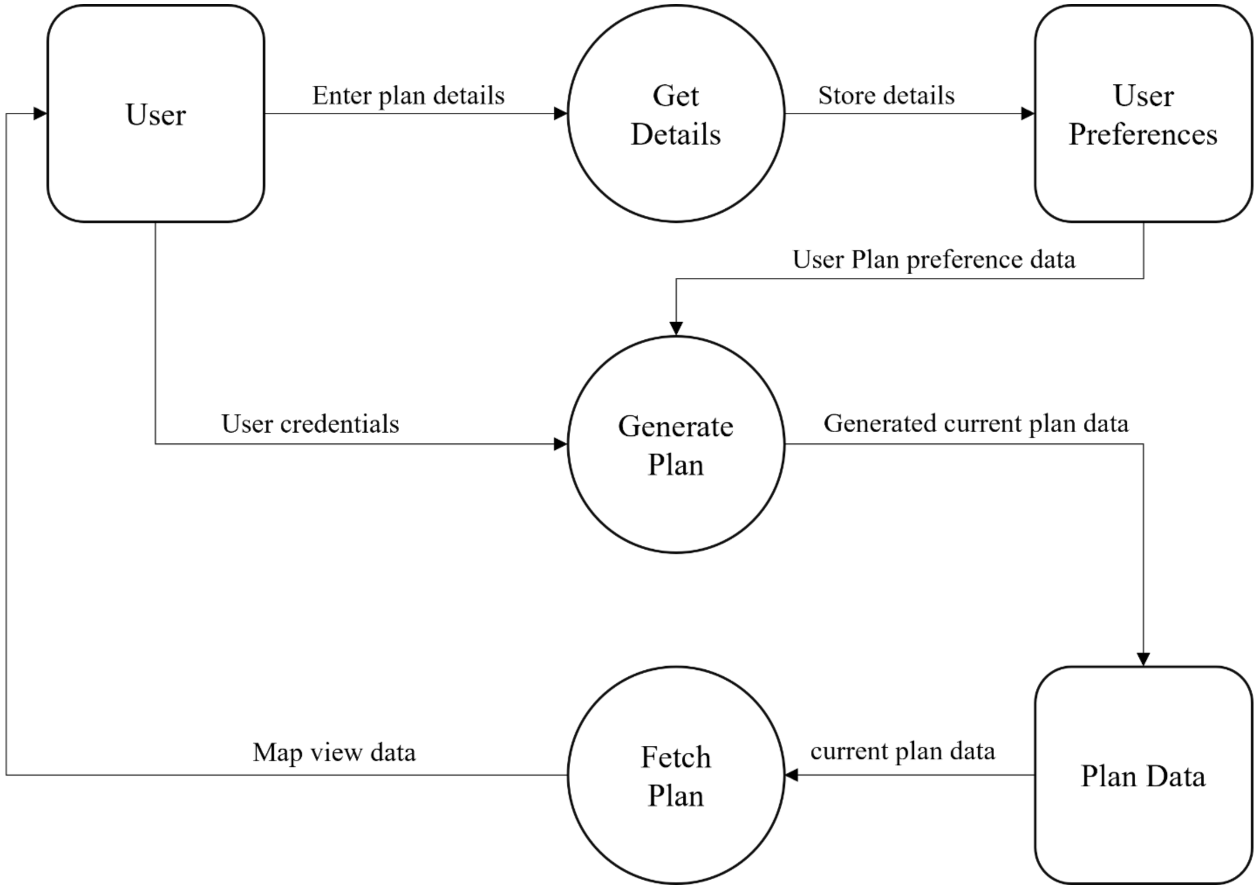


Fig 4.6.2 Dataflow diagram level 1

The first level of data flow diagram of Tour planning application system shows the various client program levels and their corresponding report. At the first level of the data flow diagram for the Tour planning application system, client program levels are detailed along with their corresponding reports, enhancing the understanding of system functionalities.

4.6.3 SECOND LEVEL

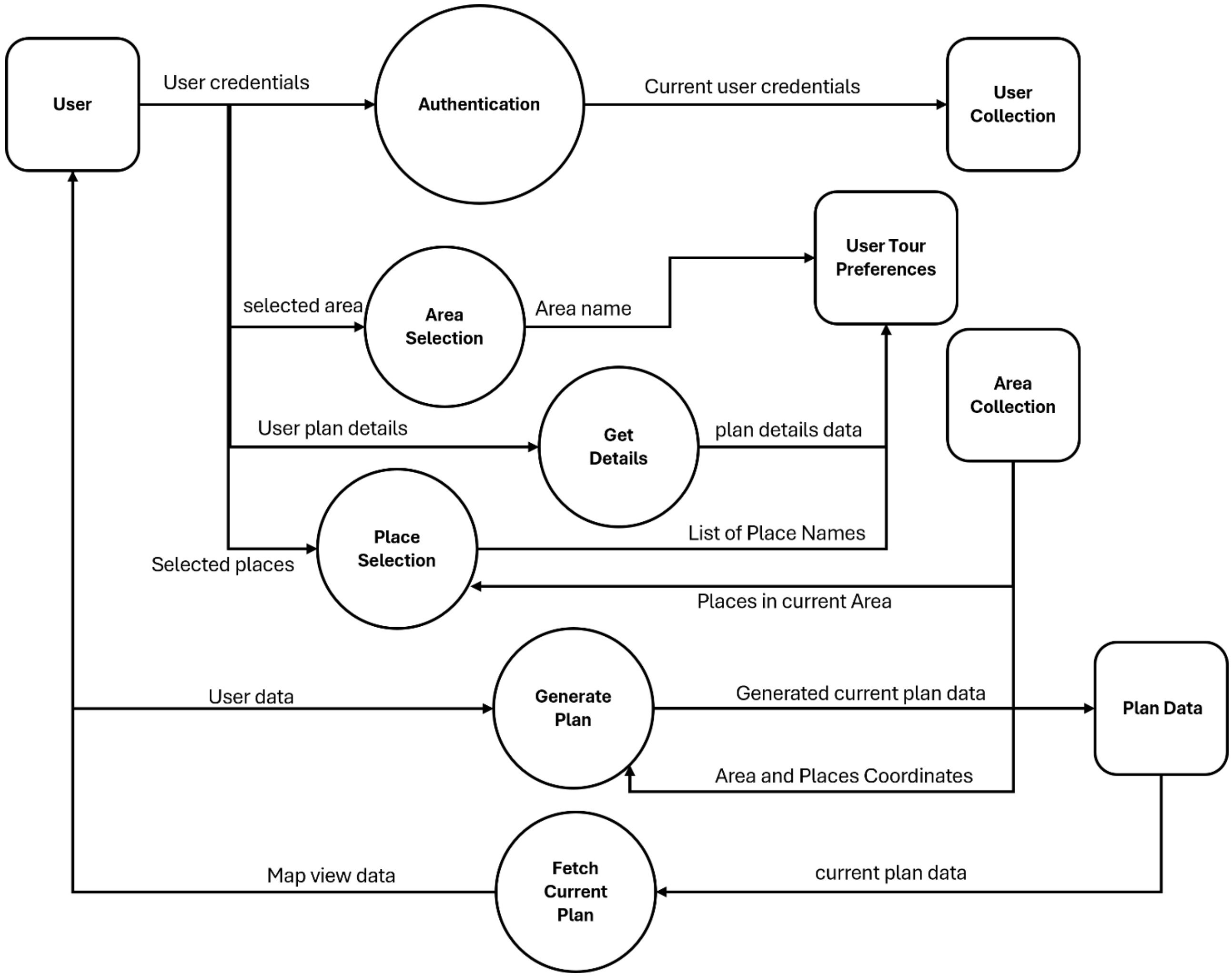


Fig 4.6.3 Dataflow diagram level 2

The second level of data flow Tour planning application system shows the various details of actions. At the second level of the data flow diagram for the Tour planning application system, intricate details of actions within the system are delineated to provide a comprehensive view of its operational processes. By delineating detailed actions at this level, the data flow diagram facilitates a granular understanding of system operations, enabling effective development, management, and optimization of the Tour planning application system.

CHAPTER 5

SYSTEM IMPLEMENTATION

5.1 PRESENTATION LAYER

- Authentication screens: Sign Up Screen and Sign in Screen.
- Home Screen: Displays recommend trips and its details.
- Trip Planning Screens: The Trip Plan Screen provides users with a comprehensive interface to organize and manage their travel itineraries efficiently.
- Trip Data Screen: The Trip Data feature offers insights into the generated day-wise trip plan and provides a comprehensive map view for efficient navigation and visualization of the itinerary.

The application features Sign Up Screen and Sign in Screen for authentication, a Home Screen with search for browsing places, and Trip Planning Screens (Plan Details Screen, Place Selection Screen, Area Selection Screen, and Plan Review Screen) for organizing trips. Widgets handle business logic and data retrieval directly, ensuring smooth operation and user interaction. This architecture emphasizes maintainability, scalability, and user satisfaction through clear separation of concerns and efficient communication between components. To ensure seamless functionality and data retrieval, widgets within the application communicate with the domain layer. This layer contains the business logic and handles interactions with the data storage, ensuring that the application operates smoothly and delivers relevant information to the users as needed. By separating concerns and implementing clear communication channels between the user interface and the domain layer, the application architecture promotes maintainability, scalability, and overall user satisfaction. In addition to the core functionalities described, the application architecture prioritizes extensibility and flexibility. It allows for easy integration of additional features and modules, facilitating future enhancements and updates to meet evolving user needs. The separation of concerns between the user interface and the domain layer enables efficient testing and debugging processes. Developers can isolate and test individual components, ensuring robustness and reliability across the application.

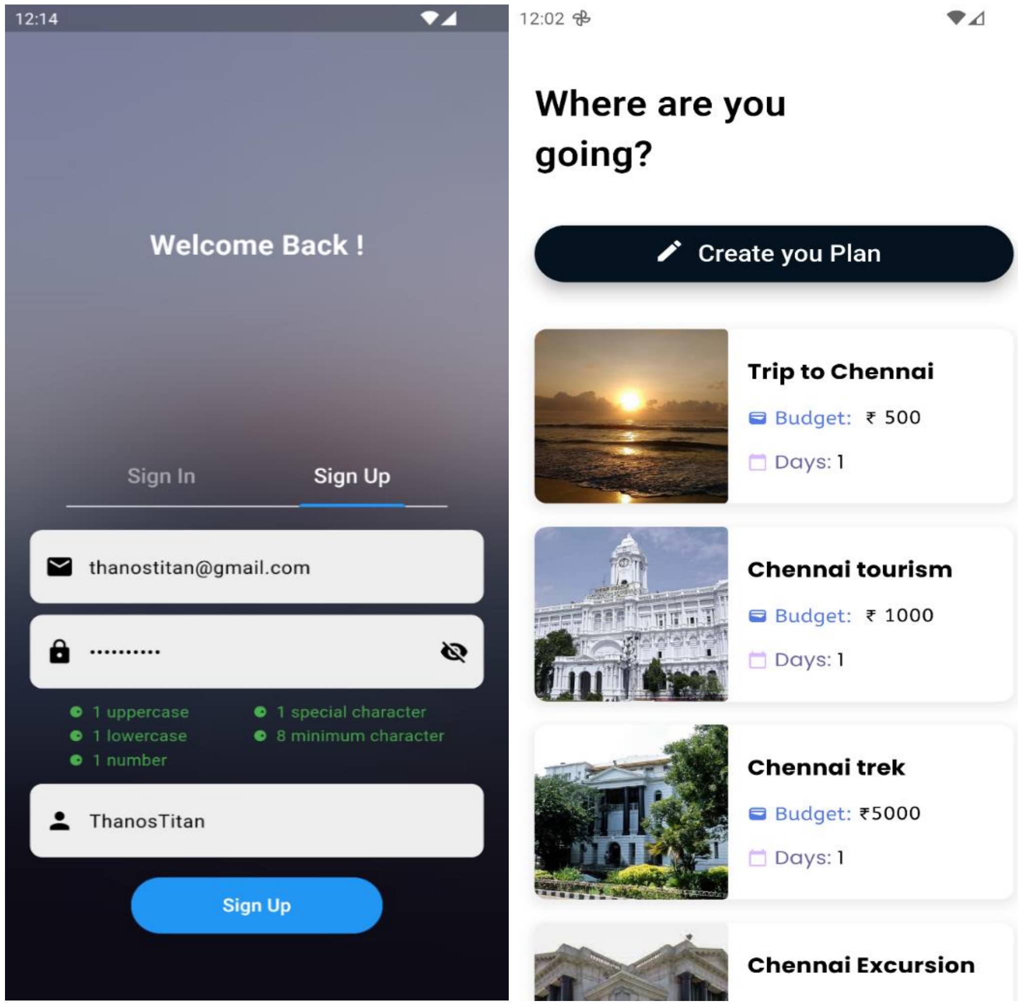


Fig 5.1.1 Sign up and home view

The Sign-Up Screen is a user-friendly interface for new users to register and create accounts easily. It collects essential information like username, email, and password securely, streamlining the onboarding process and expanding the user base effectively. The Home Screen serves as the central hub of the application, offering users a gateway to explore various destinations and plan their trips effectively. It provides a visually appealing interface where users can discover places of interest and access search functionality to find specific destinations or activities. The screen aims to engage users by presenting relevant and enticing information, such as popular tourist spots, nearby attractions, and personalized recommendations. Through intuitive design and seamless navigation, the Home Screen enhances the user experience, encouraging exploration and facilitating trip planning with ease and efficiency.

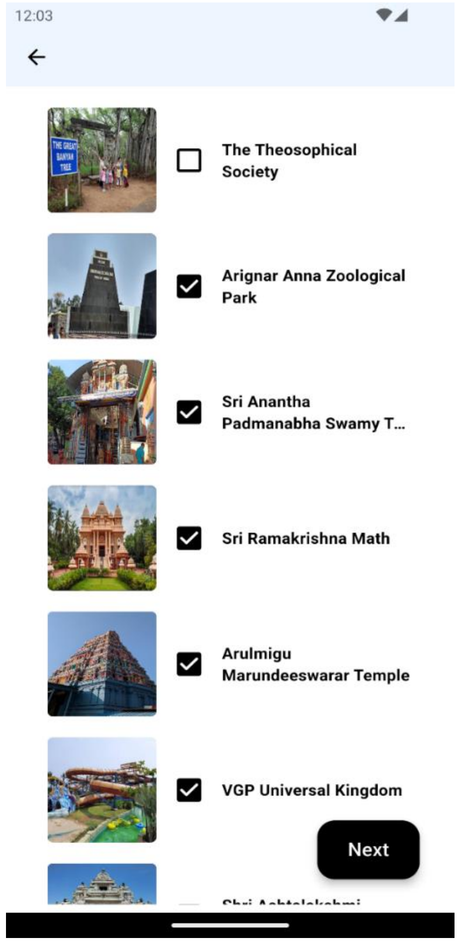


Fig 5.1.2 Place selection

The Destination Selector is a crucial feature within the application, designed to assist users in discovering and selecting their travel destinations. By leveraging various criteria such as user preferences, interests, budget, and time constraints, the Destination Selector provides personalized recommendations tailored to each user's unique needs. This functionality enhances the user experience by streamlining the destination selection process and ensuring that users find places that align with their interests and preferences. Overall, the Destination Selector serves as a valuable tool for users to explore and choose their ideal travel destinations, facilitating a more enjoyable and personalized travel planning experience.

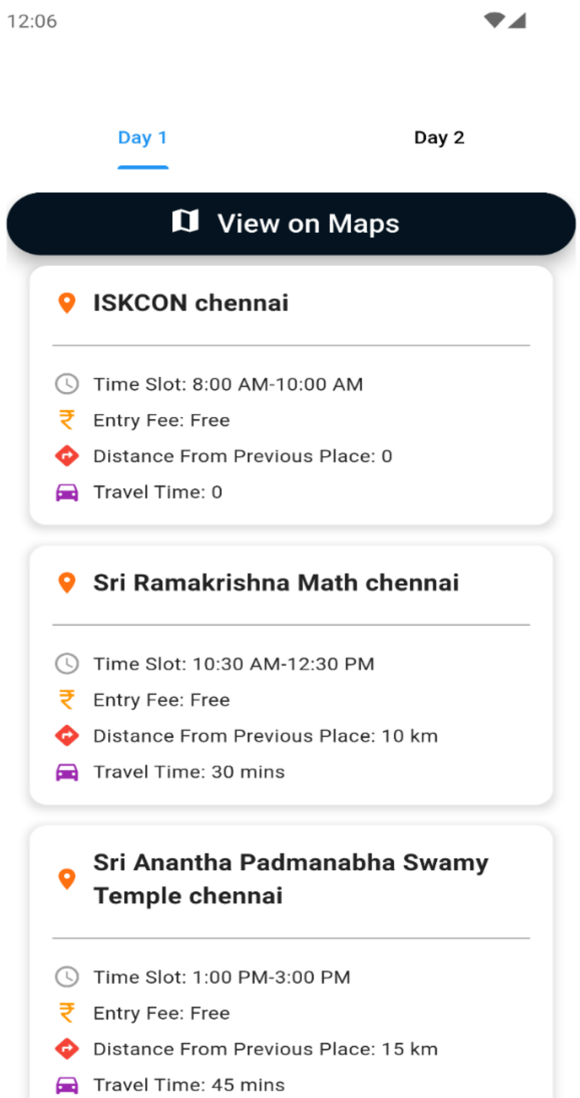


Fig 5.1.3 Plan Data

The Tour Plan Page is a central component of the application, providing users with a comprehensive overview of their planned trips. It typically displays detailed itineraries including selected destinations, activities, and timelines, allowing users to visualize their travel plans effectively. Additionally, the Tour Plan Page may offer feature such as map integration. Users can customize their tour plans, add or remove destinations, and adjust schedules as needed, ensuring flexibility and adaptability throughout the planning process. Overall, the Tour Plan Page serves as a one-stop destination for users to organize, manage, and optimize their travel itineraries, facilitating seamless and enjoyable travel experiences.

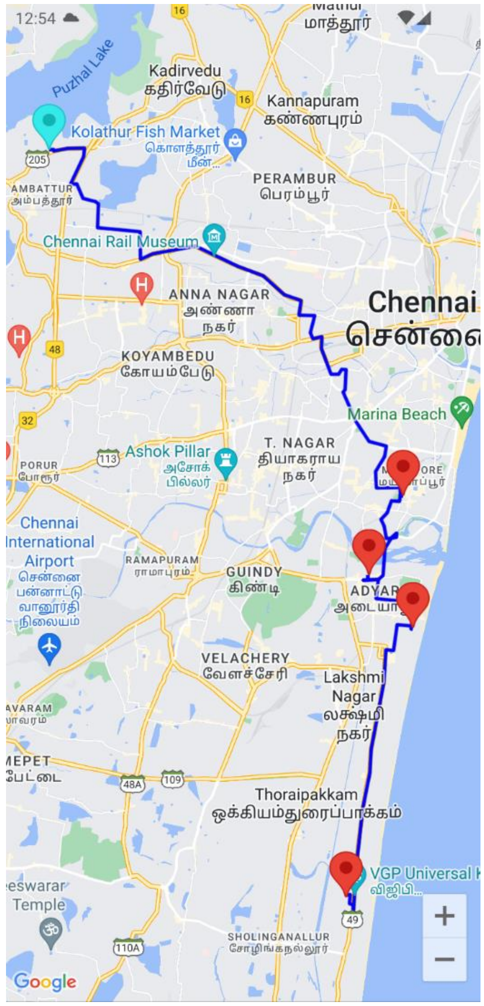


Fig 5.1.4 Map View

The Map View Page is a key feature within the application, offering users a visual representation of their planned trips. It typically integrates interactive maps, allowing users to explore destinations, routes, and points of interest efficiently. Users can view their itinerary overlaid on the map, providing a clear understanding of the geographical layout of their trip. Additionally, the Map View Page may offer functionalities such as real-time navigation, location-based recommendations, and the ability to search for nearby attractions or amenities. This feature enhances the user experience by providing a dynamic and intuitive way to interact with their travel plans, facilitating better decision-making and exploration. Overall, the Map View Page serves as a valuable tool for users to visualize, navigate, and optimize their travel experiences effectively.

5.2 DOMAIN AND REPO LAYER

- Manages user authentication with authentication bloc.
- Previous trip data states handled by get trips bloc.
- User trip plan preferences handled by various plan details bloc, area selection and search, fetch places and selection.
- Tour plan generation handled by tour plan model bloc.

Authentication: Sign in, sign up blocs are distinct sections of a user interface that facilitate the process of accessing a system or application securely. The sign in bloc allows existing users to enter their credentials, typically a username and password, to gain access to their account. On the other hand, the sign-up bloc provides new users with the means to create a new account by supplying necessary information such as username, email, and password, ensuring they can engage with the platform or application effectively. These blocs are fundamental components of user authentication systems, ensuring controlled access and personalized experiences for users. "Get Trips" is a feature or section within a platform or application designed to provide users with access to a curated selection of travel itineraries. This functionality allows users to browse through various travel destinations, packages, or experiences available through the platform. In the "Tour Details" section of our travel application, users embark on a journey of customization and planning to tailor their ideal travel experiences. Initially, they select a desired area or region they wish to explore, setting the stage for their adventure. From there, users can refine their choices by searching for specific locations or attractions within the chosen area, ensuring they narrow down options to suit their preferences. In the "Plan details" stage, users delve into the finer aspects of their trip, specifying details such as accommodation preferences, transportation modes, preferred activities, and budget constraints. In our tour application, the "Plan Details: Accommodation" feature serves as a pivotal component in crafting personalized travel experiences for users. Here, travelers can meticulously outline their lodging preferences to ensure comfort and satisfaction throughout their journey.

CHAPTER 6
SYSTEM TESTING

6.1 TESTING AND PERFORMANCE ANALYSIS

6.1.1 Test Case and Report:

TEST CASE ID	TESTCASE/ACTION TO BE PERFORMED	EXPECTED RESULT	ACTUAL RESULT	PASS/FAIL
1.	Opening tour application	Login page appears	Login page appears	Pass
2.	App version check and update	Update successful	Update successful	Pass
3.	Enter required sign up details	Registered successfully	Registered successfully	Pass
4.	Enter login details	Home page appears	Home page appears	Pass
5.	Select search bar	Destination selection page displayed	Destination selection page displayed	Pass
6.	Enter Destinations	Display the places	Display the places	Pass
7.	Enter the budget for the touring purpose	Budget Tab appears	Budget appears	Pass
8.	Enter time, date required	time, date, and accommodation tab appears	time, date, and accommodation tab appears	Pass
9.	Select accommodation	Accommodation Dialogue	Accommodation Dialogue	Pass
10.	Get user location	User current location will be displayed	User current location will be displayed	Pass

11.	Now select all the areas needed to be visited	Area selection page appears	Area selection page appears	Pass
12.	Plan review containing all the details	Review page displays	Review page displays	Pass
13.	Check the generated tour plan	Tour plan will be displayed	Tour plan will be displayed	Pass
14.	Check the timing and places in the generated tour plan	Areas will be displayed along with best timing	Areas will be displayed along with best timing	Pass
15.	Confirm the generated plan and route in case of no issues	Plan will be finalized and used for navigation	Plan will be finalized and used for navigation	Pass

6.1.1 Test Case and Report table for Tour Planning System

6.1.2 Performance analysis:

Fig 6.1.2 shows the performance analysis of the tour planning application reveals a direct relationship between the size of the plan and the time taken for processing, indicating scalability and efficient resource management. Despite longer processing times for larger plans, the application maintains satisfactory performance, prioritizing accuracy and user experience. This feedback loop allows for continual optimization, ensuring the application remains efficient and capable of handling diverse user requirements.

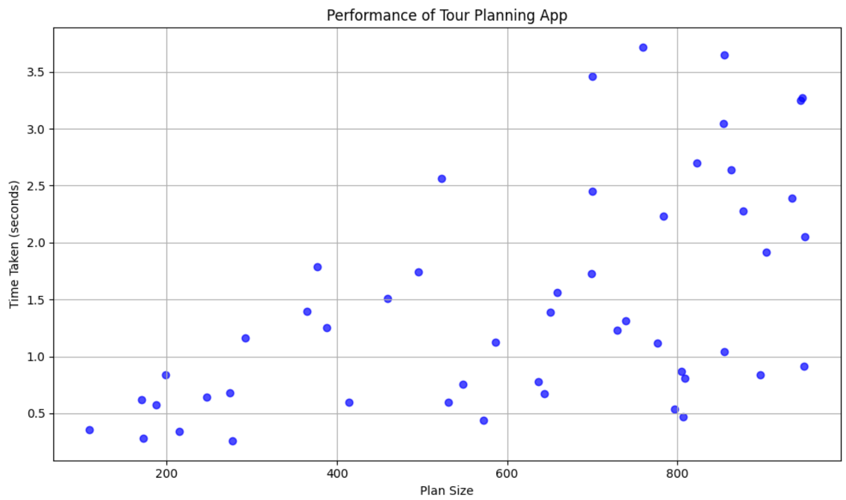


Fig 6.1.2 Performance Analysis Chart

6.2 RESULTS

Assessing the results of the tour-based application which employs distance matrix and TSP with Gemini Model to craft personalized tour plans based on user constraints like budget and time. Through iterative learning, it refines recommendations for destinations, activities, and transportation, enhancing efficiency and user satisfaction.

Tour Generation: The application utilizes a distance matrix and TSP with Gemini Model to create personalized tour plans based on user constraints like budget, time, and transportation preferences.

Unique Experience: The personalized approach distinguishes the application, offering users a unique and memorable travel experience tailored to their preferences.

Efficiency: The application optimizes resource utilization by suggesting cost-effective transportation options and tourist attractions, enhancing efficiency in travel planning.

Data Privacy: Stringent data privacy measures ensure user confidentiality, with personal information securely stored and used solely to enhance the travel experience, fostering trust and confidence among users.

Dynamic Adaptability: Real-time updates and feedback mechanisms enable tour plans to remain dynamic and responsive to changing circumstances, maintaining the quality of recommendations and user satisfaction.

6.3 DISCUSSIONS

Discussing the tour application presents a transformative approach to travel planning, leveraging advanced algorithms to provide personalized itineraries tailored to individual preferences and constraints.

Objective: The objective of implementing the tour application is to create a user-friendly platform that generates personalized travel itineraries based on user-provided constraints such as budget, time availability, and transportation preferences.

Algorithmic Efficiency: Discuss how algorithms such as TSP with Gemini Model enhance efficiency in generating personalized tour plans and optimizing user experiences.

Privacy Measures: Investigate the importance of stringent data privacy measures in maintaining user trust and confidence and explore how transparency in data usage fosters trust among users.

Effectiveness: The effectiveness of the tour application lies in its ability to efficiently generate personalized travel itineraries tailored to user preferences, optimizing resource utilization and enhancing user satisfaction.

Coverage and reach: The tour application aims to achieve wide coverage and reach by providing personalized travel itineraries to users across diverse geographical regions and travel preferences. Through its online platform, accessible via web browsers and mobile devices, it seeks to cater to a global audience

Financial inclusion: It refers to the accessibility and availability of financial services to individuals and businesses, particularly those who are underserved or excluded from traditional banking systems.

CHAPTER 7

7.1 CONCLUSION

In conclusion, the utilization of the distance matrix and advanced TSP with Gemini Model within our tour-based application represents a significant advancement in personalized tour planning. By leveraging TSP, we have crafted a system capable of generating the optimal tour plan for users, considering their specified budget constraints, time limitations, and transportation preferences. Through the iterative learning process of Gemini model, our application can effectively analyze vast amounts of data to recommend the most suitable destinations, activities, and transportation options tailored to each user's unique requirements. This not only enhances user satisfaction but also optimizes resource utilization and promotes sustainable travel practices. Furthermore, the integration of Google Maps into our application enables continuous improvement and adaptation based on user feedback and evolving trends. By harnessing the power of machine learning, we can ensure that our tour plans remain dynamic, relevant, and tailored to meet the changing needs and preferences of our diverse user base. In essence, our tour-based application powered by Gemini Model represents a paradigm shift in the way individuals plan and experience travel.

7.2 FUTURE ENHANCEMENTS

- In future iterations, the tour-based application has the potential to expand globally, offering comprehensive tour options worldwide through partnerships with local tourism agencies and the integration of real-time traffic data for optimized transportation recommendations.
- Strategic alliances with travel entities could unlock promotional offers while diversifying transportation modes and implementing language localization features would enhance accessibility and convenience for users globally.
- Further advancements in personalized recommendations, community engagement, and augmented reality integration promise to elevate the user experience, making the application a comprehensive and indispensable travel companion.

CHAPTER 8
REFERENCES

[1] Fan-Hsun Tseng (2020). "Optimal Route Planning System for Logistics Vehicles Based on Artificial Intelligence."

[2] Xiao Zhou, Mingzhan Su, Zhong Liu, De Zhang (2019). "Smart Tour Route Planning Algorithm based on Clustering Center Motive Iteration Search."

[3] H.H. Owaied, H.A Farhan, N. AI- Hawamdeh, N. AI-Okialy (2011). "A Model of Intelligent Tourism Guide System."

[4] Mobin Ding, Volume2021, Article ID 2227798, 7 pages https://doi.org/10.1155/2021/2227798 "Research on Tourism Route Planning Based on Artificial Intelligence Technology."

[5] Takashi Hasuike, Hideki Katagiri, Hiroe Tsubaki, Hiroshi Tsuda (2013). "Personal Tour Planning Incorporating Standard Tour Routes and Tourist Satisfaction."

[6] Feiran Huang, Jie Xu, Jian Weng (2020). "Multi-Task Travel Route Planning with a Flexible Deep Learning Framework."

[7] Shou-Chih Lo, Fu-Chiang Huang, Chien-Chung Wang (2014). "Transport Route Planning for Mobile Tour Applications."

[8] Yuan Yao, Zhe Peng, Bin Xiao (2018). "Parallel Hyper-Heuristic Algorithm for Multi-Objective Route Planning in a Smart City."

[9] Kewad Rathod, Vinay Govekar, Mayuri Gaikwad, Ichhanshu Jaiswal, JETIR May 2021, Volume 8, Issue 5, “AI – Based Tourist Guide.”

APPENDICES

A.1 SDG GOALS

- SDG 8: Decent Work and Economic Growth: By promoting tourism and facilitating bookings for accommodations, your application contributes to economic growth by supporting businesses in the hospitality sector, creating employment opportunities, and stimulating local economies in tourist destinations.

- SDG 9: Industry, Innovation, and Infrastructure: Your application leverages advanced technologies and robust data integration to provide a seamless platform for travel planning, accommodation booking, and destination exploration, thereby contributing to the development of innovative infrastructure within the tourism industry.

- SDG 11: Sustainable Cities and Communities: By providing transportation recommendations and promoting responsible tourism practices, your application encourages sustainable mobility and helps alleviate the environmental impact of tourism on urban areas, contributing to the development of sustainable cities and communities.

A.2 SOURCE CODE

Main.dart

```
import 'package:flutter/material.dart';
import 'package:firebase_core/firebase_core.dart';
import 'package:flutter/services.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import 'package:user_repository/user_repository.dart';
 import 'firebase_options.dart';
import 'app.dart';
import 'simple_bloc_observer.dart';

void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp(
    options: DefaultFirebaseOptions.currentPlatform,
  );
  Bloc.observer = SimpleBlocObserver();
  SystemChrome.setPreferredOrientations([DeviceOrientation.portraitUp]);
  runApp(MainApp(FirebaseUserRepo()));
}
```

App.dart

```
class MainApp extends StatelessWidget {
  final UserRepository userRepository;
  const MainApp(this.userRepository, {super.key});
  @override
  Widget build(BuildContext context) {
    return MultiRepositoryProvider(providers: [
      RepositoryProvider<AuthenticationBloc>(
          create: (_) => AuthenticationBloc(myUserRepository: userRepository)),
      RepositoryProvider<GetTripBloc>(
```

```
        create: (_) => GetTripBloc(FirebaseTripRepo())),
      RepositoryProvider<SearchAreaBloc>(
        create: (_) => SearchAreaBloc(FirebaseAreaRepo())),
      RepositoryProvider<PlanDetailsBloc>(create: (_) => PlanDetailsBloc()),
      RepositoryProvider<FetchPlacesBloc>(
        create: (_) => FetchPlacesBloc(FirebaseAreaRepo())),
      RepositoryProvider(create: (_) => SelectAreaBloc()),
      RepositoryProvider(create: (_) => SelectHotelBloc(FirebaseAreaRepo())),
      RepositoryProvider(create: (_) => SelectPlaceBloc()),
      RepositoryProvider(create: (_) => TourPlanModelBloc(ModelPlanRepo())),
    ], child: const MyAppView());
  }
}
```

Distance_matrix.py

```
import googlemaps
import numpy as np
from python_tsp.heuristics import solve_tsp_simulated_annealing
class TSPModel:
  def __init__(self,places):
 gmaps_client=googlemaps.Client(key="AIzaSyAgLFO4XZ0S8ywCdBsOdNh1tDtv3dlPMOE")
    distance_dict={}
    for place1 in places:
      for place2 in places:
        direction_result=gmaps_client.directions(place1,place2,mode="driving")
        distance_dict[place1,place2]=direction_result[0]['legs'][0]['distance']['value']
    distance=[]
    for place in places:
      distance_1d=[]
      for place2 in places:
        distance_1d.append(distance_dict[place,place2]/1000)
      distance.append(distance_1d)
```

```
    distance_matrix = np.array(distance)

    permutation,distance = solve_tsp_simulated_annealing(distance_matrix)
    route = ""
    for i in range(1,len(permutation)):
      route+=places[permutation[i]]
      route+=","
    self.places=route
```

Backend model

```
from flask import Flask,send_from_directory,request
import google.generativeai as genai
from distance import TSPModel
app = Flask(__name__)
@app.route("/")
def start():
    d={}
    inputstr=str(request.args['query'])
    lst=inputstr.split(',')
    a=lst[0]
    b=lst[1]
    c=lst[2]
    del lst[0]
    del lst[0]
    del lst[0]
    current =lst[0],lst[1]
    del lst[0]
    del lst[0]
    lst.insert(0, current)
    obj=TSPModel(lst)
    str1=a+","+b+","+c+","
```

```
    lst1=obj.places.split(',')
    for i in range(len(lst1)):
        str1+=lst1[i]
    def model(s):
        genai.configure(api_key="AIzaSyB1OICYjUzxVZIrkO7texsBGw-ZeK-4K_s")
        generation_config = {
        "temperature": 0,
        "top_p": 1,
        "top_k": 1,
        "max_output_tokens": 2048,
        }
        safety_settings = [
        {
            "category": "HARM_CATEGORY_HARASSMENT",
            "threshold": "BLOCK_MEDIUM_AND_ABOVE"
        },
        {
            "category": "HARM_CATEGORY_HATE_SPEECH",
            "threshold": "BLOCK_MEDIUM_AND_ABOVE"
        },
        {
            "category": "HARM_CATEGORY_SEXUALLY_EXPLICIT",
            "threshold": "BLOCK_MEDIUM_AND_ABOVE"
        },
        {
            "category": "HARM_CATEGORY_DANGEROUS_CONTENT",
            "threshold": "BLOCK_MEDIUM_AND_ABOVE"
        },
        ]
        model = genai.GenerativeModel(model_name="gemini-1.0-pro",
generation_config=generation_config, safety_settings=safety_settings)
```

```
        prompt_parts = [
        "\"Make a tour plan for these places in Chennai, the plan must  contains  place name, exact time slot, the entry fee, distance from previous loaction and travel time. Give results in minimum no days and  in python dict formate in (single line) eg start with {\"index\"={\"day\":trip_day,\"place_name\":\" \",\"time_slot\":\" \",\"entry_fee\":\" \",\"distance_from_previous_location\":\" \",\"travel_time\":\" \"},\"index\"=detials} .Maximum days="+s,
        ]
        response = model.generate_content(prompt_parts)
        d['output'] = response.text
    model(str1)
    return d
if __name__ == '__main__':
    app.run()
```

Function.dart

```
import 'dart:convert';
 import 'package:plan_repository/plan_repository.dart';
 import 'package:http/http.dart' as http;
 fetchdata(String url) async {
  http.Response response = await http.get(Uri.parse(url));
  return response.body;
}
 Future<String> getdata(String url) async {
  var data = await fetchdata(url);
  var decoded = jsonDecode(data);
  var output = decoded['output'];
  return output;
}int calculateDays(String startDate, String endDate) {
DateTime startDateTime = DateTime.parse(startDate);
DateTime endDateTime = DateTime.parse(endDate);
```

```
Duration difference = endDateTime.difference(startDateTime);
  return difference.inDays;
}
 List<DayPlan> getDayPlanData(String val) {
  Map<String, dynamic> jsonData = json.decode(val);
  List<DayPlan> dayPlanData = [];
  jsonData.forEach((key, value) {
    dayPlanData.add(DayPlan.fromEntity(DayPlanEntity.fromDocument(value)));
  });
  return dayPlanData;
}
 List<List<DayPlan>> getDayPlans(List<DayPlan> dayPlanData) {
  Map<String, List<DayPlan>> dayPlans = {};
 for (DayPlan dayPlan in dayPlanData) {
    final String day = dayPlan.day;
    if (dayPlans.containsKey(day)) {
      dayPlans[day]!.add(dayPlan);
    } else {
      dayPlans[day] = [dayPlan];
    }
  }
 return dayPlans.values.toList();
}
 List<String> planPlaces(List<DayPlan> plan, String areaName) {
  List<String> places = [];
 for (int i = 0; i < plan.length; i++) {
    String place = plan[i].placeName;
    List<String> str = place.split(" ");
 if (str[str.length - 1].toLowerCase() == areaName.toLowerCase()) {
      String correct = "";
      for (int j = 0; j < str.length - 2; j++) {
```

```
        correct += "${str[j]} ";
      }
      correct += str[str.length - 2];
      places.add(correct);
    } else {
      places.add(place);
    }
  }
  return places;
}
```

Plan_Repo.dart

```
class PlanRepo {
  String url = "";
  @override
  Future<List<List<DayPlan>>> getPlan(MyPlan plan) async {
    String value = "";
    int days = 1;
    try {
      days = calculateDays(plan.startDate, plan.endDate);
      value += "$days" ",";
      value += "${plan.startTime}-${plan.endTime}time fram per day,";
      value += "${plan.budget}total budget for trip,";
      value +=
          "${plan.accommodation.latitude!},${plan.accommodation.longitude!},";
      for (int i = 0; i < plan.places.length - 1; i++) {
        value += "${plan.places[i].placeName} chennai, ";
      }
      value += "${plan.places[plan.places.length - 1].placeName} chennai";
      url = 'http://10.0.2.2:5000/?query=' + value;
      final String ans = await getdata(url);
```

```
      List<DayPlan> dayPlanData = getDayPlanData(ans);
      log(dayPlanData.toString());
      return getDayPlans(dayPlanData);
    } catch (e) {
      log(e.toString());
      rethrow;
    }
  }
  @override
  List<MarkPoint> getPoints( List<DayPlan> dayPlan, List<Place> selectedPlaces, String
areaName) {
    List<MarkPoint> points = [];
    List<String> places = planPlaces(dayPlan, areaName);
    log(selectedPlaces.first.placeName);
    for (int i = 0; i < places.length; i++) {
      log(places[i]);
      Place? found =selectedPlaces.firstWhere((element) =>element.placeName.toLowerCase() ==
places[i].toLowerCase());
points.add(MarkPoint( name: places[i], coordinates: LatLng(double.parse(found.latitude ??
""),double.parse(found.longitude ?? ""))));
    }
    return points;
  }

  @override
  MarkPoint getCurrentPoint(Place place) {
    return MarkPoint(
        name: place.placeName,
        coordinates: LatLng(double.parse(place.latitude ?? ""),double.parse(place.longitude ??"")));
  }
}
```

A.3 SCREENSHOTS

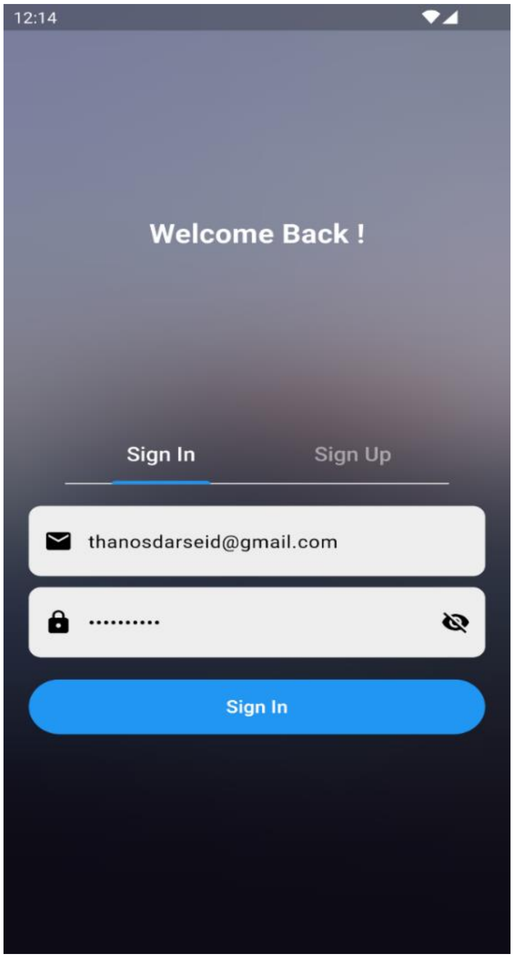


Fig A.3.1 Sign In Screen

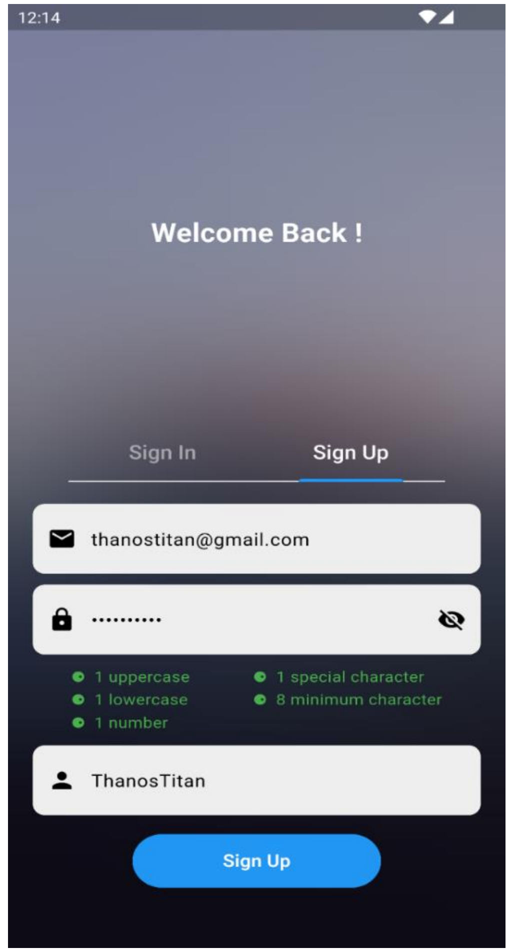
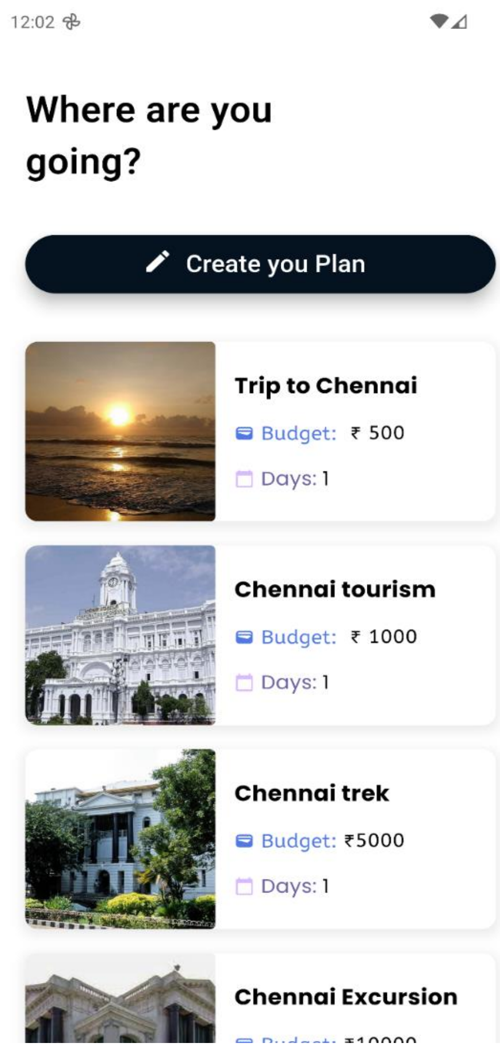


Fig A.3.2 Sign Up Screen

The signup screen collects essential user information like username and email to create accounts, while the sign-in screen allows returning users to access their accounts securely. Both screens prioritize user convenience and security, employing encryption protocols to safeguard data and intuitive design elements to enhance the user experience. Clear instructions and intuitive design elements enhance the user experience, fostering trust and engagement.



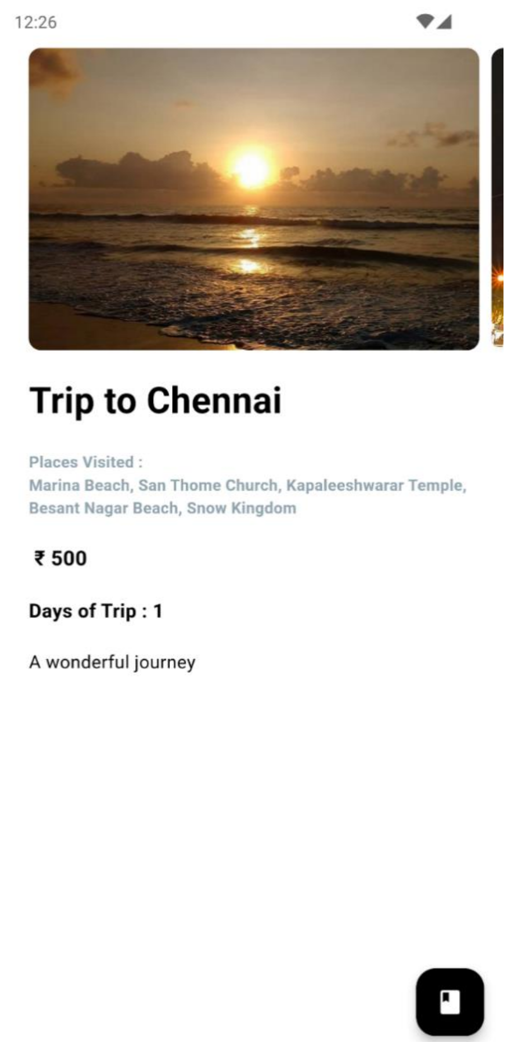


Fig A.3.3 Home Screen **Fig A.3.4 Details Screen**

Home screen: The Home screen serves as the central hub where users can access key features and navigate to various sections of the app. It typically presents users with a visually appealing interface that showcases popular or recommended tours, along with options for budget and number of days

Details screen: The application provides users with comprehensive information about a specific tour. It typically includes details such as duration, pricing, inclusions, and exclusions. Users can view photos or videos, read descriptions, and select required places for their tour.

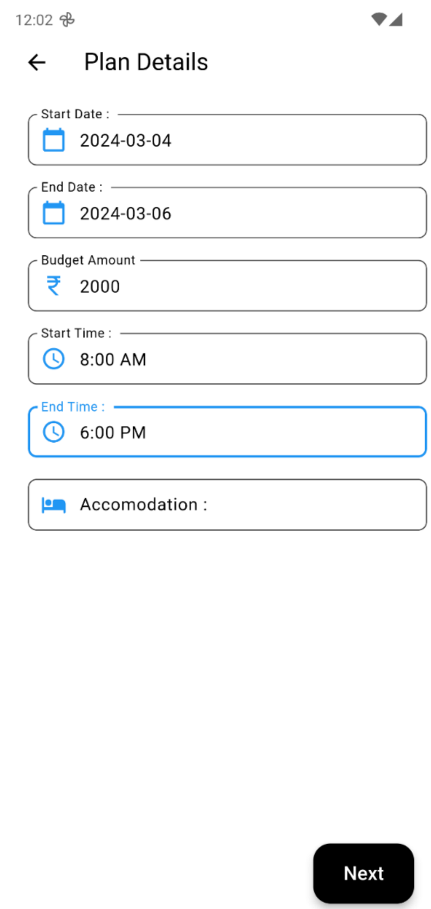
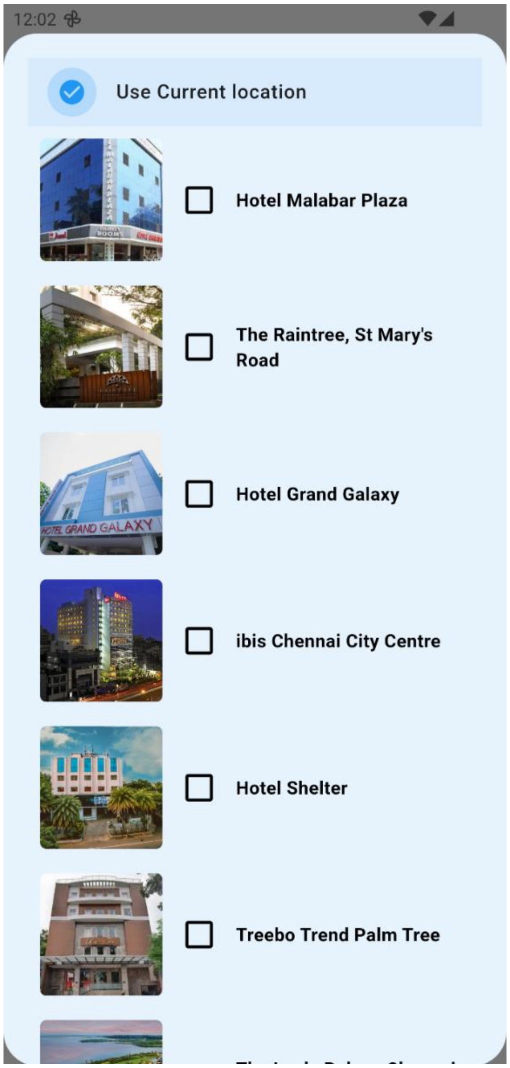


Fig A.3.5 Area Selection **Fig A.3.6 Plan Details Screen**

Area selection: The Area Selection feature present in the tour application allows users to choose their desired destination or region for exploring available tours.

Plan details screen: The Plan Details Screen offers pricing information, time options, and the ability to customize the tour package according to the user's preferences. Overall, the screen aims to provide users with all the necessary information to make an informed decision and plan their travel itinerary effectively, enhancing their overall experience with your application.



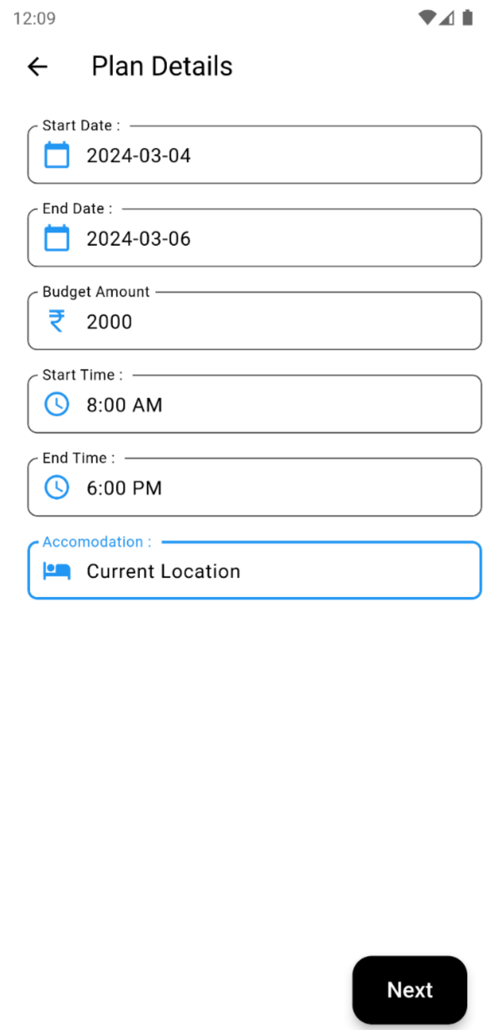
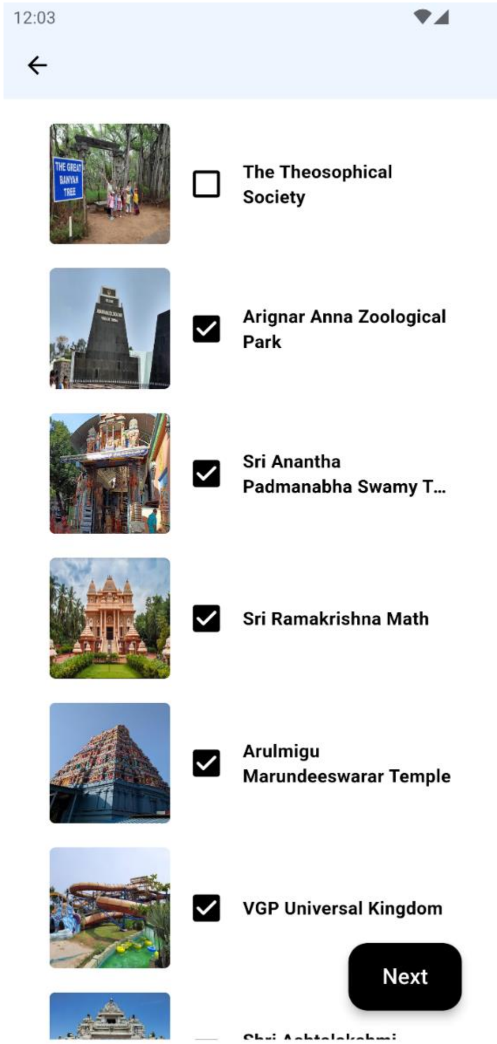


Fig A.3.7 Accommodation Selection Dialog **Fig A.3.8 Plan Details with Accommodation**

Accommodation Selection Dialog: The Accommodation Selection Dialog in the tour application offers users the convenience of choosing accommodation options with ease. Integrated with the current location feature, it provides users with nearby hotel recommendations based on their location, ensuring convenience and accessibility.

Plan details with accommodation: The Plan Details with Accommodation feature in the tour application offers users a comprehensive view of their travel itinerary along with date, time and budget customization accommodation options.



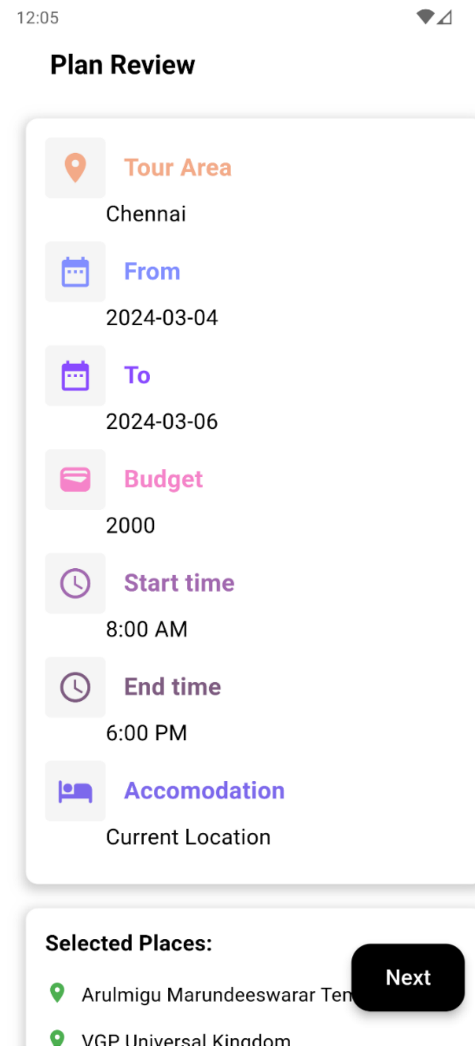
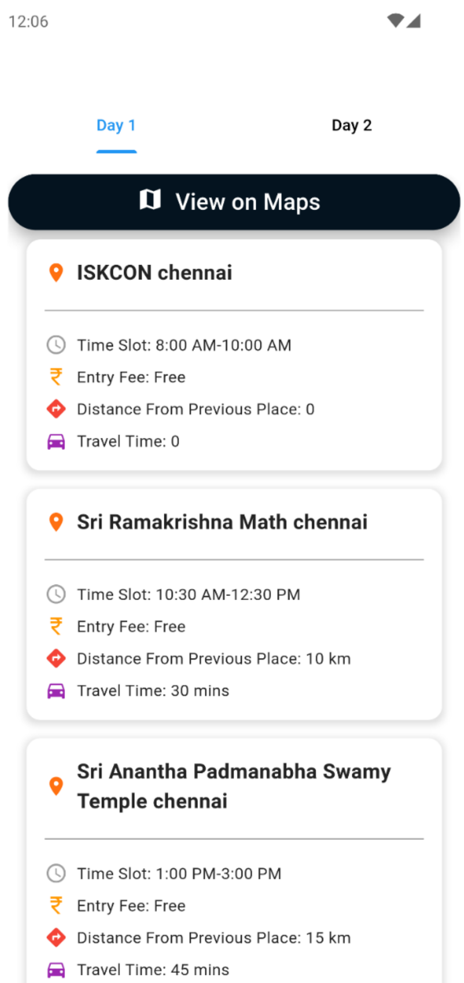


Fig A.3.9 Place Selection Page

Fig A.3.10 Plan Review Page

Place selection page: The Place Selection Page in the application allows users to explore and choose their desired destinations. It presents users with a user-friendly interface where they can search for specific places or browse through a curated list of popular destinations.

Plan review page: The application provides users to evaluate and finalize their travel plan before confirmation. It presents users with a summary of their selected tours, activities, accommodations, and any other relevant details. Users can review each component of their plan, including dates, times, and pricing.



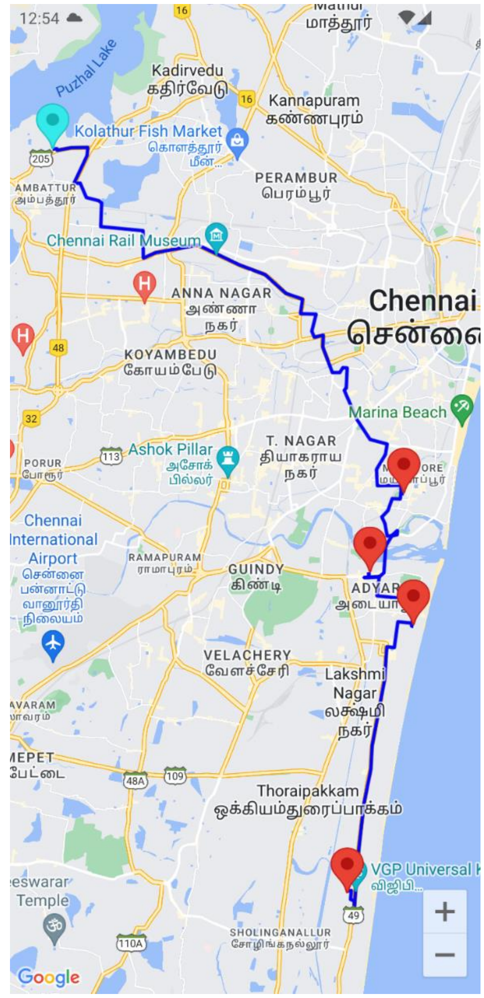


Fig A.3.11 Tour Plan Data Page

Fig A.3.12 Map View Screen

Tour plan data page: The page offers a clear and organized layout, presenting users with comprehensive insights into their upcoming travel arrangements. Users can review and edit their tour plans, add or remove activities, update booking details, and view important notifications related to their trips.

Map view screen: The Map View Screen in the application offers users a visual representation of tour destinations, attractions, and points of interest. Integrated with map technology, this screen provides an interactive and dynamic experience, allowing users to explore their travel plan.

PLAGIARISM REPORT

Transformative Tour Planning with Dynamic Route Optimization using Big Data

Dhayanand G.B,
Department of Computer Science and Engineering,
Panimalar Engineering College, India

Bangarru Srijesh M,
Department of Computer Science and Engineering,
Panimalar Engineering College, India

Barathan K,
Department of Computer Science and Engineering,
Panimalar Engineering College, India

***Abstract*—** Planning a tour before embarking on a journey is crucial for a more fulfilling experience. Tour planning is a challenging task and lots of people scrap their tour journey due to ineffective planning. Sometimes it may lead to delays in travel and unsafe journeys etc. Our research solves this problem by autonomous tour planning with multiple destinations. Most route planning algorithms work on constraints like route traffic, road conditions, construction works, and non-safety roads. With all these constraints we include a better visiting time which will perfect the route planning with efficient travel and better viewing experience. With our vast dataset at your fingertips, you have the power to prioritize the places that intrigue you the most. It is all about tailoring your journey to your desires. This system weaves together your preferences, budgets, precious time, your residence's proximity, and the ideal mode of transport. With all this our system effortlessly devises the finest possible plans. The heart of our model is developed with the RBM (Restricted Boltzmann Machine) algorithm, with this deep-learning recommendation algorithm optimal plans are formulated by considering a multitude of factors including such as the ideal time for visitation, minimizing travel distances, and ensuring budget-friendly options. Through meticulous analysis, experimentation, and subsequent results, the system crafts a meticulously organized tour itinerary, ensuring an efficient and impactful journey. Notably redundant, the system operates with remarkable performance, further enhancing its reliability and effectiveness.

Keyword: Tour Planning, Multiple destination, Ideal visiting time, User preferences, Tour budgets, Data analysis, Restricted Boltzmann Machine, Deep learning algorithm.

I. INTRODUCTION

The tourism industry stands as a thriving sector in the contemporary global market. The best way to escape the holiday boredom is to plan a perfect tour vacation. However, the complications involved in planning, such as budget considerations and time constraints, often derail travel expectations.
The lack of a well-structured plan further contributes to tour cancellations. Although the planning process demands a significant investment of time as one sits through numerous hotels, attractions, and restaurants. The exhaustive task involves scrutinizing ratings, reviews, and various attributes, all while trying to tailor the experience to fit within a specified budget.

Crafting a vacation plan that comprehensively considers individual travel preferences without scouring through countless websites seems nearly unattainable. Our project's focus is to manage the problematic planning area, allowing travellers to dedicate more time to enjoying their vacation.

By gathering details such as destination, travel dates, budget, hotel amenities, preferred attraction categories, and cooking preferences, we aim to generate a personalized travel plan. This comprehensive plan will guide users on where to stay, suggest activities throughout the day, and recommend dining options for each meal.

Using Data Science tools and techniques, our application aims to serve as a singular solution for travellers who need efficient vacation planning, along with budgetary control and best timing and spots, etc. It also incorporates budget-based tour planning which enables the user to visit places according to their financial plan and their income level. This application also offers users insights into the optimal timing and seasonality for visiting specific places based on their chosen destinations. By providing information on the best times to visit, users can enhance their overall travel experience.

Moreover, the application can perform the duty of a guide by informing the users with a well-organized list of hotels and restaurants that are positioned along their selected routes.

This feature serves the requirements of tourists by ensuring that they have convenient options for refreshment and rest throughout their journey. Furthermore, the application serves as a comprehensive travel companion by offering detailed transport-related information, including timings and vehicle types, for reaching their destinations, especially in cases where users don't have access to personal transportation. Whether it's public transport, ridesharing services, or other modes of transportation, the application seeks to present a thorough overview of available options by enhancing the overall accessibility and flexibility of travel plans.

With the integrations of budget considerations, optimal timing recommendations, transport information, and a well-planned selection of accommodations and dining, this application aims to perfect the travel planning experience for users. This project aims to be a go-to tool for travellers planning their vacation which can be done by leveraging Data Science tools and techniques, which will be explained in this report.

II. RELATED WORKS

Route planning is a testament to human intelligence. In the olden days, people would travel independently, finding their way to their destinations. This process has evolved from mental calculations to complex problems. Some of the classical route or path planning problems include the Travelling Salesman Problem (TSP) [1], the Vehicle Routing Problem (VRP) [1], the Chinese Postman Problem [1], the Rural Postman Problem [1], Dijkstra's algorithm [1], and the Bellman-Ford algorithm [1]. In the TSP, a person visits 'n' cities exactly once and returns to the source city at the lowest travel cost. It is an NP-hard problem. Solutions can be found using dynamic programming or branch-and-bound algorithms. For large-scale problems, genetic algorithms can be applied. There is also an Euclidean TSP, which focuses on measuring path cost based on Euclidean distance. It is also NP-hard. The VRP determines the optimal vehicle route for transportation, considering factors like minimum cost, distance, and travel time. It is an NP-complete problem. The Chinese Postman Problem focuses on the edges with the least cost to create the shortest route. This problem is adapted from the Rural Postman Problem, which considers a subset of edges. The Rural Postman Problem is an NP-complete problem. Dijkstra's algorithm is used for finding the shortest paths between nodes in a weighted graph. It is applicable when the edge weights are non-negative. The Bellman-Ford algorithm solves the same problem but allows for negative edge weights. Compared to Dijkstra's algorithm, the A* algorithm solves the single-source shortest path problem with a smaller search space. The more preferred classical algorithm is the

Floyd-Warshall algorithm [1], which solves the all-pairs shortest paths problem and can be applied in transportation services. With these classical algorithms, additional models can be applied to achieve an efficient solution. For instance, the K-means clustering method can be combined with the TSP to determine tour paths [2]. Feiran Huang, Jie Xu, and Jian Weng proposed a multi-task deep travel route planning framework that works on establishing relations between users and points of interest and uses a network to learn their features [6]. This method focuses on gathering information based on the most visited places and the points of interest in a deep learning model. Shou-Chih Lo proposed a scan-based algorithm that arranges places with route smoothness [7]. This scan-based algorithm finds the shortest path with time-varying traffic data. With the Intelligent Travel Recommendation System, a collaborative filtering method is added based on the Restricted Boltzmann Machine with Hybrid K-means [3]. This system uses the Restricted Boltzmann Machine (RBM) and matrix factorization, along with a memory-based collaborative filtering technique. The RBM is a generative artificial neural network that uses unsupervised learning [3]. It consists of two layers of neurons – the visible layer, which represents the input, and the hidden layer, which represents the features learned by the neural network. Collaborative filtering is a recommendation technique that filters items based on similar user reactions [3]. It is an automatic filtering method that collects preferences or taste information from many users. Big data analysis has evolved and been implemented in many domains [8]. It utilizes the Hadoop Distributed File System (HDFS) for processing big data and MapReduce to calculate the potential value by assigning logical calculations to each dataset [8]. It operates on a master-slave architecture. Additionally, Reinforcement Learning (RL) algorithms such as Q-learning and Deep Q-networks (DQN) have been explored for optimizing route planning in dynamic environments where conditions change over time [9]. These RL algorithms learn optimal policies through interaction with the environment, enabling adaptive route planning strategies. With all these existing algorithms and models, our model uses the RBM framework with the collaborative filtering process and big data analysis to analyze and recommend the best route to travel.

III. THE PROPOSED SCHEME

The proposed scheme for leveraging the Restricted Boltzmann Machine (RBM) involves data preparation, model architecture definition, forward pass computation, reconstruction phase, training with Contrastive Divergence, evaluation, and deployment. Data is preprocessed and fed into the RBM, where the forward pass computes hidden node activations and the reconstruction phase reconstructs input data.

3.0 System Model:

The trip planning system begins with the user interface (UI), where users can access recommended trips from the system model and make adjustments to customize their trip plans. The system utilizes a combination of data analysis techniques and models to provide personalized recommendations and optimize trip plans.

The User Interface is the entry point for users to interact with the trip planning system. The UI provides a user-friendly interface for accessing recommended trips and making modifications to suit individual preferences. The trip analyzer module collects user preferences and requirements for trip planning. It utilizes big data analysis

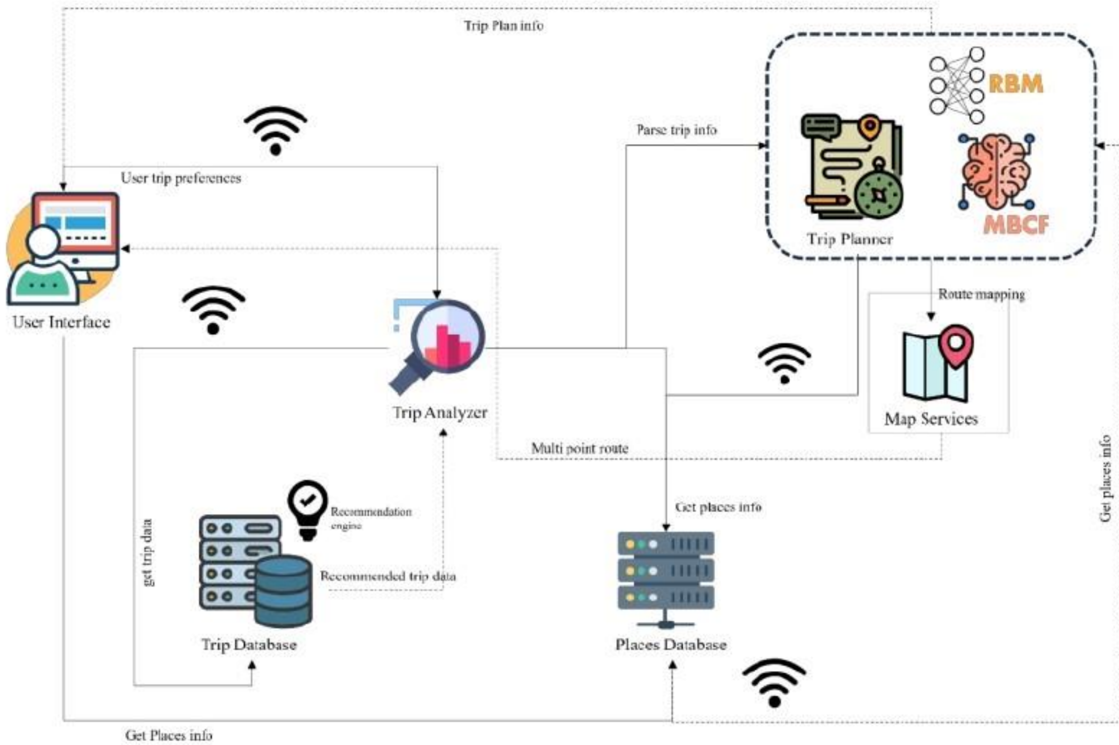


Figure 1: System Model

techniques to analyze historical trip data, user behavior, and external factors (such as weather, events, and traffic conditions) to generate best suitable personalized trip recommendations.

Big Data Analysis in Recommendation Stage:

In the recommendation stage, big data analysis techniques are employed to process large volumes of data, including historical trip data, user profiles, and contextual information, to generate personalized trip recommendations. This involves:

The system utilizes collaborative filtering techniques, such as Matrix Factorization or Restricted Boltzmann Machines (RBM), to analyze user preferences and similarities among users. By identifying patterns and similarities in user behavior, the system can recommend trips that are likely to be of interest to the user based on the preferences of similar users.

Content-based filtering techniques analyze the characteristics of trips and user preferences to generate recommendations. This involves analyzing features such as destination preferences, trip duration, budget constraints, and activities of interest to recommend trips that align with the user's preferences.

The trip database server stores a vast repository of trip data, including historical trip records, user profiles, and trip recommendations generated by the recommendation engine. It facilitates efficient data retrieval and storage, ensuring quick access to trip information during the planning process.

Embedded within the trip database server, the recommendation engine employs machine learning algorithms and big data analysis techniques to generate personalized trip recommendations for users. It leverages collaborative filtering, content-based filtering, and hybrid recommendation approaches to provide diverse and relevant trip options to users.

The trip planner module generates optimized trip plans based on user preferences and recommended trips from the recommendation engine. It utilizes optimization algorithms, such as the RBM framework and the Matrix Factorization model, to create trip itineraries that maximize user satisfaction while considering constraints such as budget, time, and user preferences.

Mapping services are integrated into the system to visualize trip plans and provide navigation assistance. These services utilize geospatial data and routing algorithms to generate multi-stop routes and optimize travel time and distance for the selected destinations.

The places database contains comprehensive information about various destinations, attractions, accommodations, and points of interest. It serves as a repository of data for the trip planner to retrieve detailed information about recommended places and incorporate them into trip plans.

Users have the option to provide feedback on their trip experiences and suggest new trip plans. The trip database stores user feedback and trip suggestions, which can be used to enhance the recommendation engine's performance and expand the database of trip options over time.

The system incorporates real-time data processing capabilities to handle streaming data sources, such as live traffic updates and event notifications, to provide users with up-to-date information and recommendations.

To accommodate the processing of large volumes of data and serve a growing user base, the system is designed to be scalable and performant, leveraging distributed computing frameworks and cloud-based infrastructure.

3.1 Matrix Factorization using Alternating least Squares technique (MF-ALS):

Matrix Factorization using Alternating Least Squares (MF-ALS) is a widely used Collaborative Filtering technique, particularly notable for its role in the Netflix Prize Challenge. This method leverages the user-item interaction matrix to learn embeddings representing the interactions, thereby predicting ratings for items that users have not yet rated. ALS, as a matrix factorization algorithm, operates efficiently in a parallel manner, making it suitable for large-scale collaborative filtering tasks. Its simplicity and ability to address sparsity in ratings data make it highly scalable and applicable to distributed file systems. For the MF-ALS model, hotel data containing user information, hotel details, and user ratings was utilized for training. Additionally, a 10% sample of new user data was generated based on hotel preferences, with the remaining 90% used to generate recommendations for these new users. The model was trained with varying ranks (latent factors) of 4, 8, and 12, along with parameters such as maximum iterations (5), regularization parameter (0.01), and attributes including user ID, hotel ID, and user ratings. Evaluation of the model was conducted using Root Mean Square Error (RMSE) for each rank setting, with the model exhibiting the lowest RMSE selected for recommendations. RMSE quantifies the disparity between predicted and actual ratings, with lower RMSE indicating superior learning of user-item interactions and more accurate predictions for unseen items. Overall, MF-ALS presents a robust approach for personalized recommendation systems, facilitating efficient learning and prediction of user preferences in a lower-dimensional latent space.

Let's denote:

R is the user-item interaction matrix, where Rij represents the rating given by user i to item j.

U is the user latent factor matrix, where each row represents a user and each column represents a latent feature.

V is the item latent factor matrix, where each row represents an item and each column represents a latent feature.

The objective of MF-ALS is to approximate the user-item interaction matrix R by the product of U and VT, such that R≈UVT.

The optimization problem can be formulated as minimizing the squared error between the observed ratings and the predicted ratings.

Mathematically, it can be represented as:

$$\min_{U,V} \sum_{(i,j)\in\Omega}(R_{ij} - U_i V_j^T)^2 + \lambda(||U||_F^2 + ||V||_F^2)$$

Where:

- Ω Is the set of observed ratings.
- ||·||F represents the Frobenius norm, which is the square root of the sum of the squares of all elements in the matrix.
- λ is the regularization parameter to prevent overfitting.

MF-ALS solves this optimization problem iteratively by alternating between updating U and V while fixing the other matrix.

- Update U:

$$U = (V^T V + \lambda I)^{-1} V^T R$$

- Update V:

$$V = (U^T U + \lambda I)^{-1} U^T R^T$$

These updates are performed iteratively until convergence. Each iteration involves alternating between updating U and V while keeping the other matrix fixed.

By decomposing the user-item interaction matrix into two lower-dimensional matrices U and V, MF-ALS effectively captures the latent factors underlying user preferences and item characteristics, allowing for accurate prediction of user-item ratings. This technique is widely used in recommendation systems due to its simplicity and effectiveness in handling large-scale datasets.

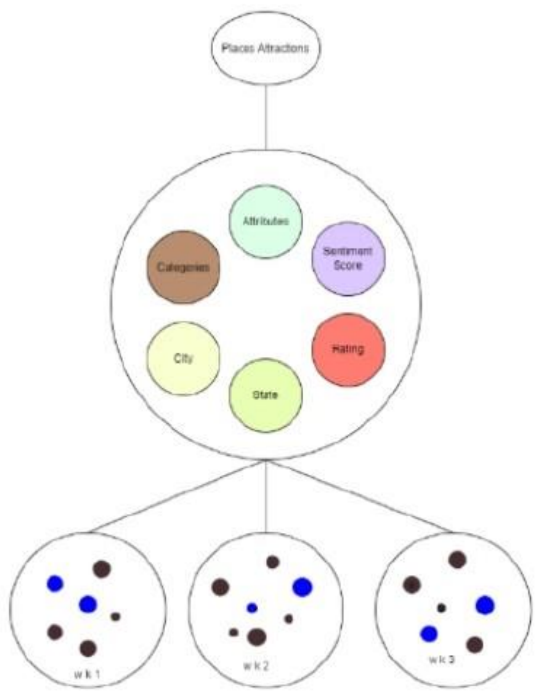


Figure 2 : Cluster Illustration

3.2 Restricted Boltzmann Machine (RBM):

The Restricted Boltzmann Machine (RBM) is a variant of Boltzmann machines, a class of stochastic and generative neural networks known for learning internal representations and solving complex combinatorial problems. RBMs consist of two types of nodes: visible and hidden nodes.

Unlike regular Boltzmann machines, RBMs have restricted connections, forming a fully bipartite structure with no connections within the same category of nodes. In RBMs, there are additional layers of bias units, including hidden bias and visible bias.

During the forward pass, the hidden bias activates the hidden nodes, while the visible bias aids in reconstructing the input during the backward pass. As there are no connections among visible units, RBMs rely on weights and biases to reconstruct input during the reconstruction phase, resulting in a different output from the actual input. The model is trained to minimize reconstruction error using a gradient-based algorithm called Contrastive Divergence.

It aims to reduce the discrepancy between the input and reconstructed output by updating the weights. Contrastive Divergence computes the difference between two Kullback-Leibler divergences of the input and reconstructed output, guiding the weight updates to bridge the gap between the two.

Overall, RBMs offer a powerful framework for learning internal representations and addressing complex combinatorial problems, leveraging stochasticity and generative capabilities to model intricate data distributions effectively.

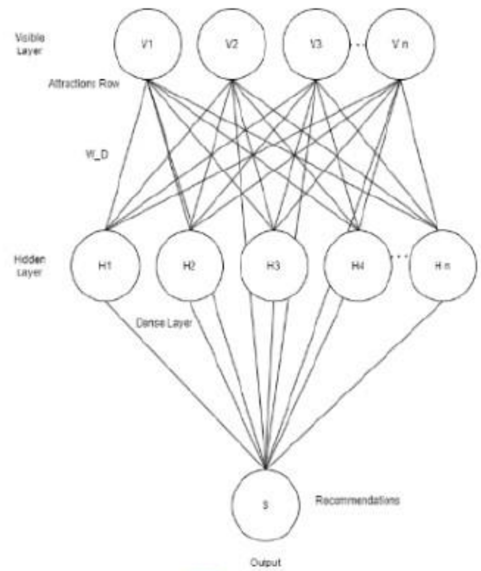


Figure 3: MLP Model

The energy function of an RBM is defined as:

$$E(\mathbf{v},\mathbf{h}) = -\sum_i \sum_j W_{ij} v_i h_j - \sum_i b_i v_i - \sum_j c_j h_j$$

Where:

- Wij is the weight connecting visible unit vi to hidden unit hj.
- bi is the bias term for visible unit vi.
- cj is the bias term for hidden unit hj.
- v and h are the states of the visible and hidden layers respectively.

The probability distribution of a visible vector v given a hidden vector h is defined by the following softmax function:

$$P(\mathbf{v}|\mathbf{h}) = \frac{1}{Z} \exp(-E(\mathbf{v},\mathbf{h}))$$

Where Z is the normalization constant (also called the partition function) that ensures the probabilities sum up to 1.

Similarly, the probability distribution of a hidden vector h given a visible vector v is:

$$P(\mathbf{h}|\mathbf{v}) = \frac{1}{Z} \exp(-E(\mathbf{v},\mathbf{h}))$$

The RBM learns by adjusting the weights and biases to minimize the difference between the observed data distribution and the distribution of the RBM's reconstructions. This is typically done using techniques like Contrastive Divergence, which iteratively updates the weights and biases to improve the model's ability to reconstruct the input data.

Overall, the RBM captures complex patterns in the data by learning the joint probability distribution of the visible and hidden units, which allows it to generate new samples and make useful inferences about the data.

IV. EXPERIMENT RESULTS

4.1 Efficiency of travel

The efficiency of travel refers to how effectively the recommended tour itinerary minimizes travel time and distance while maximizing time spent at desired destinations. This metric evaluates the system's ability to optimize the route planning process, ensuring that users can cover their desired destinations in the most time-efficient manner possible. The goal is to provide users with itineraries that allow them to make the most of their available time while minimizing unnecessary travel and delays.

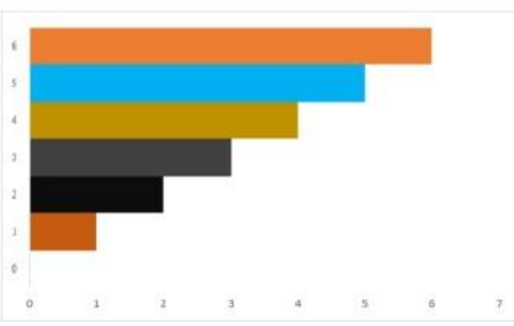


Figure 4: Time and Efficiency Chart

4.2 Accuracy of recommendations

The accuracy of recommendations refers to how closely the suggested tour itineraries match the preferences and expectations of the user. It assesses the system's ability to generate personalized and relevant travel plans based on user input and available data. The accuracy of recommendations can be measured by comparing the proposed itineraries with manually curated plans or with the user's own preferences and satisfaction with the suggested destinations, activities, and routes. A higher accuracy indicates that the system effectively understands and caters to the user's needs and desires, leading to a more fulfilling travel experience.

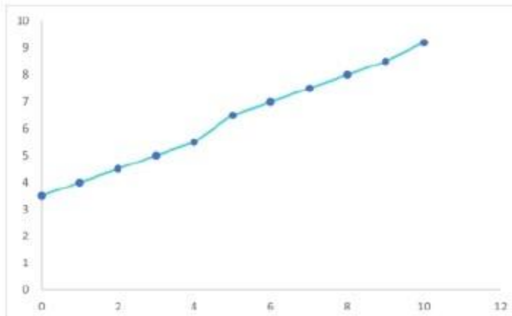


Figure 5: Ratings and Recommendation Chart

4.3 Analysis of feedback

Feedback analysis involves systematically collecting and interpreting user feedback to gain insights into the strengths and weaknesses of the tour planning application. It entails gathering comments, suggestions, and ratings provided by users regarding their experience with the system and using this feedback to identify areas for improvement. By analyzing user feedback, developers can understand user preferences, pain points, and areas of satisfaction, allowing them to iteratively

refine the application to better meet user needs. This process helps in enhancing user satisfaction, improving system performance, and driving continuous innovation.

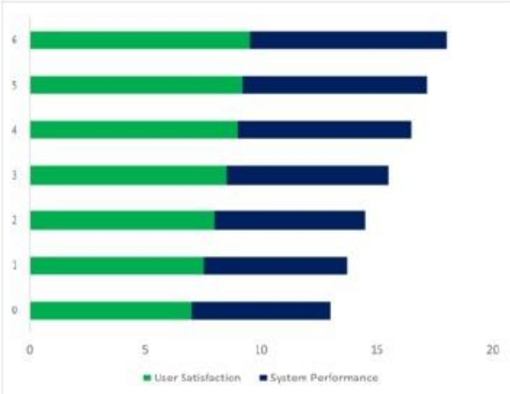


Figure 6: Time and User Satisfaction and System Performance Chart

4.4 Safety and reliability

The safety and reliability aspect involves tracking incidents like accidents, road closures, and traffic congestion along the recommended routes. The system aims to prioritize routes that ensure safe and reliable travel conditions for users, enhancing their overall experience and reducing potential risks during their journey.

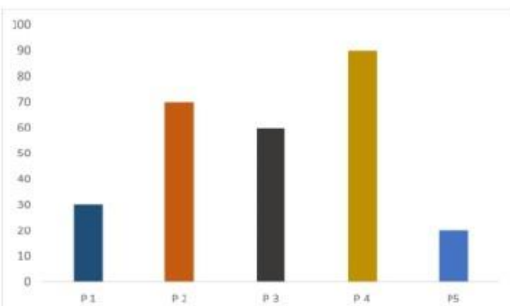


Figure 7: Places and Safety Chart

IV. CONCLUSION

In conclusion, our automated tour application represents a significant advancement in travel planning, offering users tailored itineraries that maximize their enjoyment and convenience while minimizing logistical challenges. By seamlessly integrating user preferences, budget considerations, optimal routes, and ideal visiting times, our application empowers travelers to embark on unforgettable journeys suited to their unique desires and constraints. Through rigorous data analysis and the implementation of cutting-edge algorithms, our system delivers superior tour plans irrespective of budget constraints, travel routes, or preferred visiting times. The incorporation of user feedback mechanisms ensures continual refinement and enhancement, guaranteeing that our recommendations evolve in tandem with user preferences and changing circumstances. Ultimately, our automated tour application revolutionizes the travel experience by providing unparalleled convenience, efficiency, and satisfaction. Whether exploring new destinations or rediscovering familiar locales, users can trust our application to deliver personalized itineraries that transform their travel dreams into reality.

REFERENCES

[1] Fan-Hsun Tseng (2020). "Optimal Route Planning System for Logistics Vehicles Based on Artificial Intelligence."

[2] Xiao Zhou, Mingzhan Su, Zhong Liu, De Zhang (2019). "Smart Tour Route Planning Algorithm based on Clustering Center Motive Iteration Search."

[3] H.H. Owaied, H.A Farhan, N. Al-Hawamdeh, N. Al-Okialy (2011). "A Model of Intelligent Tourism Guide System."

[4] Maobin Ding, Volume 2021, Article ID 2227798, 7 pages https://doi.org/10.1155/2021/2227798 "Research on Tourism Route Planning Based on Artificial Intelligence Technology."

[5] Takashi Hasuike, Hideki Katagiri, Hiroe Tsubaki, Hiroshi Tsuda (2013). "Personal Tour Planning Incorporating Standard Tour Routes and Tourist Satisfaction."

[6] Feiran Huang, Jie Xu, Jian Weng (2020). "Multi-Task Travel Route Planning with a Flexible Deep Learning Framework."

[7] Shou-Chih Lo, Fu-Chiang Huang, Chien-Chung Wang (2014). "Transport Route Planning for Mobile Tour Applications."

[8] Yuan Yao, Zhe Peng, Bin Xiao (2018). "Parallel Hyper-Heuristic Algorithm for Multi-Objective Route Planning in a Smart City."

[9] Kewad Rathod, Vinay Govekar, Mayuri Gaikwad, Ichhanshu Jaiswal, JETIR May 2021, Volume 8, Issue 5, "AI – Based Tourist Guide."

[10] Kai Wang, Xinping Yan, Yupeng Yuan, Xiali Jiang, Gabriel Lodewijks, Rudy R. Negenborn, 2017 4th International Conference on Transportation Information and Safety (ICTIS), August 8-10, 2017, Banff, Canada, " Study on Route Division for Ship Energy Efficiency Optimization Based on Big Environment Data."

[11] Wen Ouyang, Chang Wu Yu, Pei-Ju Huang, Huai-Tse Chang, Sixth International Conference on Future Generation Communication Technologies (FGCT 2017), " Non-Commutative path planning for tours with diversified attractions."

1258

ORIGINALITY REPORT

14%	7%	9%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Ramkumar Jayaraman, Karthikeyan C, A. Sahaya Anselin Nisha, K. Somasundaram, N. Naga Saranya, Vijendra Babu D. "Effective Location-based Recommendation Systems for Holiday using RBM Machine Learning Approach", 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS), 2023 Publication	2%
2	content.iospress.com Internet Source	1%
3	Lo, Shou Chih, Fu Chiang Huang, and Chien Chung Wang. "Transport Route Planning for Mobile Tour Applications", 2014 IEEE International Conference on Computer and Information Technology, 2014. Publication	1%
4	Wen Ouyang, Chang Wu Yu, Pei-Ju Huang, Huai-Tse Chang. "Non-commutative path planning for tours with diversified attractions", 2017 Sixth International	1%

Conference on Future Generation Communication Technologies (FGCT), 2017
Publication

5	Submitted to Loughborough University Student Paper	1%
6	Submitted to Liverpool John Moores University Student Paper	1%
7	Submitted to Macquarie University Student Paper	1%
8	Submitted to University of Nebraska at Omaha Student Paper	1%
9	www.bankofcanada.ca Internet Source	1%
10	www.mdpi.com Internet Source	1%
11	listens.online Internet Source	1%
12	scindeks-clanci.ceon.rs Internet Source	1%
13	Chaolong Zhang, Yigang He, Lifeng Yuan, Sheng Xiang. "Analog Circuit Incipient Fault Diagnosis Method Using DBN Based Features Extraction", IEEE Access, 2018 Publication	<1%

14	Submitted to Blackburn College, Lancashire Student Paper	<1%
15	Submitted to North Down and Ards Institute of Further and Higher Education Student Paper	<1%
16	Submitted to University of Florida Student Paper	<1%
17	Submitted to University of South Florida Student Paper	<1%
18	ouci.dntb.gov.ua Internet Source	<1%
19	www.hindawi.com Internet Source	<1%
20	www.igi-global.com Internet Source	<1%
21	J.-S. Park, M. Penner, V.K. Prasanna. "Optimizing graph algorithms for improved cache performance", IEEE Transactions on Parallel and Distributed Systems, 2004 Publication	<1%
22	Jun Ying, Joyita Dutta, Ning Guo, Chenhui Hu, Dan Zhou, Arkadiusz Sitek, Quanzheng Li. "Classification of Exacerbation Frequency in the COPDGene Cohort Using Deep Learning With Deep Belief Networks", IEEE Journal of Biomedical and Health Informatics, 2020	<1%

Publication

23	www.bmva.org Internet Source	<1%
24	Xinyu Li, Xiaoguang Gao, Chenfeng Wang. "A Novel Restricted Boltzmann Machine Training Algorithm With Dynamic Tempering Chains", IEEE Access, 2021 Publication	<1%
25	Mehdi Kargar, Zhibin Lin. "A socially motivating and environmentally friendly tour recommendation framework for tourist groups", Expert Systems with Applications, 2021 Publication	<1%
26	"Web and Big Data", Springer Science and Business Media LLC, 2018 Publication	<1%
27	dokumen.pub Internet Source	<1%
28	Lecture Notes in Computer Science, 2014. Publication	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		

PAPER PUBLICATION

Transformative Tour Planning with Dynamic Route Optimization using Big Data

Barathan K[a], Dhayanand G B[b], Bangarru Srijesh M[c]
Panimalar Engineering College, Chennai, India

Abstract— Planning a tour before embarking on a journey is crucial for a more fulfilling experience. Tour planning is a challenging task and lots of people scrap their tour journey due to ineffective planning. Sometimes it may lead to delays in travel and unsafe journeys etc. Our research solves this problem by autonomous tour planning with multiple destinations. Most route planning algorithms work on constraints like route traffic, road conditions, construction works, and non-safety roads. With all these constraints we include a better visiting time which will perfect the route planning with efficient travel and better viewing experience. With our vast dataset at your fingertips, you have the power to prioritize the places that intrigue you the most. It is all about tailoring your journey to your desires. This system weaves together your preferences, budgets, precious time, your residence's proximity, and the ideal mode of transport. With all this our system effortlessly devises the finest possible plans. The heart of our model is developed with the RBM (Restricted Boltzmann Machine) algorithm, with this deep-learning recommendation algorithm optimal plans are formulated by considering a multitude of factors including such as the ideal time for visitation, minimizing travel distances, and ensuring budget-friendly options. Through meticulous analysis, experimentation, and subsequent results, the system crafts a meticulously organized tour itinerary, ensuring an efficient and impactful journey. Notably redundant, the system operates with remarkable performance, further enhancing its reliability and effectiveness.

Keywords— *Tour Planning, Multiple destination, Ideal visiting time, User preferences, Tour budgets, Data analysis, Restricted Boltzmann Machine, Deep learning algorithm.*

INTRODUCTION

The tourism industry stands as a thriving sector in the contemporary global market. The best way to escape the holiday boredom is to plan a perfect tour vacation. However, the complications involved in planning, such as budget considerations and time constraints, often derail travel expectations. The lack of a well-structured plan further contributes to tour cancellations. Although the planning process demands a significant investment of time as one sits through numerous hotels, attractions, and restaurants. The exhaustive task involves scrutinizing ratings, reviews, and various attributes, all while trying to tailor the experience to fit within a specified budget. Crafting a vacation plan that comprehensively considers individual travel preferences without scouring through countless websites seems nearly unattainable.

Our project's focus is to manage the problematic planning area, allowing travellers to dedicate more time to enjoying their vacation. By gathering details such as destination, travel dates, budget, hotel amenities, preferred attraction categories, and cooking preferences, we aim to generate a personalized travel plan. This comprehensive plan will guide users on where to stay, suggest activities throughout the day, and recommend dining options for each meal. Using Data Science tools and techniques, our application aims to serve as a singular solution for travellers who need efficient vacation planning, along with budgetary control and best timing and spots, etc. It also incorporates budget-based tour planning which enables the user to visit places according to their financial plan and their income level. This application also offers users insights into the optimal timing and seasonality for visiting specific places based on their chosen destinations. By providing information on the best times to visit, users can enhance their overall travel experience. Moreover, the application can perform the duty of a guide by informing the users with a well-organized list of hotels and restaurants that are positioned along their selected routes. This feature serves the requirements of tourists by ensuring that they have convenient options for refreshment and rest throughout their journey. Furthermore, the application serves as a comprehensive travel companion by offering detailed transport-related information, including timings and vehicle types, for reaching their destinations, especially in cases where users don't have access to personal transportation. Whether it's public transport, ridesharing services, or other modes of transportation, the application seeks to present a thorough overview of available options by enhancing the overall accessibility and flexibility of travel plans. With the integrations of budget considerations, optimal timing recommendations, transport information, and a well-planned selection of accommodations and dining, this application aims to perfect the travel planning experience for users. This project aims to be a go-to tool for travellers planning their vacation which can be done by leveraging Data Science tools and techniques, which will be explained in this report.

RELATED WORKS

Route planning is a testament to human intelligence. In the olden days, people would travel independently, finding their way to their destinations. This process has evolved from mental calculations to complex problems. Some of the classical route or path planning problems include the Travelling Salesman Problem (TSP) [1], the Vehicle Routing Problem (VRP) [1], the Chinese Postman Problem [1], the Rural Postman Problem [1], Dijkstra’s algorithm [1], and the Bellman-Ford algorithm [1]. In the Traveling Salesperson Problem (TSP), an individual journeys through 'n' cities once and then returns to the initial city, aiming to minimize travel expenses. It is an NP-hard problem. Solutions can be discovered through the application of dynamic programming or branch-and-bound algorithms. For large-scale problems, genetic algorithms can be applied. There is also a Euclidean TSP, which focuses on measuring path cost based on Euclidean distance. It is also NP-hard. The VRP determines the optimal vehicle route for transportation, considering factors like minimum cost, distance, and travel time. It is an NP-complete problem. The Chinese Postman Problem focuses on the edges with the least cost to create the shortest route. This problem is adapted from the Rural Postman Problem, which considers a subset of edges. The Rural Postman Problem is an NP-complete problem. Dijkstra's algorithm is utilized to determine the shortest paths between nodes within a weighted graph. It is applicable when the edge weights are non-negative. The Bellman-Ford algorithm addresses the same problem but allows for negative edge weights. Compared to Dijkstra’s algorithm, the A* algorithm resolves the single-source shortest path issue by exploring a reduced search area. The more preferred classical algorithm is the Floyd-Warshall algorithm [1], resolves the problem of finding the shortest paths between all pairs of nodes and can be applied in transportation services. With these classical algorithms, additional models can be applied to achieve an efficient solution. For instance, the K-means clustering method can be combined with the TSP to determine tour paths [2]. Feiran Huang, Jie Xu, and Jian Weng proposed a multi-task deep travel route planning framework that works on creating connections between users and points of interest and uses a network to learn their features [6]. This method focuses on gathering information based on the most visited places and the points of interest in a deep learning model. Shou-Chih Lo proposed a scan-based algorithm that arranges places with route smoothness [7]. This scan-based algorithm finds the shortest path with time-varying traffic data. With the Intelligent Travel Recommendation System, a collaborative filtering method is added based on the Restricted Boltzmann Machine with Hybrid K-means [3]. This system uses the Restricted Boltzmann Machine (RBM) and matrix factorization, along with a memory-based collaborative filtering technique. The RBM is an artificial neural network that employs unsupervised learning and is capable of generating data [3]. It consists of two layers of neurons – the visible layer, which represents the input, and the hidden layer, which represents the features learned by the neural network. Collaborative filtering is a recommendation technique that filters items based on similar user reactions [3]. It is an automatic filtering method that collects preferences or taste information from many users. Big data analysis has evolved and been implemented in many domains [8]. It utilizes the Hadoop Distributed File System (HDFS) for processing big data and MapReduce to calculate the potential value by assigning logical calculations to each dataset [8]. It operates on a master-slave architecture. Additionally, Reinforcement Learning (RL) algorithms such as Q-learning and Deep Q-networks (DQN) have been explored for optimizing route planning in dynamic environments where conditions change over time [9]. These reinforcement learning algorithms acquire optimal policies by interacting with the environment, enabling adaptive route planning strategies. With all these existing algorithms and models, our model uses the RBM framework with the collaborative filtering process and big data analysis to analyze and recommend the best route to travel.

THE PROPOSED SCHEME

The proposed scheme for leveraging the Restricted Boltzmann Machine (RBM) involves data preparation, model architecture definition, forward pass computation, reconstruction phase, training with Contrastive Divergence, evaluation, and deployment. Data is preprocessed and fed into the RBM, where the forward pass computes hidden node activations and the reconstruction phase reconstructs input data.

System Model: The trip planning system begins with the user interface (UI), where users can access recommended trips from the system model and make adjustments to customize their trip plans. The system utilizes a combination of data analysis techniques and models to provide personalized recommendations and optimize trip plans.

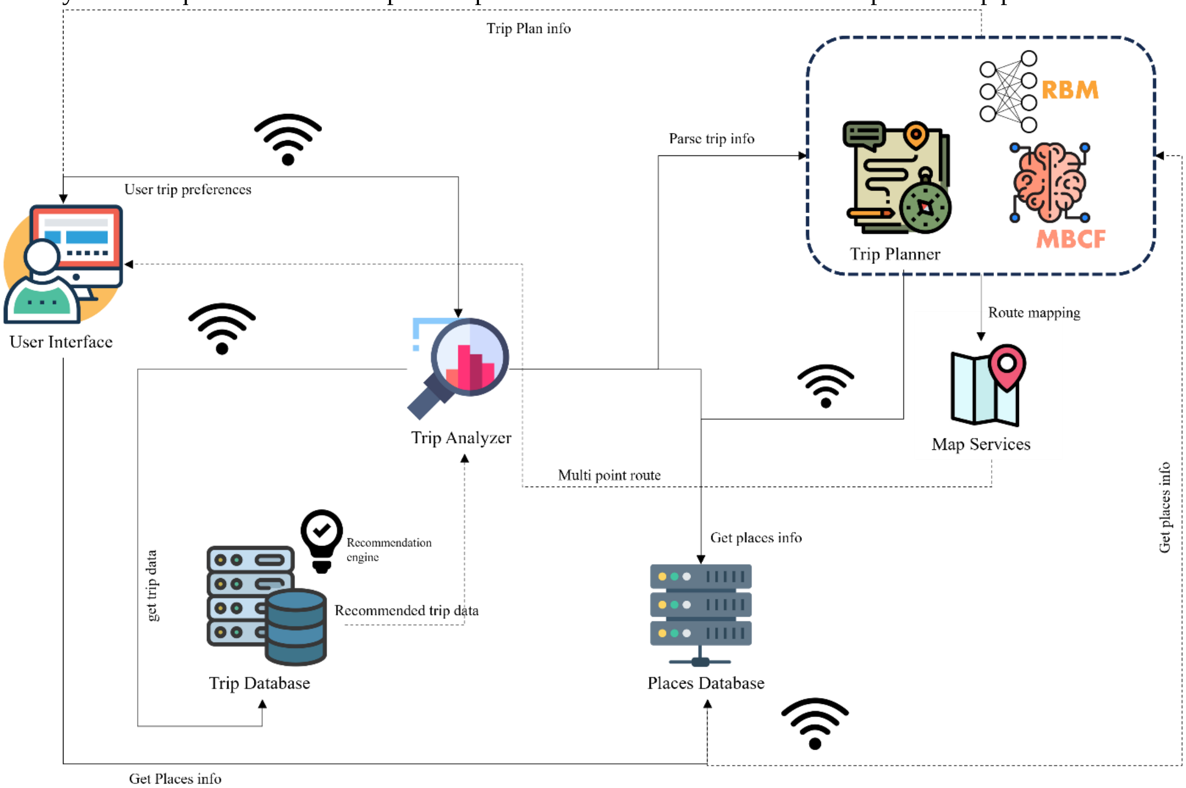


FIGURE 1: SYSTEM MODEL

The User Interface is the entry point for users to interact with the trip planning system. The UI provides a user-friendly interface for accessing recommended trips and making modifications to suit individual preferences. The trip analyzer module collects user preferences and requirements for trip planning. It utilizes big data analysis techniques to analyze historical trip data, user behavior, and external factors (such as weather, events, and traffic conditions) to generate best suitable personalized trip recommendations.

Big Data Analysis in Recommendation Stage: In the recommendation stage, big data analysis techniques are employed to process large volumes of data, including historical trip data, user profiles, and contextual information, to generate personalized trip recommendations. This involves:

The system utilizes collaborative filtering techniques, such as Matrix Factorization or Restricted Boltzmann Machines (RBM), to analyze user preferences and similarities among users. By identifying patterns and similarities in user behavior, the system can suggest trips that the user might like based on what other similar users have enjoyed.

Content-based filtering techniques analyze the characteristics of trips and user preferences to generate recommendations. This involves analyzing features such as destination preferences, trip duration, budget constraints, and activities of interest to recommend trips that align with the user's preferences.

The trip database server stores a vast repository of trip data, including historical trip records, user profiles, and trip recommendations generated by the recommendation engine. It facilitates efficient data retrieval and storage, ensuring quick access to trip information during the planning process. Embedded within the trip database server, the recommendation engine employs machine learning algorithms and big data analysis techniques to generate personalized trip recommendations for users. It uses collaborative filtering, content-based filtering, and a mix of both methods to offer users a wide range of trip options that are both diverse and relevant.

The trip planner module generates optimized trip plans based on user preferences and recommended trips from the recommendation engine. It utilizes optimization algorithms, such as the RBM framework and the Matrix Factorization model, to create trip itineraries that maximize user satisfaction while considering constraints such as budget, time, and user preferences.

Mapping services are integrated into the system to visualize trip plans and provide navigation assistance. These services utilize geospatial data and routing algorithms to generate multi-stop routes and optimize travel time and distance for the selected destinations. The places database contains comprehensive information about various destinations, attractions, accommodations, and points of interest. It serves as a repository of data for the trip planner to retrieve detailed information about recommended places and incorporate them into trip plans. Users have the option to provide feedback on their trip experiences and suggest new trip plans.

The trip database stores user feedback and trip suggestions, which can be used to enhance the recommendation engine's performance and expand the database of trip options over time. The system incorporates real-time data processing capabilities to handle streaming data sources, such as live traffic updates and event notifications, to provide users with up-to-date information and recommendations. To accommodate the processing of large volumes of data and serve a growing user base, the system is designed to be scalable and performant, leveraging distributed computing frameworks and cloud-based infrastructure.

Matrix Factorization using Alternating least squares technique (MF-ALS): Matrix Factorization using Alternating Least Squares (MF-ALS) is a widely used Collaborative Filtering technique, particularly notable for its role in the Netflix Prize Challenge. This method leverages the user-item interaction matrix to learn embeddings representing the interactions, thereby predicting ratings for items that users have not yet rated. ALS, as a matrix factorization algorithm, operates efficiently in a parallel manner, making it suitable for large-scale collaborative filtering tasks. Its simplicity and ability to address sparsity in ratings data make it highly scalable and applicable to distributed file systems. For the MF-ALS model, hotel data containing user information, hotel details, and user ratings was utilized for training. Additionally, a 10% sample of new user data was generated based on hotel preferences, with the remaining 90% used to generate recommendations for these new users. The model was trained with varying ranks (latent factors) of 4, 8, and 12, along with parameters such as maximum iterations (5), regularization parameter (0.01), and attributes including user ID, hotel ID, and user ratings. Evaluation of the model was conducted using Root Mean Square Error (RMSE) for each rank setting, with the model exhibiting the lowest RMSE selected for recommendations. RMSE quantifies the disparity between predicted and actual ratings, with lower RMSE indicating superior learning of user-item interactions and more accurate predictions for unseen items. Overall, MF-ALS presents a robust approach for personalized recommendation systems, facilitating efficient learning and prediction of user preferences in a lower-dimensional latent space.

Let's denote:

R is the user-item interaction matrix, where Rij represents the rating given by user i to item j.

Matrix U represents users' latent factors, with each row representing a different user and each column representing a particular characteristic.

Matrix V represents items' latent factors, with each row representing a different item and each column representing a specific characteristic.

The objective of MF-ALS is to approximate the user-item interaction matrix R by the product of U and VT, such that R≈UVT.

The optimization problem can be formulated as minimizing the squared error between the observed ratings and the predicted ratings.

Mathematically, it can be represented as:

$$\min u, v \sum (i, j) \in \Omega \left(R_{ij} - u_i v_j^T\right)^2 + \lambda(\|u\|_F^2 + \|v\|_F^2)$$

Where:

- Ω Is the set of observed ratings.
- ||·|| F represents Frobenius norm, which is calculated as the square root of the sum of the squares of all elements within the matrix.
- λ is the regularization parameter to prevent overfitting.

MF-ALS solves this optimization problem iteratively by alternating between updating U and V while fixing the other matrix.

- Update U:

  $U = (V^T V + \lambda I)^{-1} V^T R$

- Update V:

  $V = (U^T U + \lambda I)^{-1} U^T R^T$

These updates are performed iteratively until convergence. Each iteration involves alternating between updating U and V while keeping the other matrix fixed. By breaking down the matrix that shows how users interact with items into two smaller matrices U and V, MF-ALS effectively captures the hidden factors that influence user preferences and item characteristics, allowing for accurate prediction of user-item ratings.

This method is commonly employed in recommendation systems because of its simplicity and effectiveness in managing extensive datasets.

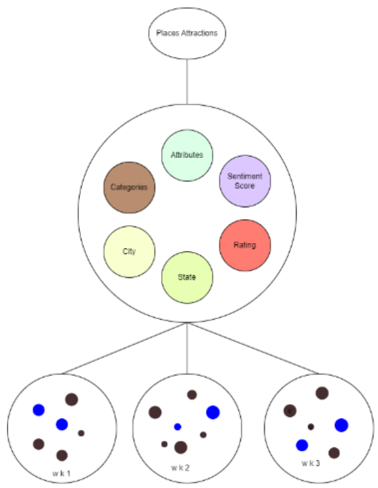


FIGURE 1 : CLUSTER ILLUSTRATION

Restricted Boltzmann Machine (RBM): The Restricted Boltzmann Machine (RBM) is a type of neural network that belongs to the Boltzmann machine family. These networks are adept at learning internal representations and tackling intricate combinatorial problems. RBMs comprise two kinds of nodes: visible nodes and hidden nodes.

Unlike regular Boltzmann machines, RBMs have restricted connections, forming a fully bipartite structure with no connections within the same category of nodes. In RBMs, there are extra layers of bias units, which include hidden bias units and visible bias units. During the forward pass, the hidden bias activates the hidden nodes, while the visible bias helps in reconstructing the input data during the backward pass. As there are no connections among visible units, RBMs rely on weights and biases to reconstruct input during the reconstruction phase, resulting in a different output from the actual input.

The model is trained to minimize reconstruction errors through a gradient-based optimization approach known as Contrastive Divergence. Its objective is to minimize the difference between the input and reconstructed output by adjusting the weights. Contrastive Divergence calculates the gap between two Kullback-Leibler divergences of the input and reconstructed output, directing the weight adjustments to narrow this difference.

Overall, RBMs offer a powerful framework for learning internal representations and addressing complex combinatorial problems, leveraging stochasticity and generative capabilities to model intricate data distributions effectively. The RBM learns by adjusting the weights and biases to minimize the difference between the observed data distribution and the distribution of the RBM's reconstructions.

This is typically done using techniques like Contrastive Divergence, which iteratively updates the weights and biases to improve the model's ability to reconstruct the input data.

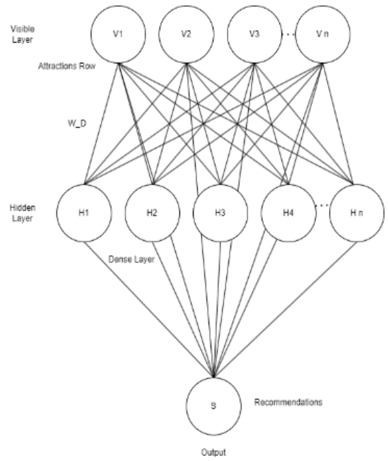


FIGURE 2: MLP MODEL

The energy function of an RBM is defined as:

$E(v,h) = -\Sigma_i \Sigma_j W_{ij} v_i h_j - \Sigma_i b_i v_i - \Sigma_j e_j h_j$

Where:

- W_{ij} represents the weight linking visible unit v_i to hidden unit h_j.
- b_i denotes the bias term for visible unit v_i.
- c_j signifies the bias term for hidden unit h_j.
- v and h indicate the states of the visible and hidden layers, respectively.

The probability distribution of a visible vector v given a hidden vector h is defined by the following SoftMax function:

$P_{(v|h)} = \frac{1}{Z} \exp\big(-E(v,h)\big)$

Where Z represents the normalization constant (also known as the partition function) ensuring that the probabilities sum up to 1. Similarly, the probability distribution of a hidden vector h given a visible vector v is:

$P_{(h|v)} = \frac{1}{Z} \exp\big(-E(v,h)\big)$

Overall, the RBM captures complex patterns in the data by learning the joint probability distribution of the visible and hidden units, which allows it to generate new samples and make useful inferences about the data.

EXPERIMENT RESULTS

Efficiency of travel: The efficiency of travel refers to how effectively the recommended tour itinerary minimizes travel time and distance while maximizing time spent at desired destinations. This metric evaluates the system's ability to optimize the route planning process, ensuring that users can cover their desired destinations in the most time-efficient manner possible. The goal is to provide users with itineraries that allow them to make the most of their available time while minimizing unnecessary travel and delays.

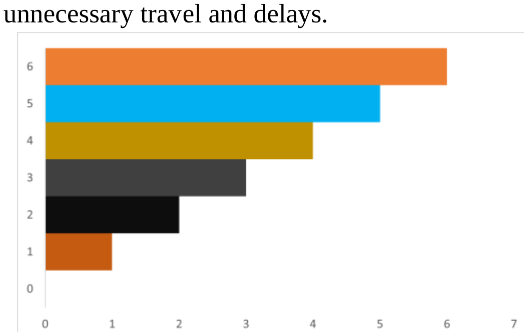


FIGURE 3: TIME AND EFFICIENCY CHART

Accuracy of recommendations: The accuracy of recommendations refers to how closely the suggested tour itineraries match the preferences and expectations of the user. It assesses the system's ability to generate personalized and relevant travel plans based on user input and available data. The accuracy of recommendations can be measured by comparing the proposed itineraries with manually curated plans or with the user's own preferences and satisfaction with the suggested destinations, activities, and routes. A higher accuracy indicates that the system effectively understands and caters to the user's needs and desires, leading to a more fulfilling travel experience.

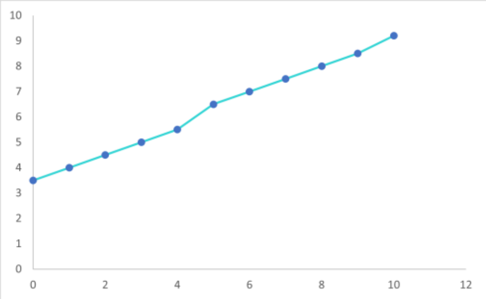


FIGURE 4: RATINGS AND RECOMMENDATION CHART

Analysis of feedback: Feedback analysis involves systematically collecting and interpreting user feedback to gain insights into the strengths and weaknesses of the tour planning application. It entails gathering comments, suggestions, and ratings provided by users regarding their experience with the system and using this feedback to identify areas for improvement. By analyzing user feedback, this process allows developers can understand user preferences, pain points, and areas of satisfaction, allowing them too iteratively refine the application to better meet user needs. This process helps in enhancing user satisfaction, improving system performance, and driving continuous innovation.

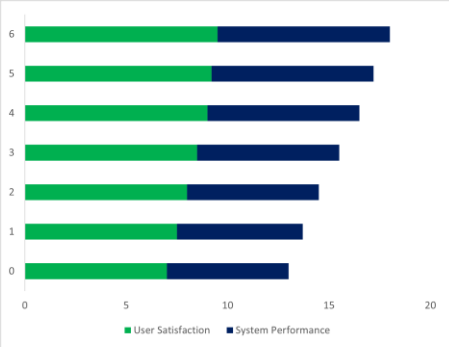


FIGURE 5: TIME AND USER SATISFACTION AND SYSTEM PERFORMANCE CHART

Safety and reliability: The safety and reliability aspect involve tracking incidents like accidents, road closures, and traffic congestion along the recommended routes. The system aims to prioritize routes that ensure safe and reliable travel conditions for users, enhancing their overall experience and reducing potential risks during their journey.

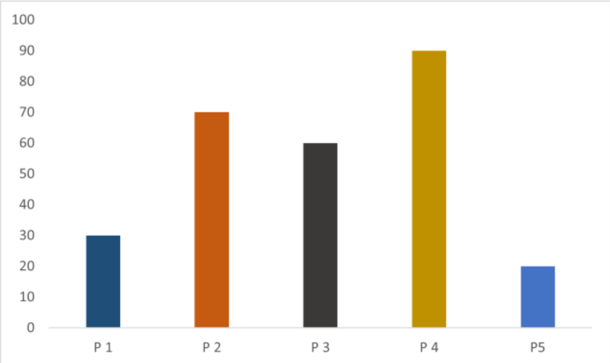


FIGURE 6: PLACES AND SAFETY CHART

CONCLUSION

In conclusion, our automated tour application represents a significant advancement in travel planning, offering users tailored itineraries that maximize their enjoyment and convenience while minimizing logistical challenges. By seamlessly integrating user preferences, budget considerations, optimal routes, and ideal visiting times, our application empowers travelers to embark on unforgettable journeys suited to their unique desires and constraints. Through rigorous data analysis and the implementation of cutting-edge algorithms, our system delivers superior tour plans irrespective of budget constraints, travel routes, or preferred visiting times. The incorporation of user feedback mechanisms ensures continual refinement and enhancement, guaranteeing that our recommendations evolve in tandem with user preferences and changing circumstances. Ultimately, our automated tour application revolutionizes the travel experience by providing unparalleled convenience, efficiency, and satisfaction. Whether exploring new destinations or rediscovering familiar locales, users can trust our application to deliver personalized itineraries that transform their travel dreams into reality.

REFERENCES

[1] Fan-Hsun Tseng (2020). "Optimal Route Planning System for Logistics Vehicles Based on Artificial Intelligence."

[2] Xiao Zhou, Mingzhan Su, Zhong Liu, De Zhang (2019). "Smart Tour Route Planning Algorithm based on Clustering Center Motive Iteration Search."

[3] H.H. Owaied, H.A Farhan, N. AI- Hawamdeh, N. AI-Okialy (2011). "A Model of Intelligent Tourism Guide System."

[4] Maobin Ding, Volume 2021, Article ID 2227798, 7 pages https://doi.org/10.1155/2021/2227798 "Research on Tourism Route Planning Based on Artificial Intelligence Technology."

[5] Takashi Hasuike, Hideki Katagiri, Hiroe Tsubaki, Hiroshi Tsuda (2013). "Personal Tour Planning Incorporating Standard Tour Routes and Tourist Satisfaction."

[6] Feiran Huang, Jie Xu, Jian Weng (2020). "Multi-Task Travel Route Planning with a Flexible Deep Learning Framework."

[7] Shou-Chih Lo, Fu-Chiang Huang, Chien-Chung Wang (2014). "Transport Route Planning for Mobile Tour Applications."

[8] Yuan Yao, Zhe Peng, Bin Xiao (2018). "Parallel Hyper-Heuristic Algorithm for Multi-Objective Route Planning in a Smart City."

[9] Kewad Rathod, Vinay Govekar, Mayuri Gaikwad, Ichhanshu Jaiswal, JETIR May 2021, Volume 8, Issue 5, "AI – Based Tourist Guide."

[10] Wen Ouyang, Chang Wu Yu, Pei-Ju Huang, Huai-Tse Chang, Sixth International Conference on Future Generation Communication Technologies (FGCT 2017), "Non-Commutative path planning for tours with diversified attractions."

[11] Kai Wang, Xinping Yan, Yupeng Yuan, Xiali Jiang, Gabriel Lodewijks, Rudy R. Negenborn, 2017 4th International Conference on Transportation Information and Safety (ICTIS), August 8-10, 2017, Banff, Canada, " Study on Route Division for Ship Energy Efficiency Optimization Based on Big Environment Data."

Gmail Barathan karthikeyan <barathankarthikeyan06@gmail.com>

Acceptance of Paper ID #1258 for ICONIC 2K24 Presentation

1 message

PECTEAM 2K24 <pecconference2024@gmail.com> Mon, Mar 18, 2024 at 11:43 AM
To: Dhayanand bhoopalan <dhayanand182@gmail.com>, Bangarru Murali <bangarrumurali@gmail.com>, Barathan karthikeyan <barathankarthikeyan06@gmail.com>

Dear Authors,

Congratulations on the acceptance of your paper ID #**1258** titled **"Transformative Tour Planning with Dynamic Route Optimization using Big Data "** for oral presentation at **ICONIC 2K24**. We appreciate your contribution to the conference. To proceed with the publication process, please carefully go through the attached reviewer comments and make necessary modifications to address the identified deficiencies in your paper. Ensure that the corrected version follows the CRP (Camera-Ready Paper) format provided in the websites.

Submission Guidelines:

➤ Upload the CAMERA-READY version of your paper along with a "Response to Reviewer Comments" addressing all the comments received by the reviewers.
➤ Strictly adhere to the template provided on the website; no other styles are allowed.
➤ The plagiarism report is attached below. Maintain a similarity index of less than 15% and ensure there is no AI-generated content in the paper.
➤ Register for the conference before 20th March 2024, using the provided registration link below:
https://docs.google.com/forms/d/e/1FAIpQLScMCbClrtyzSuIzkuAA2V4Xu8HKw53rR_zREbWGiP5mttO34Q/viewform?usp=sf_link

➤ For Camera Ready Paper (CRP) format, please visit https://pecteam.co.in/.

Please note that your registration becomes valid after your payment. View registration details and process at 7th INTERNATIONAL CONFERENCE on INTELLIGENT COMPUTING(https://pecteam.co.in/)

ICONIC 2K24

The 7th International Conference on Intelligent Computing (IConIC 2K24) took place on March 22nd and 23rd, 2024, in Chennai, India. The conference likely focused on the latest advancements and research in the field of intelligent computing, bringing together experts, researchers, and professionals from around the world to share insights, exchange ideas, and discuss emerging trends. Topics covered during the conference could include artificial intelligence, machine learning, data mining, computational intelligence, and related areas. Furthermore, the conference might have featured special sessions or tracks focusing on emerging trends such as deep learning, natural language processing, explainable AI, and the ethical implications of intelligent systems. evolving landscape of intelligent computing.