A PROJECT REPORT ON
TRAVEL PLANNER

A major project report submitted in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology

In

Computer Science and Engineering

Submitted by

BISWAJIT BISWAL	**2001292257**
NISHANT S SAHOO	2001292108
SATYABRATA BRAHMACHARY	2001292126
SHISHIR KUMAR MOHANTA	2001292130

Under the supervision of

Asst. Prof. Sarmistha Nanda

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
GANDHI ENGINEERING COLLEGE, BHUBANESWAR
Gandhi Vihar, Madanpur, Bhubaneswar-752054
2020-2024

CERTIFICATE

This is to certify that the work in the Project entitled '**Travel Planner'** by **Biswajit Biswal**, bearing ID **2001292257**, is a record of an original research work carried out under my supervision and guidance in fulfilment of the requirements for the award of the Bachelor's Degree in Technology in Computer Science & Engineering. Neither this project nor any part has been submitted for any degree or academic award elsewhere.

To the best of my knowledge, **Biswajit Biswal** bears a good moral character and decent behaviour.

Mrs. Sarmistha Nanda
Assistant Professor,
Department of Computer Science & Engineering,
Gandhi Engineering College,
Bhubaneswar

ACKNOWLEDGEMENT

Our sincere thanks to **Dr. Dushmanta Kumar Padhi**, Professor and Head of the Department of Computer Science Engineering, Gandhi Engineering College (GEC), Bhubaneswar, for his encouragement and valuable suggestions during the period of our Project work.

No words would suffice to express my regards and gratitude to **Asst. Prof. Sarmistha Nanda**, Department of Computer Science Engineering, for her inspiring guidance and constant encouragement, immense support and help during the project work.

Place: Bhubaneswar
Date: 20.04.2024

Biswajit Biswal
2001292257

ABSTRACT

The travel planner project aims to streamline the trip planning process by providing users with a user-friendly platform. The system allows users to input their destination, preferences, and budget, generating personalized itineraries that include transportation, accommodation, and activities. Additionally, users can modify their plans, view real-time travel information, and access recommendations. The project integrates data from various sources to enhance accuracy and provide a comprehensive travel experience, ultimately facilitating efficient and enjoyable trip planning for users.

The proposed system is highly automated and makes the travelling activities much easier and flexible. The user can get the right information at the very right time. This system will include all the necessary fields which are required during online reservation time. This system will be easy to use and can be used by any person. The basic idea behind this project is to save data ina central database which can be accessed by any authorized person to get information and savestime and burdens which are being faced by their customers.

There is a need for a comprehensive and user-friendly Travel Planner Web Application that notonly assists in creating desired but also provides intelligent recommendations based on user preferences and real-time data as well as this application will recommend the nearby trend place of the client with all Functionality (distance, time, booking, cost, etc…).

LIST OF FIGURES

Figure Number	Title	Page Number
3.1	Data Flow	17
3.2	Process component	17
3.3	External Entity	17
3.4	Datastore	17
3.5	Zero Level DFD	18
3.6	First Level DFD	19
3.7	Second Level DFD	20
3.8	Terminator	24
3.9	Flow Process	24
3.10	Decision Symbol	24
3.11	Flow Lines	24
3.12	Input / Output Data	25
3.13	Predefined Process	25
3.14	Database	25
3.15	Flowchart	26
3.16	ER Diagram	27
4.1	PERT	37
4.2	GANTT Chart	38
5.1	Landing page	42
5.2	Recommendation	42
5.3	Destination Suggestion	43
5.4	Login page	44
5.5	Sign Up page	44
5.6	Payment Section	45
5.7	Database of user id	47
5.8	Database of user profile	48
5.9	Database of pictures	48
5.10	Database of guides	49
5.11	Database of user bookings	50

LIST OF TABLES

Table Number	Title	Page Number
3.1	User IDs	21
3.2	Pictures	21
3.3	User Profile	22
3.4	Guide Data	22
3.5	User Bookings	23

TABLE OF CONTENTS

Chapters	Description	Page Number
Chapter -1	**Introduction to project**	
	1.1 Introduction	1
	1.2 Project Category	1
	1.3 Objectives	2
	1.4 Problem Statement	2
	1.5 Identification of Need	2
	1.6 Existing System	3
	1.7 Proposed System	4
	1.8 Key Features	5
Chapter-2	**Requirement Analysis and System Specification**	
	2.1 Feasibility Study	6
	2.1.1. Technical Feasibility	6
	2.1.2. Economical Feasibility	6
	2.1.3. Operational Feasibility	6
	2.2 Software Requirement Specification	7
	2.2.1. Data Requirement	7
	2.2.2. Functional Requirement	8
	2.2.3. Performance Requirement	8
	2.2.4. Dependability Requirement	8
	2.2.5. Maintainability Requirement	9
	2.2.6. Security Requirement	10
	2.2.7. Look And Feel Requirement	11
	2.3 Validation	12
	2.3.1. Functionality Validation	12
	2.3.2. Usability Validation	12
	2.3.3. Performance Validation	12
	2.3.4. Security Validation	12
	2.3.5. Compatibility Validation	12
	2.4 Expected Challenges	13
	2.5 SDLC model to be used	14
Chapter-3	**System Design**	
	3.1 Design Approach	15
	3.2 System Design using various Structure Analysis	16
	3.2.1 Data Flow Diagram	17
	3.2.1.1 Zero Level DFD	18
	3.2.1.2 First Level DFD	19
	3.2.1.3 Second Level DFD	20

3.2.2 Data Dictionary 21

3.2.3 Flowchart 24

3.3 ER Diagrams 27

3.4 Normalization 28

3.5 Database Manipulation 28

3.6 Database Connection Controls & String Methodology 29

Chapter-4 Implementation and Testing

4.1 Introduction to Languages 31

4.1.1 Scripting Language 31

4.1.2 Programming Language 32

4.1.3 API 33

4.1.4 IDE to be used 34

4.1.5 Technology to be used for implementation 34

4.1.6 Tools to be used 35

4.2 Coding Standards of the languages used 36

4.3 Project Scheduling using various tools 37

4.3.1 PERT 37

4.3.2 GANTT Chart 38

4.4 Testing Technology and Test Plan 39

Chapter-5 Results and Discussions

5.1 User Interface Representation 40

5.2 Brief Description of Various Modules of the system 40

5.3 Snapshots of system with brief detail of each 42

5.3.1 The Homepage or landing section 42

5.3.2 Destination Suggestion section 43

5.3.3 Sign-Up and Log-In section 44

5.3.4 Payment section 45

5.4 Back Ends Representation 46

5.5 Snapshots of Database Tables with brief description 47

Chapter-6 Conclusions & Future Scope 51

Chapter-7 References 53

INTRODUCTION TO THE PROJECT

1.1 Introduction to project

Embark on a seamless journey of exploration and adventure as we assist you in planning your next travel escapade. Whether you're a globetrotter or a first-time traveller, our platform is designed to simplify the planning process and make your travel experience unforgettable.
Let's map out your itinerary, discover new destinations and create memories that will last a lifetime. It also provides intelligent recommendations based on user preferences and real-time data.

The Travel Planner project is an innovative application designed to streamline and enhance the travel planning experience for users. With a user-friendly interface, the project aims to provide a comprehensive platform where individuals can effortlessly plan and organize their trips. The application facilitates itinerary creation, accommodation booking, and activity scheduling, offering users a centralized hub for all their travel needs. Leveraging advanced algorithms, the Travel Planner suggests personalized recommendations based on user preferences, ensuring a tailored and memorable travel experience. Through seamless integration with mapping services and real-time updates, users can navigate their destinations efficiently. The project not only simplifies travel logistics but also fosters community engagement by allowing users to share itineraries, tips, and reviews. Overall, the Travel Planner project is poised to redefine the way users plan and embark on their journeys, making travel more accessible, enjoyable, and interconnected.

1.2 Project Category

- The "Travel Planner" project falls under the Internet-based category, emphasizing its reliance on online connectivity and digital platforms to facilitate travel-related activities.
- In this context, an Internet-based Travel Planner typically involves a web or mobile application that allows users to plan, organize, and manage various aspects of theirtravel experiences.
- Users can access the service through the internet to research destinations, book accommodations and explore points of interest.
- The project may integrate features such as real-time updates, social sharing, and collaboration tools to enhance the overall travel planning and execution process.
- It leverages the connectivity of the online environment to provide users with a seamless and convenient way to navigate the complexities of travel, making the experience more efficient and enjoyable.

1.3 Objectives

- To develop a system that automates the processes and activities of a travel, and the purpose is to design a system using which one can perform all operations related to travelling.
- To develop a comprehensive and user-friendly platform that empowers travellers to efficiently plan and organize their trips, ensuring a seamless and enjoyable travel experience.
- To provide a centralized solution that caters to the diverse needs of travellers by offering features such as itinerary creation, destination information, accommodation recommendations and real-time updates.
- To streamline the travel planning process, enhance user satisfaction, and facilitate informed decision-making, ultimately fostering a sense of confidence and excitement among users as they embark on their journeys.
- To leverage modern technology to simplify complex travel arrangements, promote exploration, and contribute to memorable and stress-free travel adventures for individuals and groups alike.

1.4 Problem Statement

Design a travel planner system that allows users to efficiently plan and organize their trips. The system should enable users to input their destination, preferred activities, budget, and duration of stay. The planner should then generate a personalized itinerary, suggesting places to visit, activities to do, and accommodations within the specified budget. Users should be able to modify their preferences and receive updated itineraries dynamically. Consider factors such as travel time between locations, opening hours of attractions, and real-time updates. The goal is to create a user-friendly and flexible travel planning system that optimizes the overall travel experience.

1.5 Identification of Need

- The planning and execution of travel is significantly more difficult and time-consuming. In India, more than 6 million tourists travel to various places within a year. In past years, tourists had to rely on outdated guidebooks, travel agents with limited knowledge of specific destinations, and word-of-mouth recommendations. In recent years, tourism websites play a crucial role in facilitating smooth and enjoyable travel experiences for tourists by addressing a wide range of their problems and needs.
- However, Traditional tourism websites often offer generic recommendations that don'tconsider the user's unique interests and needs. AI can personalize recommendations by analysing the user's past travel history, search queries, and other data points. This can leadto a more enjoyable

and satisfying travel experience. If a user enjoys a particular place, it is very difficult to find other similar places that they might also like. AI can analyse the user's preferences and suggest other places that share similar characteristics, such as climate, landscape, activities, and cultural offerings. AI recommendations can help by filteringthrough the noise and suggesting places and activities that are tailored to the user's individual preferences.

1.6 Existing System

1. Make My Trip:

Make My Trip is a popular online travel platform that provides a one-stop solution for all travel-related needs. It offers services such as flight bookings, hotel reservations, holiday packages, and bus bookings. Users can plan their entire trip, including transportation and accommodation, through a user-friendly interface. The platform also provides real-time updates on flight statuses, hotel reviews, and travel guides to help users make informed decisions.

2. Ease My Trip:

Easy My Trip is another platform in the online travel industry that emphasizes simplicity and cost-effectiveness. It offers services like flight bookings, hotel reservations, and holiday packages. The platform distinguishes itself by providing transparent pricing and discounts, making it easy for users to find budget-friendly travel options. Easy My Trip aims to simplify the travel planning process and ensure a hassle-free experience for its users.

3. TripAdvisor:

TripAdvisor is a renowned travel and restaurant website that enables users to read and write reviews about hotels, restaurants, and attractions. The platform also offers a forum for users to ask and answer travel-related questions. The existing system of TripAdvisor emphasizes user- generated content, relying on the experiences and opinions of a vast community of travellers. The platform incorporates a rating system that helps users quickly assess the quality of differenttravel-related services.

Challenges in the Existing System:

- Information Overload: With a plethora of options, users may feel overwhelmed with information, making it challenging to make quick decisions.
- Integration Concerns: Coordinating multiple services, such as flights, hotels, and events.
- Competitive Pricing: Maintaining competitive pricing while ensuring quality services canbe a delicate balance.

1.7 Proposed System

The proposed system for a travel planner project aims to provide users with a comprehensive and user-friendly platform to plan and manage their travel itineraries. The system will leverage modern technologies and incorporate key features to enhance the overall travel planning experience. Here's an outline of the proposed system:

- **User Registration and Profile Creation:**
  Users can create accounts to access personalized features. The profile will include preferences, past travel history, and other relevant information to tailor the travel planning experience.
- **Intuitive User Interface:**
  A user-friendly interface will facilitate easy navigation and interaction. The design will be intuitive, ensuring that users can input and modify travel details effortlessly.
- **Travel Itinerary Creation:**
  Users can create detailed travel itineraries, specifying destinations, dates, and activities. The system will support collaborative planning, allowing multiple users to contribute to and modify a shared itinerary.
- **Continuous Improvement:**
  The system will be designed for continuous improvement based on user feedback and evolving travel trends. Regular updates and new features will be introduced to enhance the overall user experience.
- **Feedback and Reviews:**
  Users can leave feedback and reviews for hotels, restaurants, and activities, contributing to a community-driven system. Future travellers can benefit from these reviews when making decisions.

1.8 Unique features of the system

A travel planner system is designed to assist users in organizing and managing their travel itineraries efficiently. Several unique features contribute to the effectiveness and appeal of sucha system. Here are some notable aspects:

- **User-Friendly Interface:**

  An intuitive and user-friendly interface is crucial for a travel planner system. It should allow users to easily navigate through the website, input travel details, and access relevant information.

- **Multi-Platform Accessibility:**

  A robust travel planner should be accessible across various platforms, including web browsers, mobile devices, and even wearable technology. This ensures that users can plan and manage their trips conveniently, regardless of the device they are using.

- **Integration with External Data Sources:**

  To enhance the user experience, the travel planner can integrate with external data sources such as location based on IP address. This helps users make informed decisions during their travels.

- **Feedback and Reviews:**

  A system that allows users to leave feedback and reviews for hotels, restaurants, and activities helps future travellers make more informed decisions. This feature fosters a sense of community and trust within the travel planner platform.

REQUIREMENT ANALYSIS AND SYSTEM SPECIFICATION

2.1 Feasibility Study

A feasibility analysis involves a detailed assessment of the need, value and practicality of a proposed enterprise, such as systems development. The process of designing and implementing record keeping systems has significant accountability and resource implications for an organization. Feasibility analysis will help you make informed and transparent decisions at crucial points during the developmental process to determine whether it is operationally, economically and technically realistic to proceed with a particular course of action.

A feasibility study sometimes called a feasibility analysis or feasibility report is a way to evaluate whether or not a project plan could be successful. A feasibility study evaluates the practicality of your project plan in order to judge whether or not you're able to move forward with the project.

Most feasibility studies are distinguished for both users and analysts. First, the study often presupposes that when the feasibility document is being prepared, the analyst is in a position to evaluate solutions. Second, most studies tend to overlook the confusion inherent in system development - the constraints and the assumed attitudes.

2.1.1 Technical Study:

Technical feasibility centres around the existing computer system (hardware, software, etc.) and to what extend it can support the proposed addition. For example, if the current computer is operating at 80 percent capacity an arbitrary ceiling, then running another application could overload the system or require additional hardware. This involves financial considerations to accommodate technical enhancements. If the budget is a serious constraint, then the project is judged not feasible.

2.1.2 Economical Study:

Economic analysis is the most frequently used method for evaluating the effectiveness of a candidate system. More commonly known as cost benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits overweigh costs, then the decision is made to design and implement the system. Otherwise. further justification or alterations in the proposed system will have to be made if it is to have a chance of being approved. This is an ongoing effort that improves in accuracy at each phase in the system life cycle.

2.1.3 Operational Study:

People are inherently resistant to change, and computers have been known to facilitate change. An estimate should be made of how strong a reaction the user staff is likely to have toward the development of a computerized system. It is common knowledge that computer installations have something to do with turnover, transfers, retraining, and changes in employee job status. Therefore, it is understood that the introduction of a candidate system requires special effort to educate, sell and train the staff on new ways of conducting business.

2.2 Software Requirement Specification Document

The Requirements phase of a project involves gathering, documenting, and clarifying the needs and expectations of stakeholders. It establishes a foundation for the project by defining what the system or product should accomplish, specifying features, constraints, and functionality. This phase is crucial for ensuring a common understanding among team members and stakeholders before moving on to design and development.

2.2.1 Data Requirements:

The success of a travel planner project relies heavily on the effective management and utilization of data. Here's an overview of the data requirements for a comprehensive travel planner:

1. User Profile Data:

- User authentication information.
- Personal preferences such as travel style, interests, and preferred destinations.
- Historical travel data to provide personalized recommendations.

2. Destination Information:

- Detailed data about various travel destinations.
- Points of interest, landmarks, and attractions.

3. Activities and Attractions:

- Detailed information on various activities and attractions at the destination.
- Timings, ticket prices, and any special requirements.
- User reviews and recommendations.

4. Geographical and mapping data:

- Maps and navigation tools.
- Geographic data for plotting itineraries and locating points of interest.

5. Analytics and Reporting Data:

- Usage analytics to understand user behaviour.

- Reporting tools for generating insights into popular destinations and user preferences.

6. User-generated Content:

- User reviews and ratings for destinations, accommodations, and activities.
- User-generated travel itineraries and trip reports.

2.2.2 Functional Requirements:

1. The admin's privileges are role-based access control for security.
2. Each user has a unique credential which is stored encrypted in the database for security.
3. Users of the system can view different places as per their current location.
4. Users can book guide for tours with online booking and payment system.
5. Each user can get information about travel accommodations of each place.

2.2.3 Performance Requirements:

Performance requirements define how well the software system accomplishes certain functions under specific conditions. Examples include the software's speed of response, throughput, execution time and storage capacity. The service levels comprising performance requirements are often based on supporting end-user tasks.

1. The system must not lag, because the users using it don't have down-time to wait for it to complete an action.
2. The system must complete updating the databases, adding of data, manipulating data, return and update user status successfully every time the user requests such a process.
3. All the functions of the system must be available to the user every time the system is turned on.
4. The calculations performed by the system must comply according to the norms set by the user and should not vary unless explicitly changed by the user.

2.2.4 Dependability Requirements:

The travel planner project must ensure a high level of dependability, including:

1. **Reliability:** The system should consistently provide accurate and relevant travel information, minimizing errors or disruptions in service.
2. **Availability:** The travel planner should be available and accessible to users at all times, with minimal downtime for maintenance or unforeseen issues.
3. **Fault Tolerance:** The system should be resilient to failures, ensuring that a single point of failure does not compromise the entire functionality of the travel planner.
4. **Data Integrity:** All user data, including travel plans and preferences, must be stored securely and maintained with accuracy to prevent data corruption or loss.

5. **Recoverability:** In the event of a system failure, the travel planner should have mechanisms in place to quickly recover and restore normal operation with minimal data loss.
6. **Performance:** The system should consistently meet performance requirements, responding to user queries and generating travel plans within acceptable time frames.
7. **User Authentication and Authorization:** Robust authentication and authorization mechanisms must be in place to ensure that only authorized users can access and modify their travel plans, safeguarding against unauthorized access.
8. **Usability:** The travel planner should be user-friendly and intuitive, minimizing the likelihood of user errors that could impact the system's dependability.
9. **Compatibility:** The system should be compatible with various devices and browsers, ensuring a seamless user experience across different platforms.
10. **Documentation:** Comprehensive documentation should be provided to support system administrators and users, facilitating proper system operation and troubleshooting.

2.2.5 Maintainability Requirements:

The maintainability requirements for the travel planner project include:

1. **Modularity:** The system should be designed with modular components, allowing for easy updates or replacements of individual modules without affecting the entire system.
2. **Documentation:** Comprehensive and up-to-date documentation should be provided for developers. system administrators, and maintainers to facilitate understanding and future modifications.
3. **Code Maintainability:** The source code must adhere to coding standards, be well-commented, and follow best practices to ensure that future development and maintenance can be performed efficiently.
4. **Version Control:** Implement version control systems to track changes in the codebase, enabling easy rollback to previous versions if issues arise and providing a history of modifications.
5. **Configuration Management:** Maintain a clear configuration management process to handle changes in system configurations, ensuring that updates can be applied smoothly.
6. **Logging and Monitoring:** Implement robust logging mechanisms and monitoring tools to track system behaviour, identify issues, and facilitate debugging during maintenance activities.

7. **Error Handling:** The system should have effective error handling mechanisms in place, providing detailed error messages to aid developers in identifying and resolving issues promptly.
8. **Compatibility with Tools and Libraries:** Ensure compatibility with common tools for development, libraries, and frameworks to facilitate updates and integration of new features.
9. **Testing Environment:** Maintain a dedicated testing environment that mirrors the production environment to test updates and changes before deploying them to the live system.
10. **Training for Maintainers:** Provide training for maintenance personnel to ensure they are equipped with the necessary skills and knowledge to handle ongoing system maintenance and updates.
11. **Backup and Recovery Procedures:** Establish regular backup procedures and define recovery processes to protect against data loss and facilitate quick system recovery in the event of failures.

2.2.6 Security Requirements:

Security requirements for a Travel Planner project involve measures to protect sensitive information, ensure data integrity, and safeguard user privacy. Key considerations are:

1. Authentication and Authorization:

a) Implement secure user authentication mechanisms to verify the identity of users.

b) Define roles and permissions to control access to different functionalities based on user roles.

2. Data Encryption:

a) Utilize encryption protocols (such as https) to secure data transmitted between the client and server.

b) Implement encryption for sensitive data stored in databases to prevent unauthorized access.

3. Secure Storage:

a) Ensure that user credentials, personal information, and any sensitive data are securely stored.

b) Employ secure coding practices to protect against common vulnerabilities like SQL injection and cross-site scripting.

4. Session Management:

a) Implement secure session handling to prevent session hijacking or unauthorized access to user accounts.

b) Enforce session timeouts and provide mechanisms for users to log out securely.

5. **API Security:**

a) If the project involves external APIs, validate and sanitize inputs to prevent malicious data injection.

b) Use API keys or tokens to authenticate and authorize access to external services.

6. **Logging and Monitoring:**

a) Implement comprehensive logging to track system activities and detect any suspicious behaviour.

b) Set up monitoring systems to alert administrators about potential security incidents.

7. **User Privacy:**

a) Clearly communicate the project's privacy policy to users.

b) Minimize the collection of personally identifiable information and obtain user consent where necessary.

2.2.7 Look and Feel Requirements:

In the realm of software development, the look and feel requirement is pivotal, especially for projects like a travel planner where user experience is paramount. This requirement essentially outlines the visual and interactive elements that define the user interface (UI) and user experience (UX) of the application. Let's delve into some key aspects you might want to consider for your travel planner project:

1. **Visual Design:**

   The visual design sets the tone for the entire application. Consider factors such as color schemes, typography, iconography, and imagery. For a travel planner, you might opt for vibrant colours evoking a sense of adventure or calming hues reminiscent of tranquil destinations.

2. **Interactive elements:**

   Engaging interactive elements can enhance user engagement and make the planning process more enjoyable. Incorporate features such as interactive maps, drag-and-drop functionality for itinerary customization, and real-time feedback to provide users with a dynamic and interactive experience.

3. **Performance:**

   Optimize the performance of the application to ensure swift loading times and smooth interactions. Minimize unnecessary animations, compress images, and leverage caching techniques to enhance performance.

2.3 Validation

Validation in the context of a Travel Planner Project involves ensuring that the system meets its specified requirements, functions correctly, and delivers the intended user experience. The validation process aims to verify that the travel planner operates accurately, efficiently, and securely. Here's a brief overview of the key aspects of validation for a Travel Planner Project:

2.3.1 Functional Validation:

- **Itinerary Accuracy:** Ensure that the generated travel itineraries are accurate, considering factors like travel time, destination category etc.
- **User Authentication:** Verify that user authentication mechanisms work securely to protect user accounts and sensitive information.

2.3.2 Usability Validation:

- **User Interface (UI) Testing:** Assess the user interface for ease of use, clarity, and responsiveness across different devices and screen sizes.

2.3.3 Performance validation:

- **Response Time:** Evaluate the system's response time to user inputs, ensuring that it performs efficiently and provides a seamless experience.
- **Scalability:** Test the system's ability to handle varying loads, especially during peak usage periods, to ensure it can scale to accommodate increasing numbers of users.

2.3.4 Security Validation:

- **Data Security:** Confirm that user data is stored and transmitted securely, protecting it from unauthorized access and ensuring compliance with data protection regulations.

2.3.5 Compatibility Validation:

- **Cross-Browser Testing:** Ensure that the travel planner functions correctly across different web browsers to accommodate users with varying preferences.

2.4 Expected Challenges

While developing a travel planner project, various challenges and hurdles can be anticipated. Addressing these challenges effectively is crucial for the success and usability of the application. Here are some expected hurdles in a travel planner project:

1. **Data Integration and APIs:**

   Connecting to and integrating data from various sources such as accommodation availability, and local events can be complex. Different APIs may have varying data formats and requirements, leading to integration challenges.

2. **Dynamic Data Updates:**

   Managing real-time data updates, especially for factors like cancellations, or changes in local events, requires constant synchronization. Ensuring that users receive accurate and timely information poses a technical challenge.

3. **User Privacy and Security:**

   Handling sensitive user information, such as travel itineraries and personal preferences, requires robust security measures. Protecting user data from potential breaches or unauthorized access is a critical concern.

4. **Offline Functionality:**

   Providing a reliable offline mode for users who may not have consistent internet access requires careful planning. Ensuring that essential features, such as offline maps and itinerary access, function seamlessly without an internet connection can be challenging.

5. **Continuous Testing and Bug Resolution:**

   Testing the application thoroughly to identify and resolve bugs is an ongoing process for ensuring a smooth and bug-free experience to users requires.

2.5 SDLC Model to be Used

Choosing the right Software Development Life Cycle (SDLC) model is crucial for the success of a travel planner project. The SDLC model dictates the stages and activities involved in the software development process.

For a travel planner project, where requirements may evolve, and user engagement and feedback are crucial, an **Agile SDLC model** is often recommended. The Agile model is characterized by its iterative and incremental approach, allowing for flexibility and adaptability throughout the development process. Here's why Agile can be suitable for a travel planner project:

- **Iterative Development:**
  Agile divides the project into small, manageable iterations known as sprints. Each sprint typically lasts for a few weeks, during which a set of features is developed, tested, and delivered. This iterative approach allows for continuous improvement and adaptation based on changing requirements or user feedback.
- **Customer Collaboration:**
  Agile places a strong emphasis on customer collaboration throughout the development process. This is particularly important for a travel planner project, where user preferences, feedback, and changing travel plans can significantly impact the software requirements.
- **Quick Adaptation to Changes:**
  The Agile model is designed to accommodate changes in requirements at any stage of the development process. In the context of a travel planner, where new destinations, activities, or user preferences may emerge, the ability to adapt quickly is essential.
- **Continuous User Feedback:**
  Regular feedback from users is integral to the Agile model. This ongoing interaction ensures that the travel planner aligns with user expectations and provides a user-friendly experience. Users can contribute to the refinement of features and suggest improvements based on their real-world travel experiences.

SYSTEM DESIGN

3.1 Design Approach

Designing a travel planner project involves making decisions about the architecture and structure of the software. Two common design approaches are function-oriented design and object-oriented design. Let's explore both approaches in the context of a travel planner project.

3.1.1 Object Oriented Design:

In an object-oriented design, the system is modelled as a collection of objects that represent real-world entities. These objects encapsulate data and behaviour, promoting modularity and reusability. Here's how you might apply this approach to a travel planner project:

- **Class Design:**

  i. User Class: Represents a user with attributes like username, password, and preferences.

  ii. Location Class: Represents a travel destination with details such as name, attractions, and travel options.

  iii. Itinerary Class: Manages the user's travel plans, considering time, budget, and preferences.

  iv. Booking Class: Represents a booking for flights, accommodations, etc.

  v. Notification Class: Handles notifications for users.

  - **Advantages:**

    i. Encourages code reuse and modularity.

    ii. Mimics real-world scenarios, making it easier to understand.

    iii. Scalable and adaptable to changes.

3.2 System Design Using Various Structured Analysis

System design is the process of defining the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

System designing in terms of software engineering has its own value and importance in the system development process as a whole. To mention it may though seem as simple as anything or simply the design of systems, but in a broader sense it implies a systematic and rigorous approach to design such a system which fulfils all the practical aspects including flexibility, efficiency and security.

The importance of this phase may be understood by reason of the fact that it involves identifying data sources, the nature and type of data that is available. For example, in order to design a salary system, there is a need for using inputs, such as, attendance, leave details, additions or deductions etc. This facilitates understanding what kind of data is available and by whom it is supplied to the system so that the system may be designed considering all there levant factors. In addition, system designing leads to ensure that the system is created in such a way that it fulfils the need of the users and keep them at ease being user-oriented.

System design for a travel planner project involves specifying how the system will fulfil the requirements outlined in the earlier phases of the software development life cycle. Structured analysis and design tools are used to model, visualize, and document the various components and their interactions. Here are some commonly used tools and techniques in the context of a travel planner project:

i. Data Flow Diagram
ii. Data Dictionary
iii. Flowchart
iv. Entity Relationship Diagram

3.2.1 Data Flow Diagrams(DFD) :

A data flow diagram is a graphical representation or technique depicting information flow and transform that are applied as data moved from input to output. The DFD are partitioned into levels that represent increasing information flow and functional details. The processes, data store. data flow, etc are described in Data Dictionary.

1. Data flow: Data moves in a specific direction from an origin to destination.

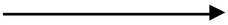


Figure 3.1 Data flow

2. Process: Procedures or devices that use or transform data.

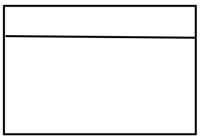


Figure 3.2: Process

3. External Entity: This defines a source (originator) or destination of system data.

Figure 3.3 External Entity

4. Data store: This indicates where data is stored in the system.

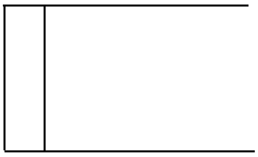


Figure 3.4 Data store

3.2.1.1 Zero Level DFD:

The Zero Level Data Flow Diagram (DFD) for the Travel Planner Project provides a high-level overview of the system's functionality without delving into specific processes. At this level, the diagram typically showcases the main external entities interacting with the system, such as users and external databases. The Zero Level DFD illustrates the basic input and output interactions between these entities and the central system, offering a simplified yet comprehensive view of the overall flow of information within the travel planning system.

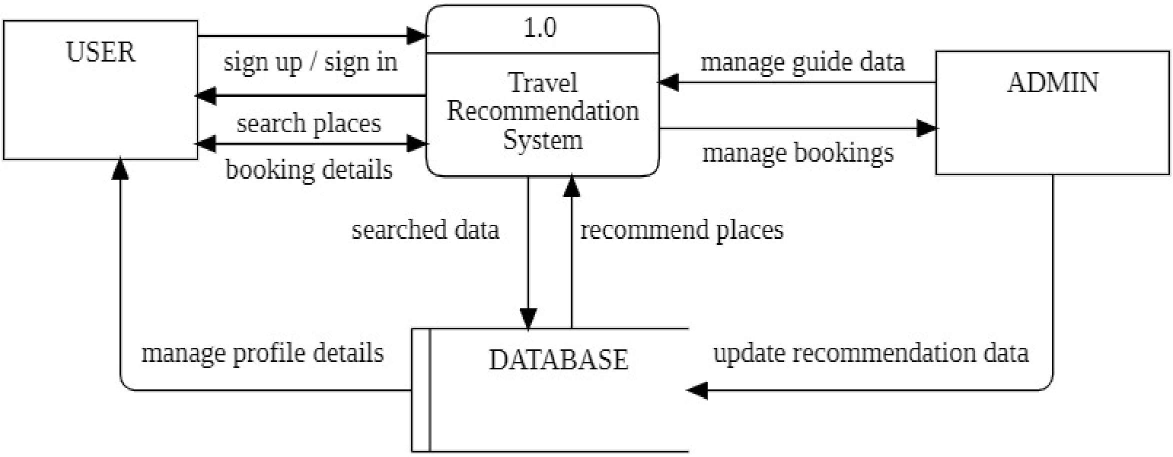


Figure 3.5: Zero level DFD

3.2.1.2 First Level DFD:

A Level 1 Data Flow Diagram (DFD) for a Travel Planner project provides a high-level overview of the system's functionality, illustrating the main processes and data flows within the system. In a travel planner system, users typically input their travel preferences, and the system helps them plan their trips by providing information on destinations, accommodations, transportation, and other relevant details. By representing the Travel Planner project at a high level through a Level 1 DFD, stakeholders can gain a clear understanding of the major processes, data flows, and external interactions within the system. This diagram serves as a foundation for more detailed DFDs and system specifications as the project progresses.

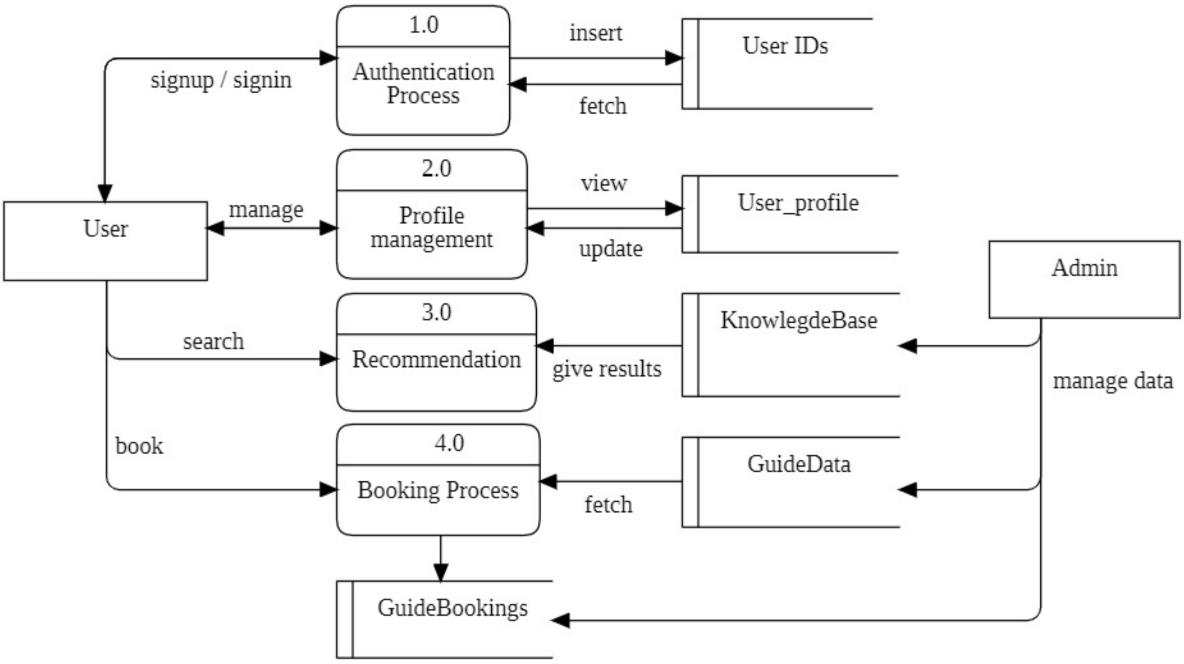


Figure 3.6: First Level DFD

3.2.1.3 Second Level DFD:

A Level 2 Data Flow Diagram (DFD) for a Travel Planner project provides a more detailed view of the system than the Level 1 DFD. It breaks down the major processes identified in the Level 1 DFD into sub-processes and illustrates the flow of data between them. In the context of a Travel Planner project, the Level 2 DFD can capture more specific functionalities and interactions within the system. his Level 2 DFD provides a more granular view of the Travel Planner system, outlining specific tasks and data interactions within each major process. It serves as a valuable tool for both developers and stakeholders to understand the detailed functionality of the system and can be used as a basis for further development and refinement.

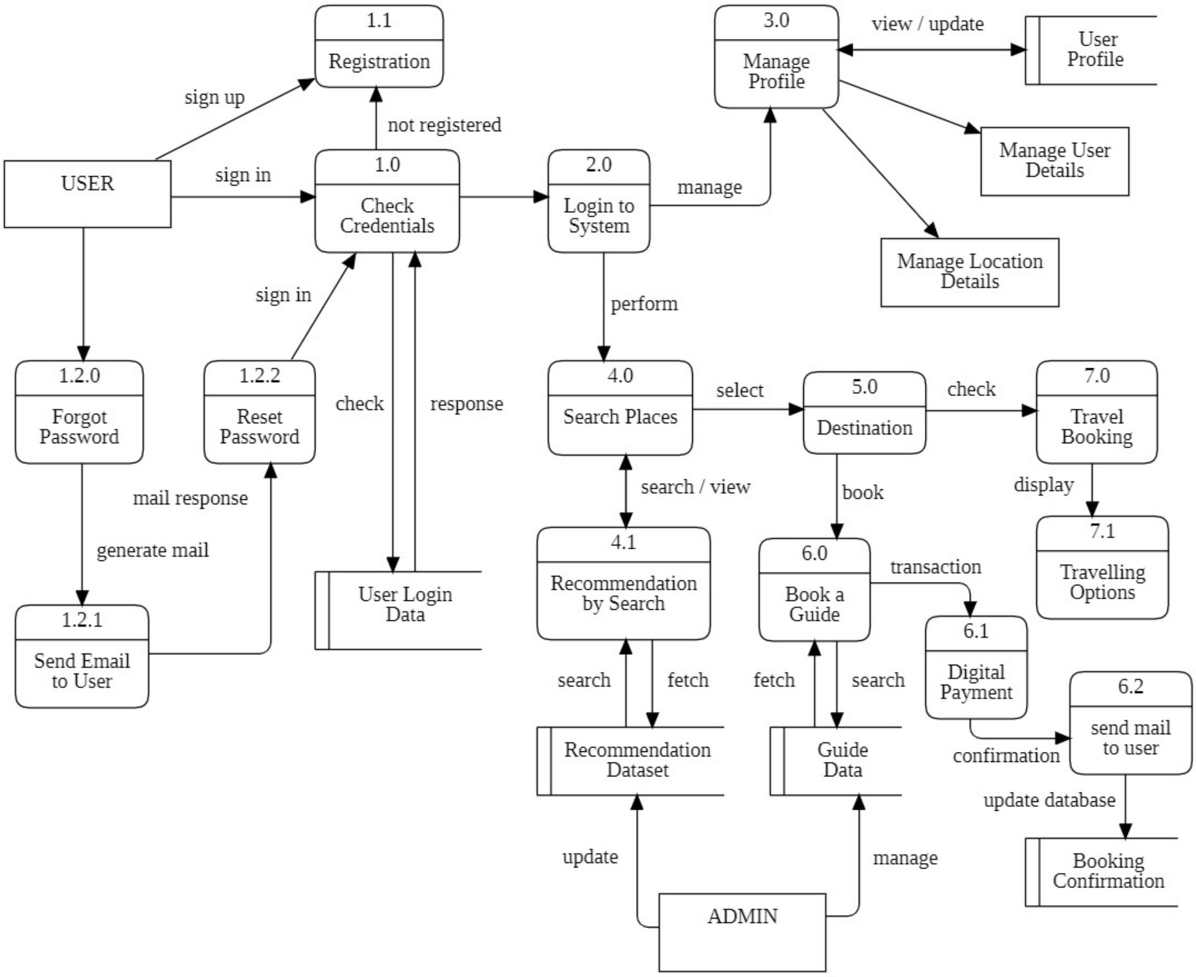


Figure 3.7: Second Level DFD

3.2.2 Data Dictionary:

In the Travel Planner Project, a data dictionary serves as a comprehensive and structured repository of metadata that defines and describes the various data elements used within the system. It provides a detailed catalogue of data entities, attributes, data types, constraints, and relationships, offering a standardized reference for developers, database administrators, and other stakeholders involved in the project. For instance, the data dictionary may include entries for entities such as "User," "Itinerary," "Destination," and "Booking." Each entry specifies the attributes associated with the entity, such as "User ID," "Destination Name," "Booking Date," etc., along with their respective data types and constraints. Furthermore, the data dictionary outlines relationships between entities, ensuring a clear understanding of the data model. This centralized documentation not only aids in system development and maintenance but also promotes consistency, accuracy, and collaboration among team members by providing a shared understanding of the data structure and its semantics within the Travel Planner Project.

Table 3.1: User IDs

FIELDS	DATATYPE	NULL	KEY	DEFAULT
u_id	varchar(20)	NO	PRIMARY	current_timestamp()
username	text	NO		NULL
email	text	NO	UNIQUE	NULL
password	text	NO		NULL

Table 3.2: Pictures

FIELDS	DATATYPE	NULL	KEY	DEFAULT
s_id	varchar(10)	NO	PRIMARY	NULL
district	varchar(30)	NO		NULL
imgurl_1	text	NO		NULL
imgurl_2	text	NO		NULL
imgurl_3	text	NO		NULL
imgurl_4	text	NO		NULL
imgurl_5	text	NO		NULL

Table 3.3: User Profile

FIELDS	DATATYPE	NULL	KEY	DEFAULT
U_ID	varchar(20)	NO	PRIMARY	current_timestamp()
username	varchar(30)	NO		NULL
fullname	varchar(50)	NO		NULL
mobile_no	bigint(12)	NO	UNIQUE	NULL
email	text	NO	UNIQUE	NULL
gender	text	NO		NULL
age	int(3)	NO		NULL
country	varchar(30)	NO		NULL
state	varchar(30)	NO		NULL
city	varchar(50)	NO		NULL
district	varchar(30)	NO		NULL
preference_1	varchar(20)	YES		NULL
preference_2	varchar(20)	YES		NULL
preference_3	varchar(20)	YES		NULL

Table 3.4: Guide Data

FIELDS	DATATYPE	NULL	KEY	DEFAULT
guide_id	bigint(10)	NO	PRIMARY	NULL
name	varchar(50)	NO		NULL
age	int(3)	NO		NULL
gender	varchar(15)	NO		NULL
mobile_no	bigint(12)	NO	UNIQUE	NULL
email_id	text	NO	UNIQUE	NULL
language_1	varchar(15)	NO		NULL
language_2	varchar(15)	NO		NULL
language_3	varchar(15)	NO		NULL
rating	int(2)	NO		NULL
availability	varchar(10)	NO		NULL

Table 3.5: User Bookings

FIELDS	DATATYPE	NULL	KEY	DEFAULT
user_id	varchar(20)	NO		NULL
guide_id	bigint(10)	NO		NULL
destination	varchar(30)	NO		NULL
startdate	date	NO		NULL
enddate	date	NO		NULL
transaction_id	varchar(255)	NO		NULL
razorpay_payment_id	varchar(255)	NO		NULL
price	int(50)	NO		NULL
status	varchar(50)	NO		NULL

3.2.3 Flowchart:

A flowchart is a visual representation of the sequence of steps and decision points in a process. It is commonly used to outline the flow of a project, breaking it down into individual tasks and showing how they relate to each other. Here's a general description of the elements you might include in a project flowchart:

1. Start/End Symbol:

The flowchart typically begins with an oval or rounded rectangle indicating the start of the project and ends with a similar symbol representing the completion or closure of the project.

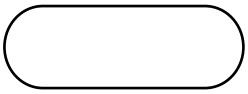


Figure 3.8: Terminator

2. Process Symbol:

Rectangles are used to represent processes or activities in the project. Each rectangle contains a brief description of the task or activity.

Figure 3.9: Flow Process

3. Decision Symbol:

Diamonds are used to represent decision points in the project where a choice must be made. The flow of the chart diverges based on the outcome of the decision.

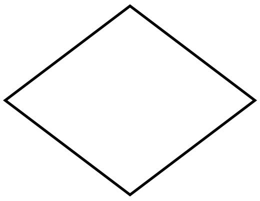


Figure 3.10: Decision Symbol

4. Arrow/ Flow Lines:

Arrows connect the symbols, indicating the flow or sequence of tasks. The direction of the arrow shows the order in which tasks are completed.

Figure 3.11: Flow Line

5. **Input/ Output Data:**

Parallelograms are used to represent inputs or outputs in the project. These can include data, materials, or information entering or leaving the process.

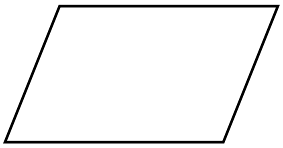


Figure 3.12: Input/ Output Data

6. **Predefined process Symbol:**

This symbol represents a complex process or operation already explained elsewhere in the flowchart. It's a rectangle with a line at each end of the shape. It helps depict sub-processes, or the smaller processes within a larger process.

Figure 3.13: Predefined Process

7. **Database:**

This cylinder-shaped flowchart symbol represents stored data.

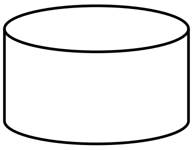


Figure 3.14 Database

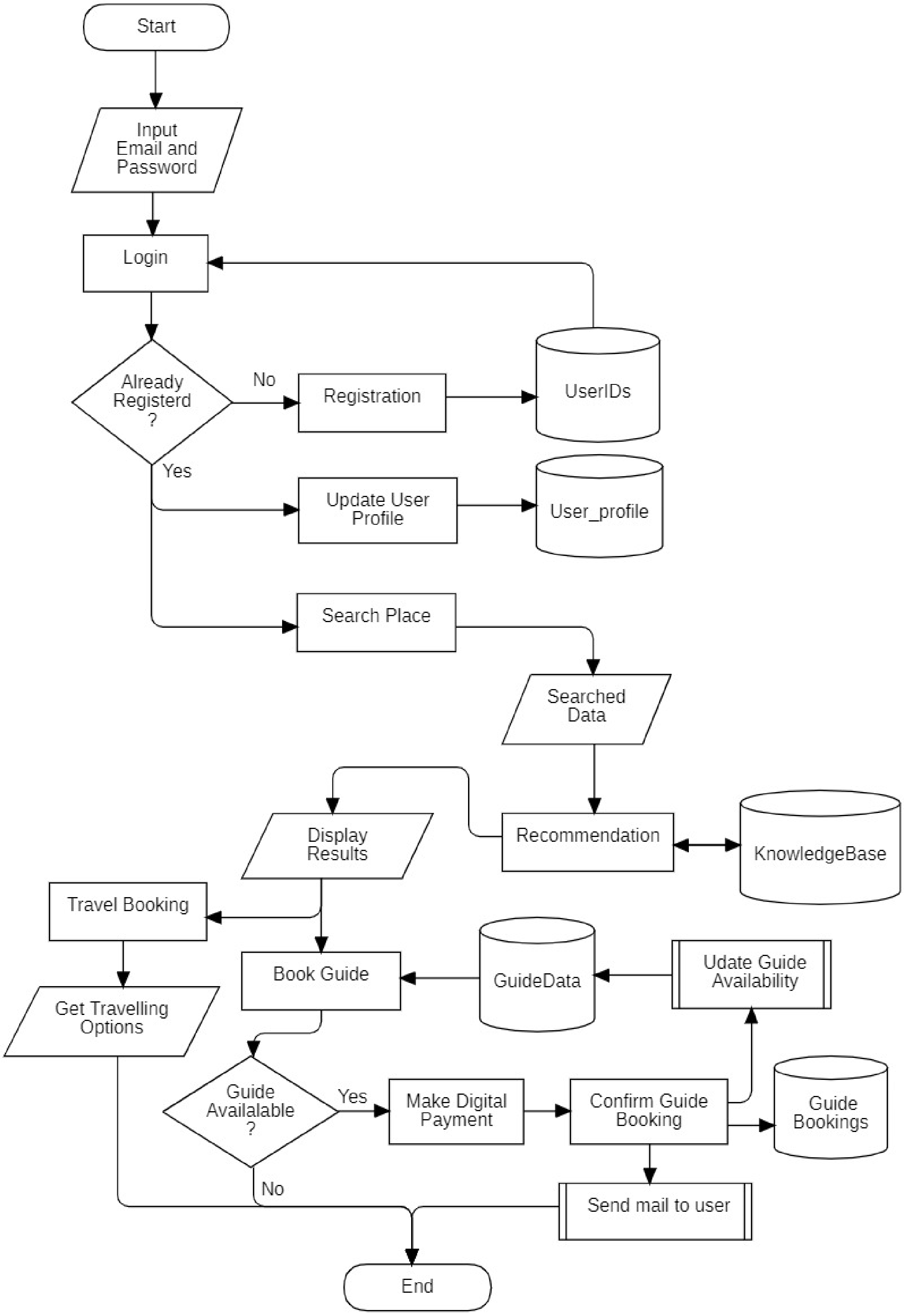


Figure 3.15: Flowchart

3.3 ER Diagram

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships. ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationship.

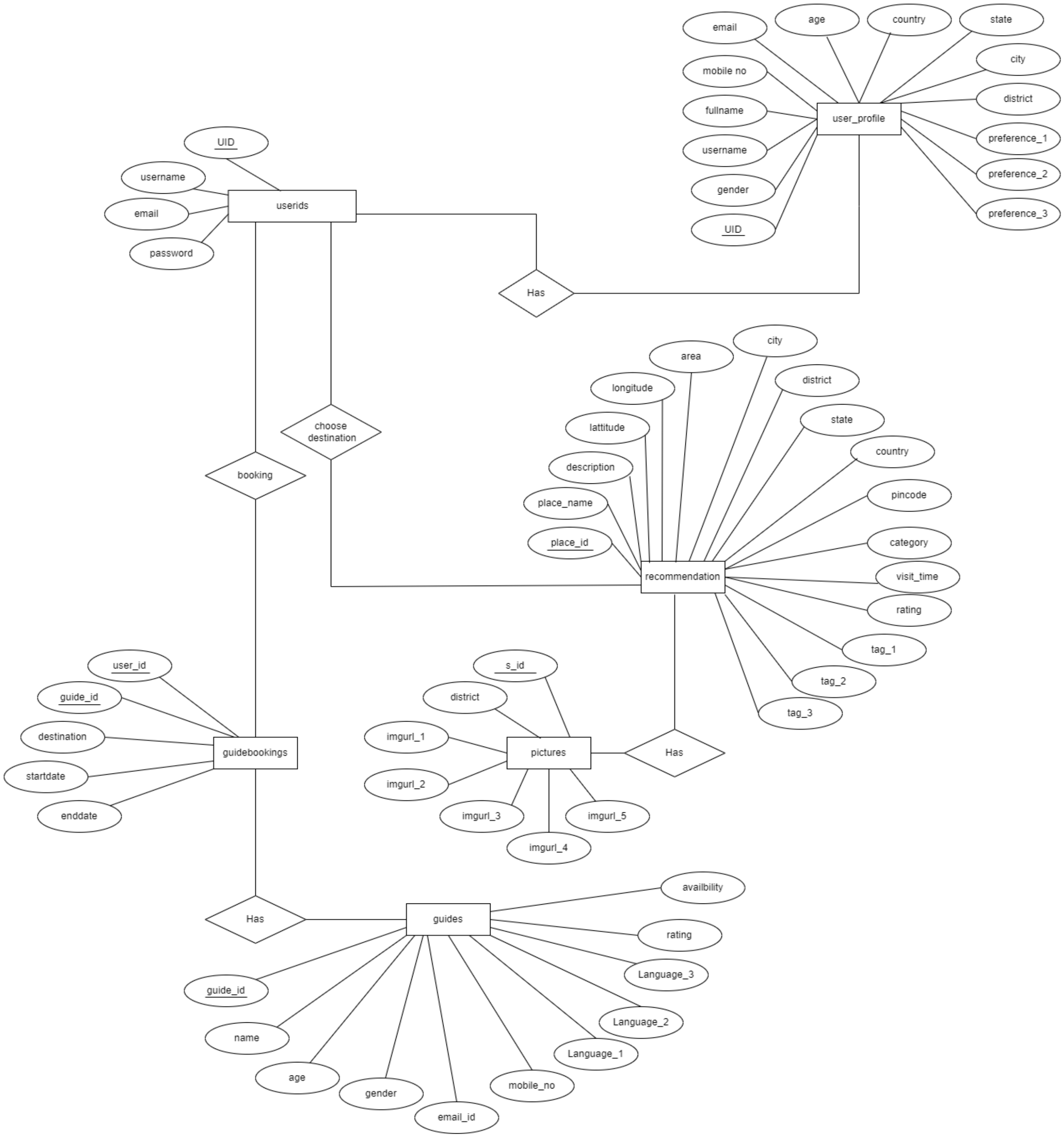


Figure 3.16: ER Diagram

3.4 Normalization

Normalization is the process of organizing data in a relational database to reduce redundancy and improve data integrity. In the context of a travel planner project, the database should be normalized to eliminate data anomalies and support flexible and efficient querying. The normalization process typically involves breaking down large tables into smaller, related tables and establishing relationships between them.

3.4.1 First Normal Form (1NF):

Ensure that each table's data is atomic, meaning it cannot be divided further. For instance, a 'Destinations' table should not store a list of cities in a single column but rather have separate columns for city, state, and country.

3.4.2 Second Normal Form (2NF):

Builds on 1NF by ensuring that each non-key column is fully functionally dependent on the primary key. In the context of a travel planner, if there is a 'Trips' table, information about the trip details (e.g., date, duration, purpose) should be separated from information about the traveller.

3.4.3 Third Normal Form (3NF):

Extends normalization by eliminating transitive dependencies. For example, if there is a 'Travelers' table, information about the traveller's address should be stored in a separate 'Addresses' table, linked by a foreign key.

3.5 Database Manipulation

Manipulating the database involves the insertion, updating, and deletion of records. In a travel planner project, this functionality is vital for users to create, modify, and remove trip details.

1. **Inserting Records:**

   Users should be able to add new profile details, book guides and get information about booking of guide. Proper validation of user inputs is essential to maintain data accuracy.

2. **Selecting Records:**

   The select statement allows users to pull a selection from the database to work with. we tell the computer what to SELECT and FROM which table. The ability to select records from the database enables the application to display destinations, activities, and other details that match the user's preferences.

3. **Updating Records:**

Travel plans are subject to change. Users need the ability to modify booking details, update accommodation choices, or change the duration of their stay. The database should support efficient updates while preserving data consistency.

3.6 Database Connection Controls and Strings Methodology

3.6.1 Create a New Connection :

To access data in the MySQL database, we need to connect to the server. Once the connection has been made, we can start interacting with the database.

```
$servername = 'localhost';
$username = 'username';
$password = 'password';
$conn = new mysqli($servername, $username, $password);
$conn = new mysqli($servername, $username, $password, $dbname);
```

3.6.2 Creating Table:

A database table consists of columns and rows. The CREATE TABLE statement is used to create a table in MySQL.

```
$sql = "create table TableName (field_1 datatype key_constraints, field2 datatype null(yes | no), field_3 datatype default, field_4 timestamp)”;
if ($conn->query($sql) === TRUE) { echo "Table created successfully";}
else {echo "Error creating table: " . $conn->error;}
```

3.6.3 Inserting data into table:

User can start adding or inserting data in the table. The INSERT INTO statement is used to add new records to a MySQL table.

```
$sql = "insert into TableName (field_1,field_2,field_3,field_4)values ('$value1','$value2', '$value3', '$value4',)";
if ($conn->query($sql) === TRUE) { echo "New record created successfully";}
else {echo "Error: " . $sql . "<br>" . $conn->error;} `
```

3.6.4 Getting Data from Table:

The SELECT statement is used to select data from one or more tables. Then, run the query and store the resulting data into a variable called $result. The function num_rows() checks if there are more than zero rows returned. If there are more than zero rows returned, the function fetch_assoc() puts all the results into an associative array that you can loop through. The while() loop loops through the result set and outputs the data.

```
$sql = "select * from TableName";
$result = $conn->query($sql);if ($result->num_rows > 0) {
while($row = $result->fetch_assoc()){
echo "id: " . $row["id"]. " - Name: " . $row["name"]. " <br>";}}else{echo "0 results";}
```

3.6.5 Updating Data in Table:

The UPDATE statement is used to update existing records in a table. The WHERE clause specifies which record or records that should be updated.

```
$sql = "update TableName set field_name='$fieldvalue' where primaryfield =
’$primaryvalue’";
if ($conn->query($sql) === true){ echo "record updated successfully";}
else { echo "error updating record: " . $conn->error;}
```

3.6.6 Close the Connection:

The connection is closed automatically when the script ends. To close the connection:

```
$conn->close();
```

IMPLEMENTATION AND TESTING

4.1 Introduction to Languages

It refers to the use of specific programming languages for developing the software that powers the travel planning system. Programming languages serve as the foundation for creating the functionality, logic, and interactions within the system. The choice of programming languages can significantly impact the project's development, with considerations such as performance, scalability, and ease of maintenance. For a travel planner, a mix of languages may be employed, including front-end languages like HTML, CSS, and JavaScript for user interface development, and back-end languages such as Python and PHP for server-side logic and data processing. Additionally, the integration of database languages like SQL might be necessary for efficient data management. The introduction to programming languages in the Travel Planner Project underscores the crucial role that technology plays in enabling the creation of a robust and user-friendly travel planning platform.

Languages used for the making of this “Travel Planner” project are categorised into two types i.e. scripting languages that are majorly used for the designing section in the project and the programming languages for the development of the backend part of the project.

4.1.1 Scripting Languages:

A scripting language is a programming language that is interpreted. It is translated into machine code when the code is run, rather than beforehand. Scripting languages are often used for short scripts over full computer programs. JavaScript, Python, and Ruby are all examples of scripting languages. All scripting languages are programming languages, but not all programming languages are scripting languages.

1. HTML

HTML is the backbone of any web page, providing the structure and semantic markup for content. In the context of a travel planner project, HTML is used to define the document structure, input fields, links and navigation, multimedia integration and more. HTML organizes the content into a structured format. Travel planners often involve user input for destinations, dates, and preferences. HTML forms and input fields allow users to provide information, facilitating the interaction between the user and the travel planner application. It supports the integration of multimedia elements such as images and videos. Travel planners can use these features to showcase destination images, videos, and other relevant visual content.

2. CSS

CSS complements HTML by styling and presenting the content defined in HTML. In a travel planner project, CSS is used for layout and positioning, typography and fonts, colour schemes, responsive design, animations and transitions. CSS styles define the layout of the travel planner, including the placement of elements, the structure of the page, and the overall visual design. With the prevalence of various devices, including desktops, tablets, and smartphones, CSS is crucial for implementing responsive design. CSS animations and transitions can be applied to elements within the travel planner, adding a level of interactivity and engagement.

3. JavaScript

JavaScript is the primary scripting language for web development. It is widely used to create dynamic and interactive user interfaces on the client side. With the advent of frameworks, JavaScript facilitates the development of single-page applications (SPAs) that deliver a seamless user experience in travel planners. Additionally, Node.js allows JavaScript to be used on the server side, enabling developers to use the same language throughout the entire stack.

4. PHP

PHP (Hypertext Preprocessor) is a server-side scripting language designed for web development. Although its popularity has waned in comparison to JavaScript and Python, PHP remains a viable option for implementing server-side logic in travel planner projects. It is often integrated with web servers like Apache or Nginx to generate dynamic web pages and interact with databases.

4.1.2 Programming Languages:

A programming language is a formal language that specifies a set of instructions for a computer to perform specific tasks. It's used to write software programs and applications, and to control and manipulate computer systems. There are many different programming languages, each with its own syntax, structure, and set of commands.

1. Python

Python is a versatile and powerful programming language that can be effectively utilized in various aspects of a Travel Planner Project. Its readability, extensive libraries, and robust frameworks make it a popular choice for backend development and various other functionalities in web applications, including travel planners.

2. Recommendation System

A recommendation system employs algorithms to analyse user preferences and behaviour, offering personalized suggestions tailored to individual needs. In the realm of travel planning, these systems sift through vast amounts of data encompassing destinations, accommodations, activities, and reviews to curate bespoke itineraries for travellers. By harnessing machine learning and data mining techniques, recommendation systems can accurately predict user preferences, thus streamlining the decision-making process and maximizing satisfaction. Whether it's suggesting hidden gems off the beaten path, highlighting popular attractions, or recommending accommodations that align with specific criteria, these systems empower travellers to discover new destinations and experiences with ease. Ultimately, recommendation systems serve as invaluable companions, transforming travel planning into a seamless and enriching journey.

4.1.3 API

APIs (Application Programming Interfaces) are vital components for modern travel planners. They allow your application to communicate and interact with other services, databases, and platforms, enabling you to access a wealth of data and functionalities.

1. **IPfy:**

IPfy is a vital component within your travel planner project, offering essential functionality. With IPfy, your application can accurately determine users' locations based on their IP addresses. This allows for seamless integration of location-based services, such as suggesting nearby attractions, recommending restaurants, or providing localized weather updates. By leveraging IPfy's robust IP geolocation capabilities, your travel planner becomes more personalized and efficient, enhancing the user experience. Its precise location tracking ensures that your users receive relevant information tailored to their whereabouts, making their journey smoother and more enjoyable. In essence, IPfy enriches your travel planner with invaluable location intelligence.

2. **RazorPay API:**

RazorPay API offers seamless payment integration for your travel planner project, streamlining transactions with efficiency and security. With its robust features, it facilitates online payments, enabling users to book flights, hotels, and activities hassle-free. It’s simple yet powerful interface allows for easy customization, ensuring a seamless checkout experience for customers. Through RazorPay, you can accept various payment methods, including credit/debit cards, net banking, and popular digital wallets, catering to diverse user preferences. With its reliable infrastructure and extensive documentation, RazorPay API empowers your travel planner project to provide a smooth and trustworthy payment process, enhancing user satisfaction and business success.

4.1.4 IDE to be Used

1. VS Code:

VS Code offers a highly responsive and intelligent code editor with built-in syntax highlighting. This feature helps developers write clean and error-free code by visually distinguishing different programming constructs. This is especially crucial when working with HTML, CSS, JavaScript, MySQL, PHP and Python, the likely technologies involved in a travel planner project. VS Code provides robust debugging support for multiple programming languages, including Python and JavaScript. This allows developers to identify and fix issues efficiently, ensuring the travel planner functions as intended.

2. Jupyter Notebook:

Jupyter Notebook is used by developers for prototyping and experimenting with algorithms or data processing techniques before integrating them into the main codebase. This iterative and interactive approach can accelerate the development of features such as personalized recommendations or predictive analytics within the travel planner.

4.1.5 Technologies to be Used for Implementation

Web Technology refers to the various tools and techniques that are utilized in the process of communication between different types of devices over the Internet. A web browser is used to access web pages. Web browsers can be defined as programs that display text, data, pictures, animation, and video on the Internet. Hyperlinked resources on the World Wide Web can be accessed using software interfaces provided by Web browsers.

Web Technology can be Classified into the Following Sections:

- **Web Browser:** The web browser is an application software to explore www (World Wide Web). It provides an interface between the server and the client and requests to the server for web documents and services.
- **Web Server:** Web server is a program which processes the network requests of the users and serves them with files that create web pages. This exchange takes place using Hypertext Transfer Protocol (HTTP).
- **Web Pages:** A webpage is a digital document that is linked to the World Wide Web and viewable by anyone connected to the internet has a web browser.

4.1.6 Tools to be Used

Tools like Draw.io and StarUML can be valuable in the development process of a Travel Planner project for creating visual representations of system architecture, design, and workflows. Xampp Control Panel with its integrated Apache and MySQL Servers which focuses on creating dynamic and data-driven PHP and MySQL web applications.

1. Draw.io:

- **Purpose:** Draw.io is a versatile diagramming tool that allows users to create a wide range of diagrams, including flowcharts, UML diagrams, network diagrams, and more.
- **Usage in Travel Planner Project:** Draw.io can be used to create visual representations of the user interface, user flows, and high-level system architecture. For instance, it can be employed to design and document the flow of travel booking processes, user interaction with the application, or the overall structure of the Travel Planner system.

2. StarUML:

- **Purpose:** StarUML is a UML modelling tool designed for software development. It supports various UML diagram types, including class diagrams, data flow diagrams, flowchart diagrams and more.
- **Usage in Travel Planner Project:** Star UML can be utilized to create detailed UML diagrams that provide a comprehensive view of the software architecture. Class diagrams can illustrate the relationships between different classes in the system, while use case diagrams can help define and understand user interactions with the Travel Planner application. Entity relationship diagrams can be beneficial for visualizing the flow of activities during the travel planning process.

3. Xampp Control Panel:

- **Purpose:** The XAMPP Control Panel is a graphical user interface (GUI) that acts as the command centre for managing the essential components of XAMPP, a popular open-source web development package. It streamlines the process of running Apache, the web server, and MySQL, the database management system, which are crucial for developing web projects using PHP and MySQL.
- **Usage in Travel Planner Project:** With a click of a button, it can effortlessly start, stop, or restart Apache and MySQL. This allows developers to quickly launch the development environment and shut it down when not in use. The Control Panel provides a clear view of the status of both Apache and MySQL, indicating whether they are running or stopped. This helps developer to stay informed about the development environment's state.

4.2 Coding Standards of language used

In the context of a travel planner project, adhering to a consistent and well-defined coding standard is crucial for ensuring maintainability, readability, and scalability of the codebase. A coding standard serves as a set of guidelines that dictate how code should be written, formatted, and organized within the project. Here's a detailed exploration of the coding standard considerations

1. **Consistency:**

   Consistency is paramount in any coding standard. All code, regardless of the developer who wrote it, should follow the same conventions. This includes naming conventions for variables, functions, classes, and other identifiers, as well as formatting rules for indentation, spacing, and line length.

2. **Readability:**

   Code should be written in a way that is easy to read and understand for other developers who may need to review, maintain, or extend it in the future. This involves using descriptive variable and function names, writing clear and concise comments.

3. **Modularity:**

   The codebase should be modular, with separate modules or components responsible for specific functionalities or features of the travel planner application. Each module should have a clear and well-defined interface, with minimal dependencies on other modules.

4. **Error Handling:**

   Proper error handling should be implemented throughout the codebase to gracefully handle unexpected situations and prevent crashes or data corruption. This includes validating user input, handling exceptions, and logging error messages with relevant context.

5. **Performance:**

   While readability and maintainability are important, performance considerations should also be taken into account, especially in a real-time application like a travel planner.

6. **Security:**

   Security is a critical aspect of any software project, especially one that deals with sensitive user data such as travel itineraries and payment information.

4.3 Project Scheduling Using Various Tools

Project scheduling is a critical aspect of managing any endeavour, ensuring that tasks are completed efficiently and on time. For your travel planner project, utilizing tools like PERT (Program Evaluation and Review Technique) and Gantt charts can greatly enhance your ability to organize tasks, allocate resources, and track progress effectively.

4.3.1 PERT:

PERT is a project management tool used to schedule, organize, and coordinate tasks within a project. It employs a network model that represents the sequence of activities required to complete the project. Here's how we applied PERT to the travel planner project:

- **Identify Tasks:** Begin by listing all the tasks involved in your travel planner project, such as research destinations, book accommodations, plan itineraries, arrange transportation, etc.
- **Sequence Tasks:** Determine the order in which these tasks need to be completed. Some tasks may be dependent on others, while some can be executed concurrently.
- **Estimate Time:** Estimate the time required to complete each task. PERT typically uses three times estimates: optimistic (the shortest time), pessimistic (the longest time), and most likely. These estimates help in calculating the expected time for each task.
- **Construct Network Diagram:** Using the task sequence and time estimates, create a network diagram that illustrates the flow of tasks. Nodes represent tasks, and arrows represent the dependencies between them.
- **Calculate Critical Path:** Identify the critical path, which is the longest path through the network diagram and determines the shortest possible project duration. Tasks on the critical path must be completed on time to prevent delays in the project.

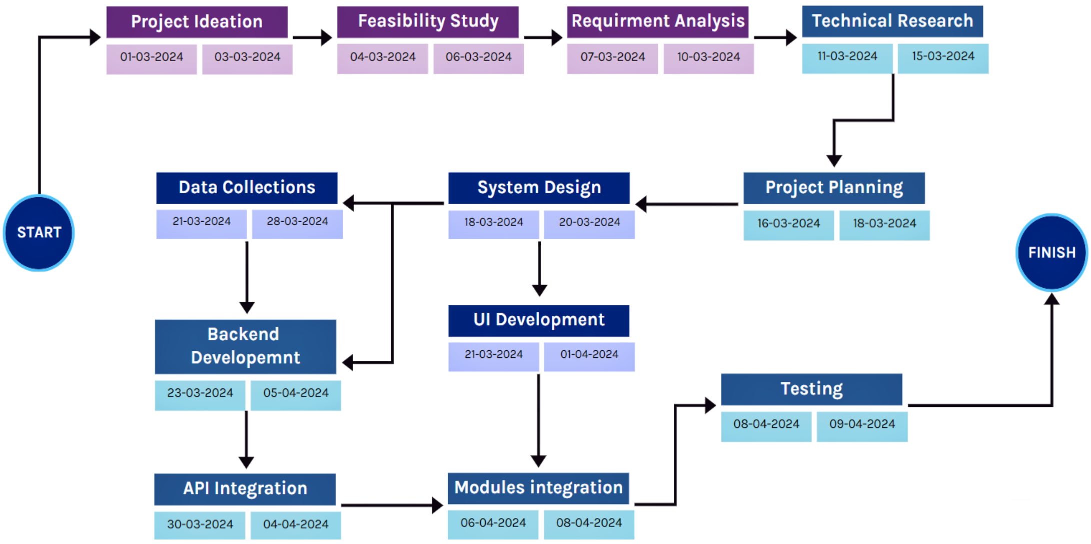
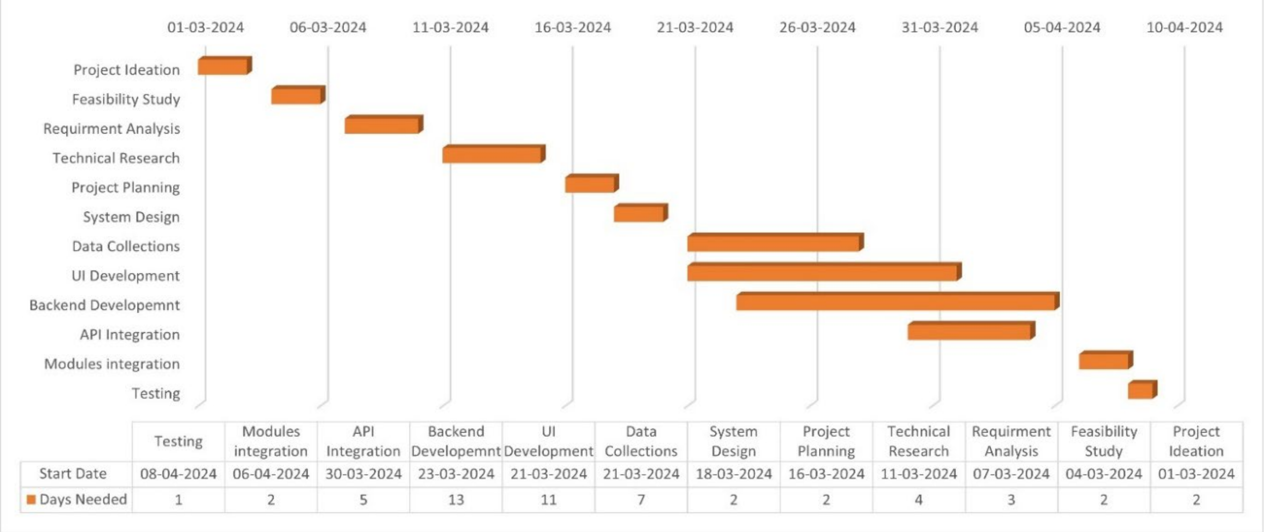


Figure 4.1: PERT Diagram

4.3.2 GANTT Chart:

A Gantt chart is a visual representation of a project schedule that shows tasks displayed against time. It provides a comprehensive overview of the project timeline, task dependencies, and resource allocation. Here's how to use a Gantt chart for your travel planner project:

- **Task Listing:** List all the tasks identified in the PERT analysis along with their start and end dates.
- **Timeline Visualization:** Create a horizontal timeline representing the duration of the project. Each task is represented as a horizontal bar on the timeline, with its length indicating its duration.
- **Task Dependencies:** Use the Gantt chart to visualize task dependencies. Tasks that must be completed before others can start are connected by arrows, indicating the sequence of execution.
- **Resource Allocation:** Allocate resources (such as team members or funds) to each task by adding resource assignments to the Gantt chart. This helps in optimizing resource utilization throughout the project.
- **Progress Tracking:** As the project progresses, update the Gantt chart to reflect actual start and end dates of tasks. This allows you to track progress, identify delays, and adjust the schedule accordingly.



	Testing	Modules integration	API Integration	Backend Developemnt	UI Development	Data Collections	System Design	Project Planning	Technical Research	Requirment Analysis	Feasibility Study	Project Ideation
Start Date	08-04-2024	06-04-2024	30-03-2024	23-03-2024	21-03-2024	21-03-2024	18-03-2024	16-03-2024	11-03-2024	07-03-2024	04-03-2024	01-03-2024
🟧 Days Needed	1	2	5	13	11	7	2	2	4	3	2	2

Figure 4.2: Gantt Chart

4.4 Testing Techniques and Test Plans

A well-designed travel planner project needs rigorous testing to ensure a smooth and positive user experience. Here are some key testing techniques to consider:

- **Functional Testing:** This verifies core functionalities like flight searches, hotel bookings, itinerary creation, and user account management. Write test cases for various scenarios, including.
- **Usability Testing:** This assesses how easy and intuitive the app is to use. Here are some ways to achieve this.
- **Non-Functional Testing:** This ensures the app performs well beyond core functionalities.
- **Performance Testing:** This tests how the app handles load. Simulate high user traffic and measure response times and stability.
- **Security Testing:** This identifies vulnerabilities related to user data, payment information, and overall system security.

4.4.1 Creating a test plan for the project

A test plan serves as a roadmap for your testing process. Here's a breakdown of the key elements:

- **Test Objectives:** Clearly define what is the aim to achieve through testing. Whether it's comprehensive functionality, smooth user experience, or robust security.
- **Test Scope:** Specify which functionalities will be tested and which will be .
- **Testing Techniques:** Outline the specific techniques to be implemented (functional testing, usability testing, etc.) and how they align with your objectives.
- **Test Cases:** Create detailed test cases for each functionality. These should include step-by-step instructions, expected results, and pass/fail criteria.
- **Entry and Exit Criteria:** Define clear conditions for starting and concluding each testing phase.
- **Resources:** Specify the tools, people, and expertise needed for testing.

RESULTS AND DISCUSSIONS

5.1 User Interface Representation

User interface representation for a travel planner project plays a crucial role in facilitating seamless interaction between the user and the application. It serves as the medium through which users input their preferences, view travel options, and make decisions regarding their journeys.

- The user interface representation is optimized for various devices, including desktops, laptops, tablets, and smartphones.
- Responsive design ensures that the interface adapts smoothly to different screen sizes and orientations, providing a consistent experience across all platforms.
- The homepage serves as the entry point for users. It features clean layout with clear navigation options. Users will be able to easily access essential functionalities such as searching for destinations, viewing their saved trips, and accessing account settings and other services.
- Visually explore potential destinations with markers or icons indicating attractions, accommodation options. Different pages showcasing specific destinations with detailed information, pictures, reviews, and booking options.
- User interface implemented with cutting-edge techniques like dynamic content delivery and lightweight UI components, ensuring a smooth experience. It is focused on blazing-fast loading times and a lag-free experience.

5.2 Brief Description of Various Modules of the System

Here's a breakdown of key components and considerations for designing the system for a travel planner project:

5.2.1 Authentication:

PHP-based mailing system designed for secure communication with real people and user's registration and login process. The system incorporates functionalities like authentication, cookies, and sessions to ensure a robust and user-friendly experience.

5.2.2 Recommendation Search:

The application searches for destinations in the database that share similar attributes with the user's search term. This allows the user to discover new locations that might be a good fit for their travel preferences, even if they haven't heard of them before. Recommendation system using content-based filtering algorithm can enhance the travel planning experience by suggesting similar locations that cater to the user's unique travel desires.

5.2.3 Destination Suggestion:

Based on the match between user preferences, current location and destination details, the system recommends a shortlist of suitable locations. Clicking on a recommendation could provide more information about the destination, like pictures, reviews, and links to booking guides.

5.2.4 Booking Integration:

Seamless integration with booking platforms enables users to directly book accommodations and guide without leaving the travel planner application. Clear pop-ups for the users to proceed with bookings, and confirmation messages should be displayed upon successful reservations.

5.2.5 Payment System:

This system offers a seamless booking experience for users regardless of the booking amount. The payment gateway ensures secure transactions by encrypting sensitive user information. Once the payment is successful, the user receives a confirmation email with details of the booking and payment.

5.2.6 Feedback Mechanism:

Incorporating feedback mechanisms such as ratings, reviews, and surveys allows users to provide input on their travel experiences and helps the platform improve its services continually.

5.3 Snapshots of System and Brief Description

The snapshots of various modules of the system would typically encompass a comprehensive overview of the different components or sections that make up the entire system:

5.3.1 The home page or the landing page

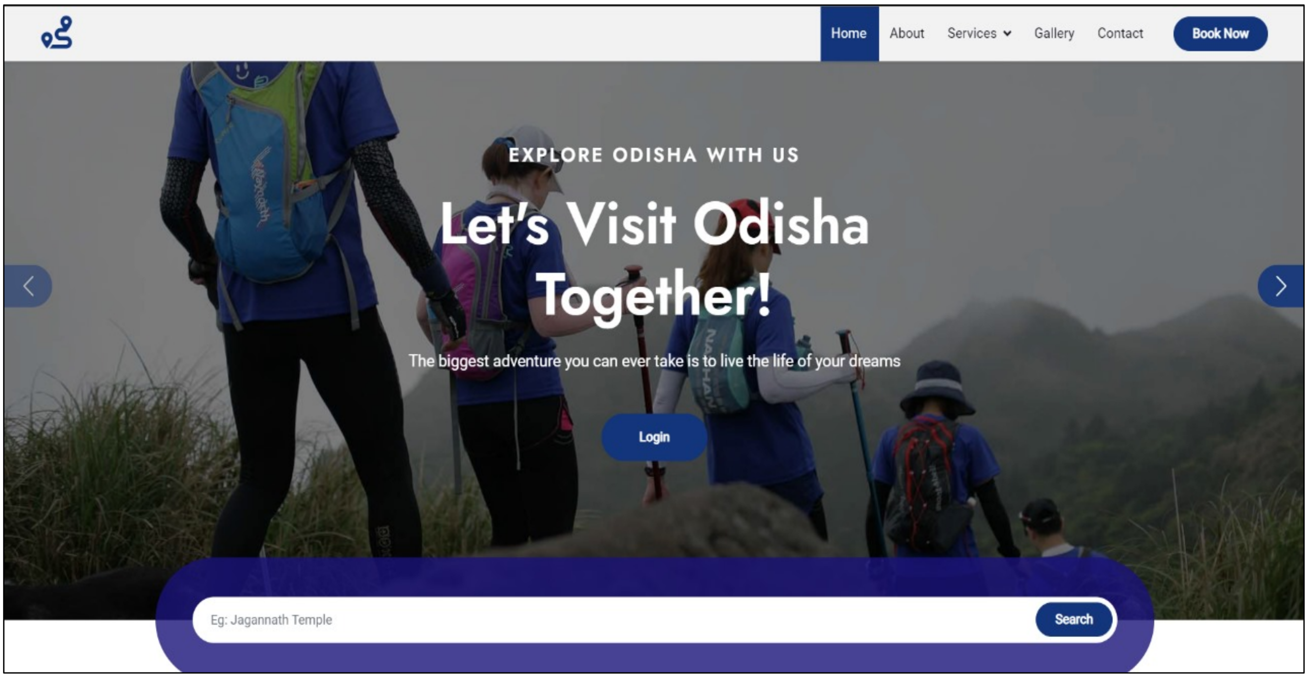
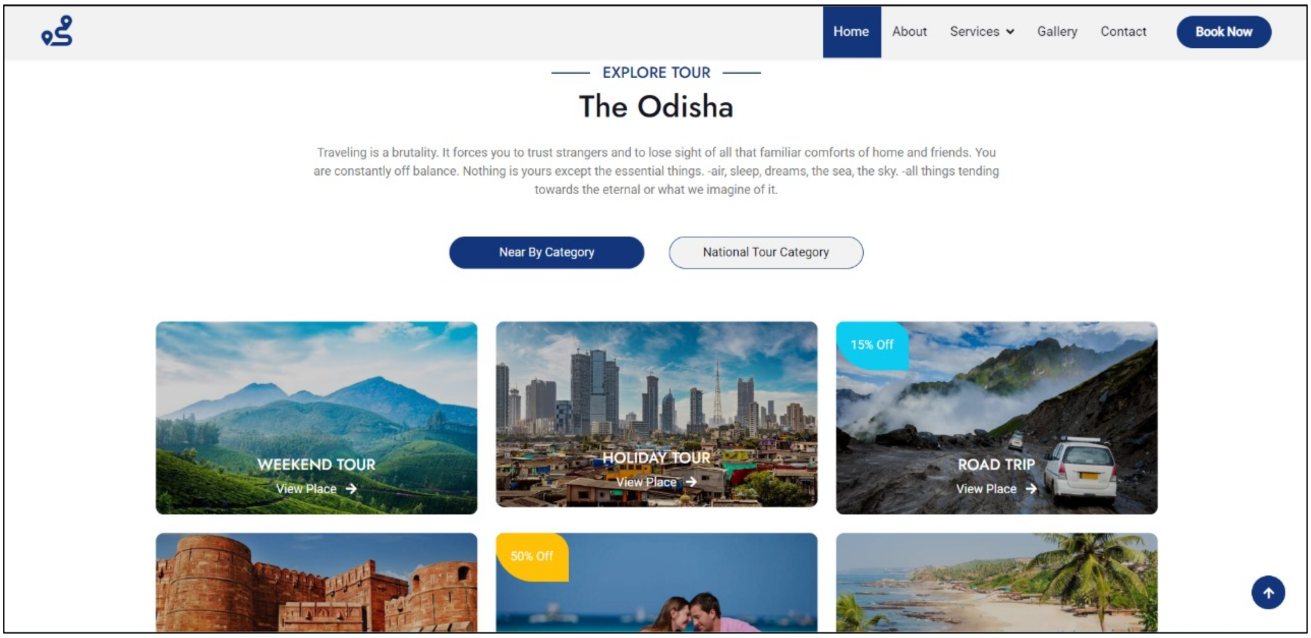

Home
About
Services
Gallery
Contact
Book Now
EXPLORE ODISHA WITH US
Let's Visit Odisha Together!
The biggest adventure you can ever take is to live the life of your dreams
Login
Eg: Jagannath Temple
Search

Figure 5.1: Home Page



Traveling is a brutality. It forces you to trust strangers and to lose sight of all that familiar comforts of home and friends. You are constantly off balance. Nothing is yours except the essential things. -air, sleep, dreams, the sea, the sky. -all things tending towards the eternal or what we imagine of it.

Figure 5.2: Recommendations

The home page module is serving as the first point of interaction for users. It aims to provide a visually appealing and user-friendly interface that entices visitors to explore further and begin planning their travels. The home page module typically consists of various components designed to engage users, showcase essential features, and facilitate seamless navigation.

1. **Header:** The header section typically contains the project logo and various navigation menu. It allows users to easily navigate to different sections of the website such as home, about, services, gallery or access their accounts.
2. **Hero section:** This is the most prominent section of the home page, usually located at the centre. It often includes captivating images or videos related to travel destinations, along with a catchy headline or tagline that captures the essence of the project. The hero section aims to grab the user's attention and create a positive first impression.
3. **Search bar:** A prominent search bar enables users to quickly search for destinations, flights, hotels, or activities based on their preferences.
4. **Footer:** The footer section typically contains links to important pages such as About Us, Contact, Privacy Policy, etc. It also serves as a secondary navigation area and may include social media icons for users to connect with the project on various platforms.

5.3.2 Destination Suggestions Sections:

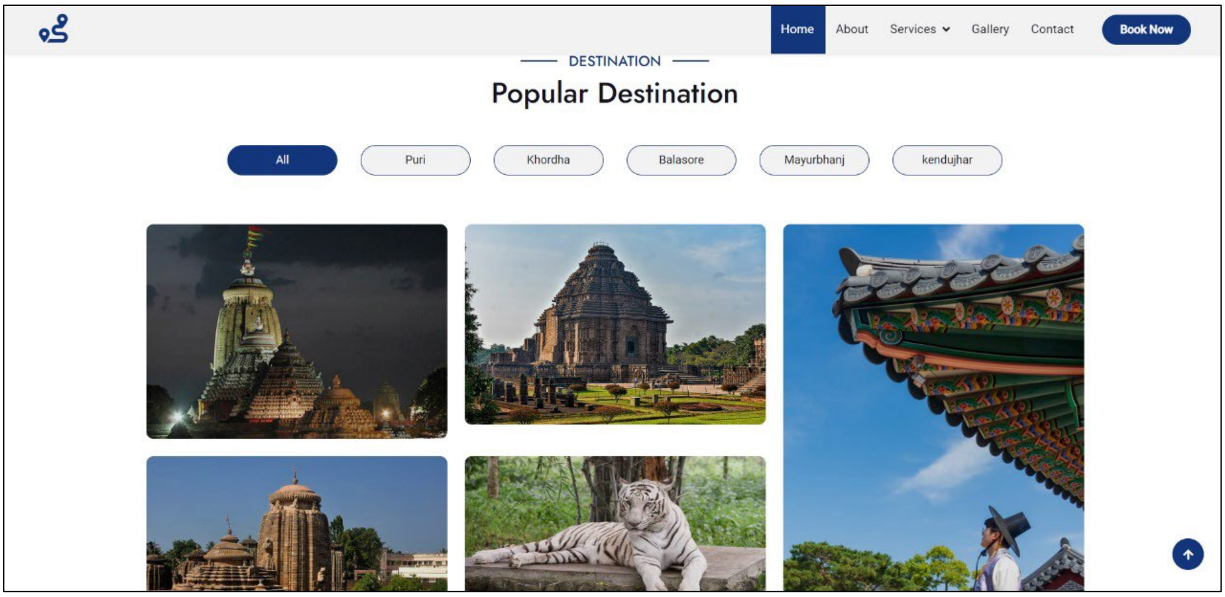


Figure 5.3: Destination Suggestions

Showcase a selection of popular or trending travel destinations to inspire users and spark their wanderlust. Each destination card includes a very brief description, captivating images, and a call-to-action button to explore further details. Recommendations are tailored to the user's travel style and preferences, leading to more fulfilling trips.

5.3.3 Login And Sign Up Section:

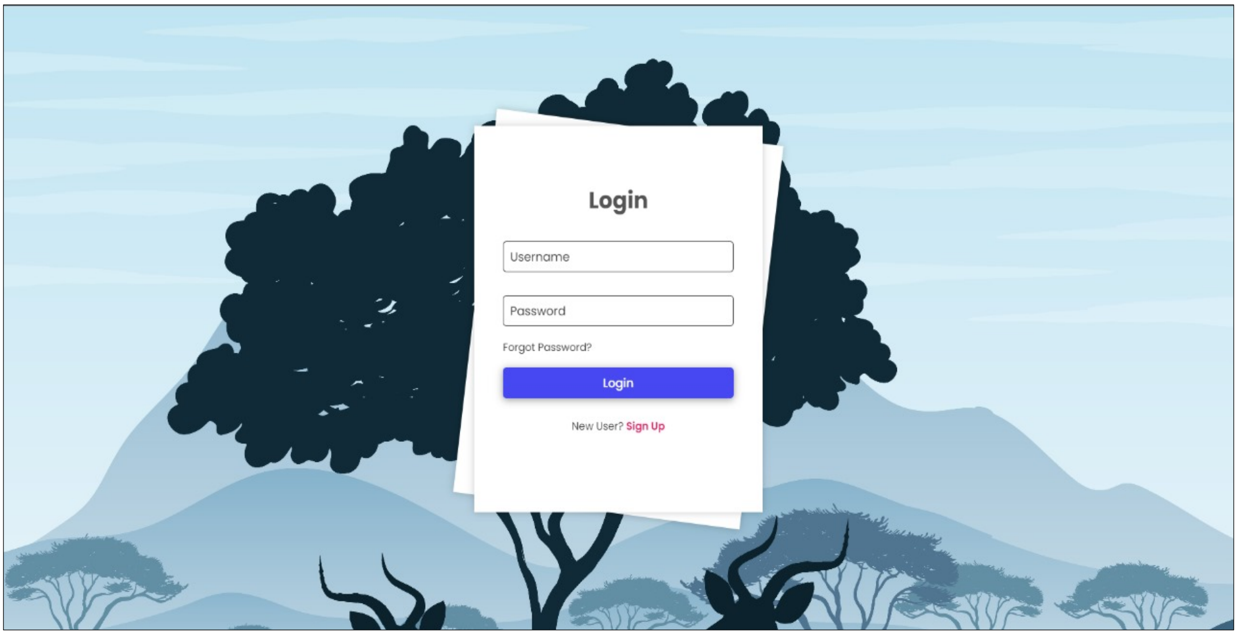


Figure 5.4: Login Page

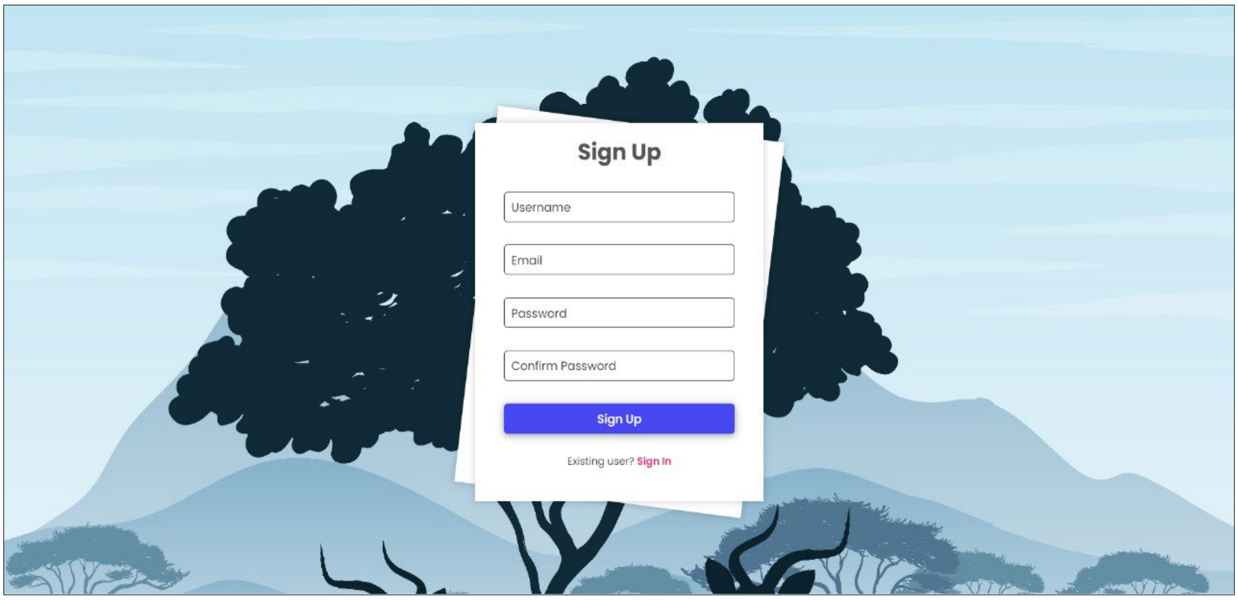


Figure 5.5: Sign Up Page

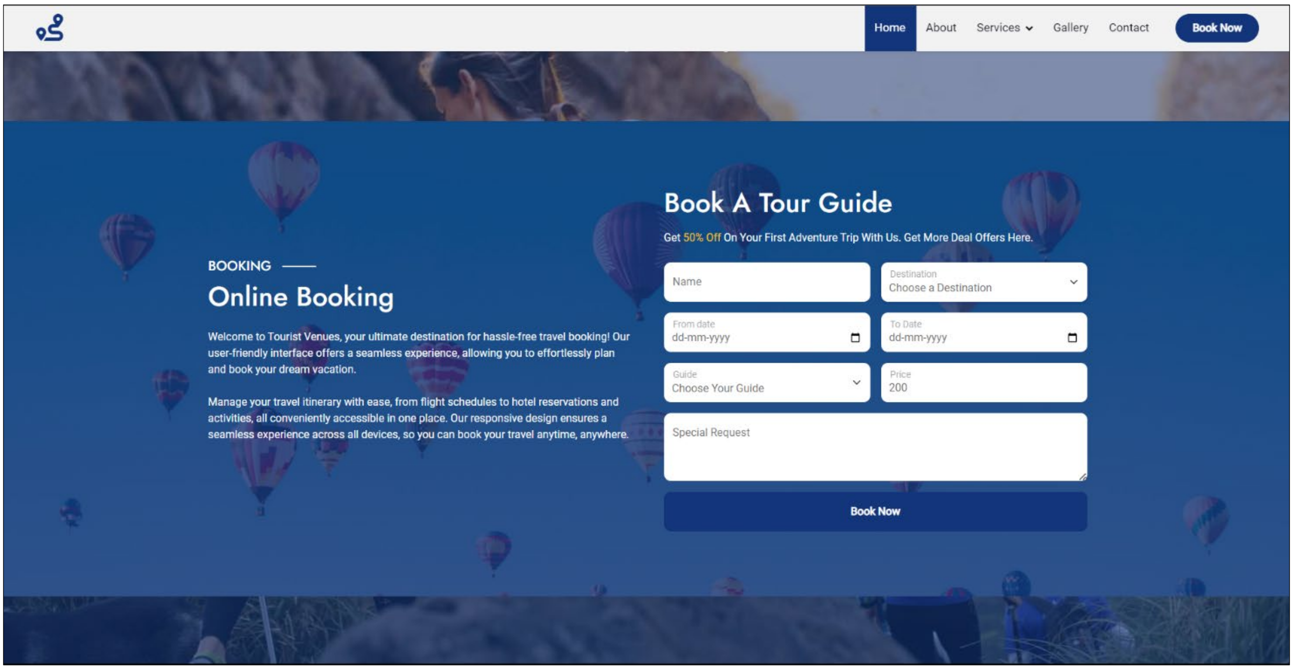
In the realm of travel planner project, a login and sign-up page module serves as the gateway for users to access the platform's features. These components play a crucial role in not only authenticating users but also in managing their accounts and personal information securely.

1. **Login Form:**
   - Include fields for username/email and password.
   - Utilize client-side validation for immediate feedback on input errors.
   - Implement features like "Forgot Password" for password recovery.
2. **Registration Form:**
   - Include fields for unique username, email, password and confirm password.
   - Use password strength indicators to encourage users to create secure passwords.
   - Implementation of email verification to ensure the validity of user accounts.

5.3.4 Payment Section:

The payment form is a critical component of our travel planner project, facilitating secure transactions for users booking travel services such as flights, accommodations, and activities. This section outlines the key features and functionalities of the payment form.

The primary objective of the payment form is to streamline the booking process by enabling users to make payments conveniently and securely. It aims to provide a seamless experience while ensuring the confidentiality and integrity of users' financial information.



BOOKING

Online Booking

Welcome to Tourist Venues, your ultimate destination for hassle-free travel booking! Our user-friendly interface offers a seamless experience, allowing you to effortlessly plan and book your dream vacation.

Manage your travel itinerary with ease, from flight schedules to hotel reservations and activities, all conveniently accessible in one place. Our responsive design ensures a seamless experience across all devices, so you can book your travel anytime, anywhere.

Book A Tour Guide

Get 50% Off On Your First Adventure Trip With Us. Get More Deal Offers Here.

Name

Destination: Choose a Destination

From date: dd-mm-yyyy

To Date: dd-mm-yyyy

Guide: Choose Your Guide

Price: 200

Special Request

Figure 5.6: Payment Section

5.4 Back Ends Representation

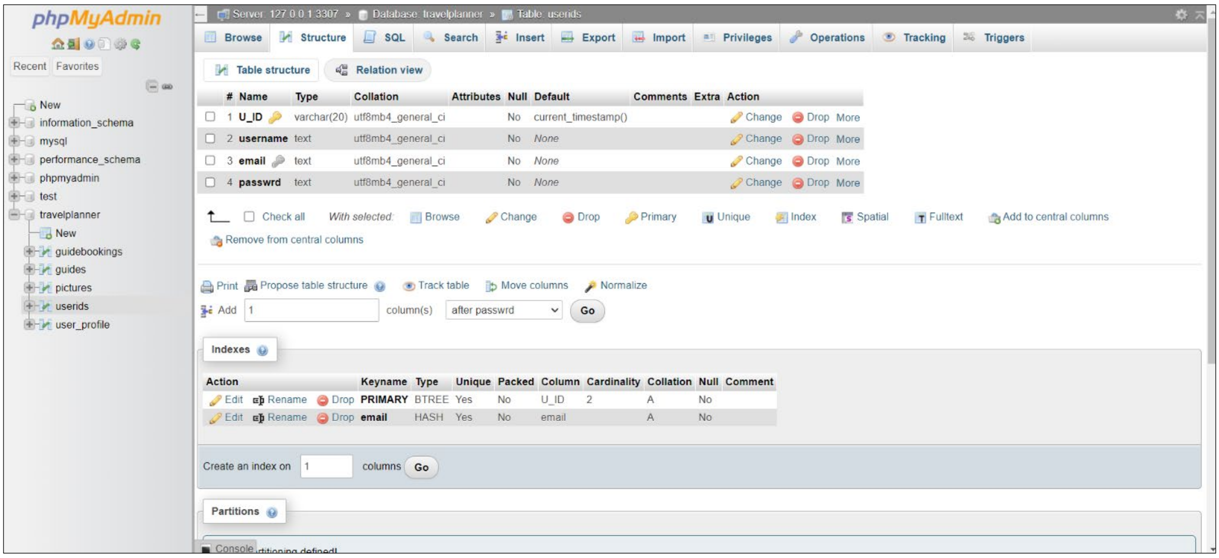
The backend and database structure of the travel planner project are essential components that enable the functionality and storage necessary for the application. By employing a normalized database structure with well-defined relationships, the travel planner project can efficiently manage data and provide a robust foundation for various functionalities.

- Users table stores user information like username, email address, and a secure password hash (using algorithms like bcrypt). It helps in verifying user credentials (username and password) during login attempts, ensuring only authorized users can access their accounts.
- A separate database contains detailed travel guide information for various destinations. Tour guide parameters encompass personal details and characteristics that define an effective guide. This includes strong communication and other skills to captivate audiences.
- Linking tables with unique attributes allows to create a more structured and efficient database for the travel planner project, enabling seamless data management and retrieval for a smooth user experience.
- Separate tables allow for optimized indexing, which speeds up data retrieval. Separate tables facilitate a more efficient search process. They ensure data integrity, improve performance, and make future scalability easier to manage.
- When a user searches and views a destination, the system retrieves the referenced image path from the database and displays the picture. This approach provides a visually appealing user experience while maintaining a clean and efficient database backend.
- By maintaining this dedicated table with foreign keys linking to Users, Guides, and Destinations, helps in creating a comprehensive record of guide bookings. This allows users to view and manage their bookings, helps manage guide availability, and facilitates communication between users and guides.

5.5 Snapshots of Database Tables and Descriptions

This section likely refers to the design of the travel planner project's backend. It outlines the structure and purpose of the core database tables. The following database tables are utilized within the travel planner project:

1. User IDs Database:



Server: 127.0.0.1:3307 » Database: travelplanner » Table: userids

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**U_ID**	varchar(20)	utf8mb4_general_ci		No	current_timestamp()			Change Drop More
□	2	**username**	text	utf8mb4_general_ci		No	*None*			Change Drop More
□	3	**email**	text	utf8mb4_general_ci		No	*None*			Change Drop More
□	4	**passwrd**	text	utf8mb4_general_ci		No	*None*			Change Drop More

Add 1 column(s) after passwrd Go

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	**PRIMARY**	BTREE	Yes	No	U_ID	2	A	No	
Edit Rename Drop	**email**	HASH	Yes	No	email		A	No	

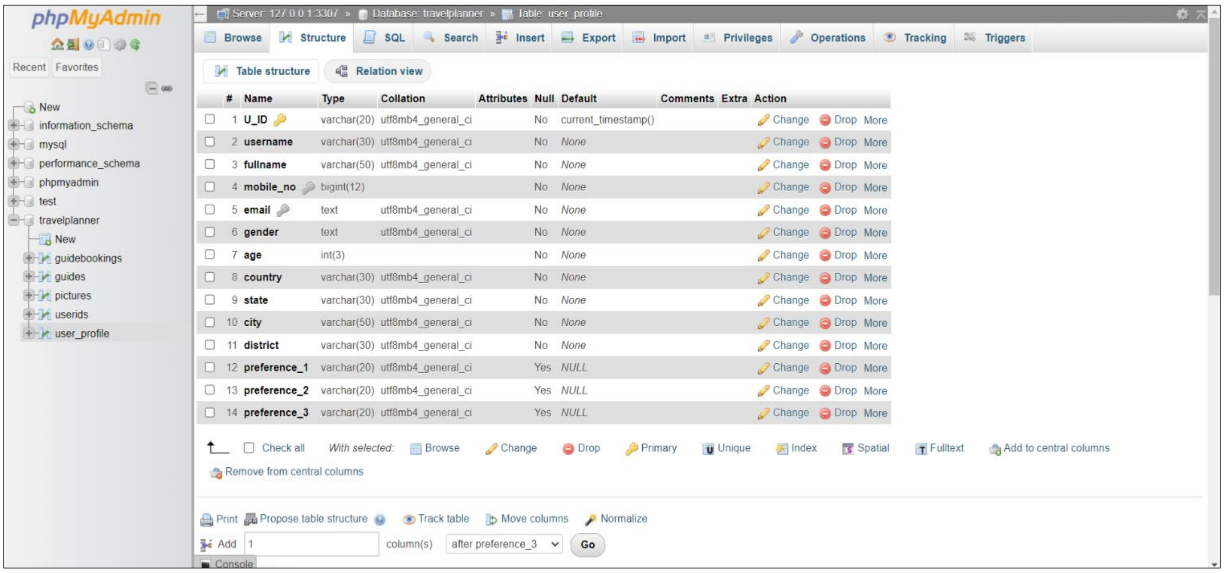
Create an index on 1 columns Go

No partitioning defined!

Figure 5.7: Database of User_IDs

The user ID database is a pivotal component of the Travel Planner project, serving as the foundation for user management and authentication within the system. In the context of the project, the user ID database is a structured collection of user information, securely stored and managed to facilitate user access and personalization features. Key features and functionalities of the user ID database within the Travel Planner project include user Authentication, User Profiles and access control. Overall, the user ID database plays a central role in the Travel Planner project, providing the necessary infrastructure for user management, authentication, and personalization features. By maintaining a secure and scalable database environment.

2. User profile Database:



Server: 127.0.0.1:3307 » Database: travelplanner » Table: user_profile

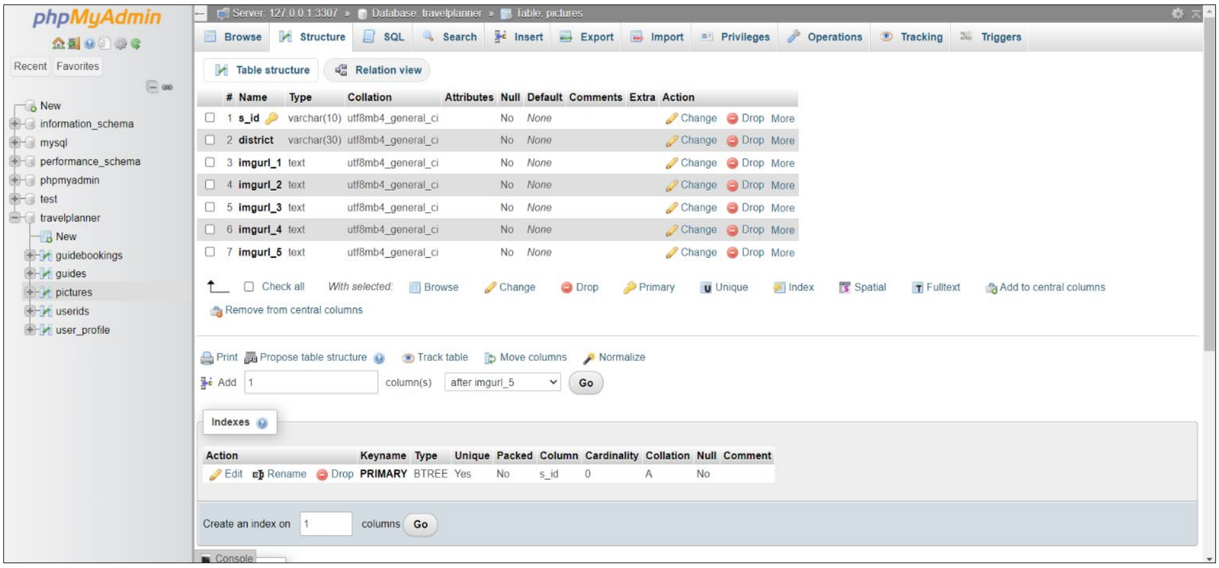
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	U_ID	varchar(20)	utf8mb4_general_ci		No	current_timestamp()			Change Drop More
2	username	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
3	fullname	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
4	mobile_no	bigint(12)			No	None			Change Drop More
5	email	text	utf8mb4_general_ci		No	None			Change Drop More
6	gender	text	utf8mb4_general_ci		No	None			Change Drop More
7	age	int(3)			No	None			Change Drop More
8	country	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
9	state	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
10	city	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
11	district	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
12	preference_1	varchar(20)	utf8mb4_general_ci		Yes	NULL			Change Drop More
13	preference_2	varchar(20)	utf8mb4_general_ci		Yes	NULL			Change Drop More
14	preference_3	varchar(20)	utf8mb4_general_ci		Yes	NULL			Change Drop More

Add 1 column(s) after preference_3 Go

Figure 5.8: Database of User Profile

The user profile database is a crucial component of the travel planner project, serving as a repository for storing and managing user information related to travel preferences, itineraries, and other relevant data. Various attributes are used to store the information of the user such as user_id, username, full name, mobile_no, email, age, country, state, district. The database also enables the backend system to save the preference of the user in the database.

3. Pictures Database:



Server: 127.0.0.1:3307 » Database: travelplanner » Table: pictures

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	s_id	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
2	district	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
3	imgurl_1	text	utf8mb4_general_ci		No	None			Change Drop More
4	imgurl_2	text	utf8mb4_general_ci		No	None			Change Drop More
5	imgurl_3	text	utf8mb4_general_ci		No	None			Change Drop More
6	imgurl_4	text	utf8mb4_general_ci		No	None			Change Drop More
7	imgurl_5	text	utf8mb4_general_ci		No	None			Change Drop More

Add 1 column(s) after imgurl_5 Go

Indexes

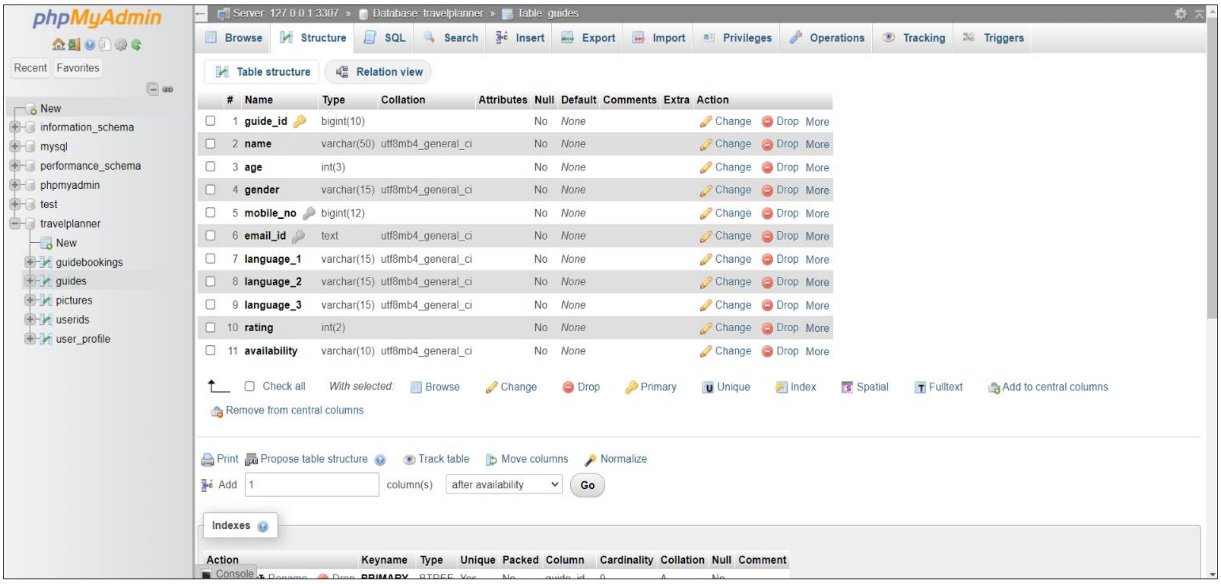
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	s_id	0	A	No	

Create an index on 1 columns Go

Figure 5.9: Database of Pictures

The picture database typically comprises several attributes, among which are the district name and one or more image URLs associated with it. These attributes allow for efficient organization and retrieval of visual content based on user queries or preferences. District attribute serves as a primary identifier for different geographical areas or regions within the travel planner's scope. Districts are based on administrative boundaries. Image_urls attribute represents a URL links pointing to the primary image associated with a particular destination of a district.

4. Guides Database:



Server: 127.0.0.1:3307 » Database: travelplanner » Table: guides

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	guide_id	bigint(10)			No	None			Change Drop More
2	name	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
3	age	int(3)			No	None			Change Drop More
4	gender	varchar(15)	utf8mb4_general_ci		No	None			Change Drop More
5	mobile_no	bigint(12)			No	None			Change Drop More
6	email_id	text	utf8mb4_general_ci		No	None			Change Drop More
7	language_1	varchar(15)	utf8mb4_general_ci		No	None			Change Drop More
8	language_2	varchar(15)	utf8mb4_general_ci		No	None			Change Drop More
9	language_3	varchar(15)	utf8mb4_general_ci		No	None			Change Drop More
10	rating	int(2)			No	None			Change Drop More
11	availability	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More

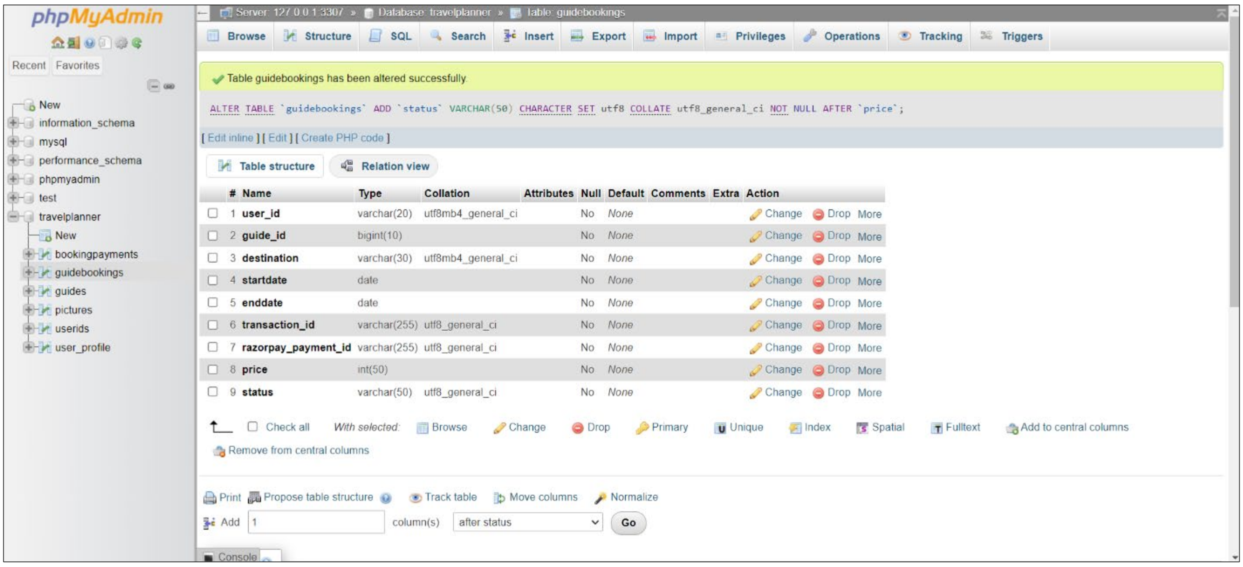
Add 1 column(s) after availability

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Rename Drop	PRIMARY	BTREE	Yes	No	guide_id	0	A	No	

Figure 5.10: Database of Guides

This database aims to enhance the user experience by offering access to knowledgeable guides who can offer personalized recommendations and insights tailored to each traveller's preferences. Human guide enlisted in the database possesses distinct expertise and specialization in particular regions, activities, or travel styles. Feedback from previous travellers forms a crucial aspect of the database. Reviews and ratings provided by users who have engaged the services of a particular guide help prospective travellers gauge the quality of service and expertise offered. The guide database allows the user to store information of the guide such as guide id, name, age, gender, mobile number, email ID, language proficiency 1, language proficiency 2, language proficiency 3 and ratings. All of these attributes help to put proper rating to the guide which helps to recommend the guide with higher rating to the users.

5. User Booking Database



✓ Table guidebookings has been altered successfully.

```
ALTER TABLE `guidebookings` ADD `status` VARCHAR(50) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL AFTER `price`;
```

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	user_id	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
□	2	guide_id	bigint(10)			No	None			Change Drop More
□	3	destination	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
□	4	startdate	date			No	None			Change Drop More
□	5	enddate	date			No	None			Change Drop More
□	6	transaction_id	varchar(255)	utf8_general_ci		No	None			Change Drop More
□	7	razorpay_payment_id	varchar(255)	utf8_general_ci		No	None			Change Drop More
□	8	price	int(50)			No	None			Change Drop More
□	9	status	varchar(50)	utf8_general_ci		No	None			Change Drop More

Add: 1 column(s), after status

Figure 5.11: Database of User Bookings

A "User Booking Database" serves as the backbone of any travel planner project, facilitating seamless transactions and managing crucial user information. Let's delve into the structure and significance of such a database, outlining its essential attributes:

- **Transaction ID:** An identifier assigned to each transaction initiated by a user. It helps in tracking the payment process and associating it with specific bookings.
- **RazorPay ID:** Razorpay is a popular payment gateway solution used for processing online payments. Storing the Razorpay ID associated with each booking allows for seamless integration with the payment system and easy reconciliation of transactions.
- **Status:** This attribute indicates the current status of a booking, such as 'Confirmed,' 'Pending,' or 'Cancelled.' Tracking the status helps in managing bookings effectively and providing timely updates to users.

CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

In the culmination of our travel planner project, we find ourselves reflecting on the journey we undertook to create a comprehensive and user-friendly tool for travellers. From the inception of the idea to the final implementation, our team dedicated countless hours to ensure that the travel planner not only meets but exceeds the expectations of users seeking a seamless and enjoyable travel experience. The integration of artificial intelligence and machine learning algorithms allowed us to provide personalized recommendations, enhancing the user's ability to create a tailored itinerary. Regular updates and continuous improvements will be implemented to keep pace with changing trends and emerging destinations. The success of our travel planner project wouldn't have been possible without the collaborative effort of our team. Each member brought up skills and perspectives to the table and fostering a creative and innovative environment. In closing, we express our gratitude to everyone who contributed to the success of this project. The journey may be ending, but the memories and experiences our travel planner helps create will endure, shaping the way people explore the world for years to come. Safe travels and happy planning!

6.2 Future Scope

The future scope for a travel planner project is vast and promising, as the travel industry continues to evolve with technological advancements. Here are some potential areas of growth and development for a travel planner project:

1. **Personalized AI-driven Recommendations:**

   Implementing advanced artificial intelligence algorithms can enable the travel planner to provide highly personalized recommendations based on individual preferences, past travel history, and real-time data. This can include personalized destination suggestions, activity recommendations, and accommodation options.

2. **Smart Travel Itineraries with Real-Time Updates:**

   Enhance the travel planning process by incorporating real-time data updates. This includes live information on weather conditions, local events, and traffic updates, allowing users to adjust their itineraries on the go for a seamless travel experience.

3. **Continuous Machine Learning for Improved Recommendations:**

   Employing machine learning algorithms that continuously learn from user behaviour and feedback can refine and improve the accuracy of travel recommendations over time. This ensures that the travel planner stays up-to-date with changing user preferences and emerging travel trends.

4. **Global Expansion and Multi-Language Support:**

   To cater to a diverse user base, expanding the travel planner's reach globally and offering support for multiple languages can be a key factor for success. This involves partnerships with service providers worldwide and localization of content.

REFERENCES

1. https://www.makemytrip.com/ : last accessed 02/04/2024

2. https://www.easemytrip.com/ : last accessed 02/04/2024

3. https://www.php.net/manual/en/book.mysql.php : last accessed 05/04/2024

4. https://app.diagrams.net / : last accessed 08/04/2024

5. https://staruml.io/ : last accessed 08/04/2024

6. https://youtu.be/1xtrIEwY_zY : last accessed 07/04/2024

7. https://www.geeksforgeeks.org/ : last accessed 03/04/2024

8. https://www.gliffy.com/blog/guide-to-flowchart-symbols : last accessed 08/04/2024