

Abstract

The Odyssey Travel website revolutionizes the way individuals plan and experience their journeys, providing a comprehensive platform that caters to all travel needs. Embracing the essence of convenience and personalization, Odyssey Travel offers a diverse range of travel packages, allowing users to effortlessly explore destinations and select their preferred accommodations and transportation options. The platform's intuitive interface ensures seamless navigation, empowering users to customize their travel plans to match their unique preferences. By bridging the gap between travelers and local service providers, Odyssey Travel promotes cultural exchange and supports local economies. The website not only simplifies the travel booking process but also enhances the overall travel experience by offering tailored recommendations and insights. With a commitment to customer satisfaction and innovation, Odyssey Travel is the ultimate companion for anyone looking to embark on unforgettable adventures.

Contents

1	Overview	5
2	Introduction	6
2.1	Purpose	6
2.2	Scope	6
2.3	Organization	6
2.4	References	7
3	Glossary	8
3.1	Web Framework	8
3.2	Supported Devices	8
3.3	DFD (Data Flow Diagram)	8
3.4	Interface	8
3.5	Maintenance prediction	8
3.6	Object Model	8
3.7	Object-Oriented (OO) Approach	8
3.8	Reliability	9
3.9	Requirement, functional	9
3.10	Requirement, non-functional	9
3.11	Sequence Diagram	9
3.12	Software Requirements Specification	9
3.13	Software Architecture	9
3.14	Stakeholder	9
3.15	User	9
3.16	User Interface	9
3.17	Unified Modeling Language (UML)	10
3.18	Use Cases	10
4	User Requirements Definition	11
4.1	External Interface Requirements	11
4.1.1	User Interface	11
4.2	Functional Requirements	12
4.3	Non Functional Requirements	13
5	System Architecture	14
5.1	Product perspective and functions	14
5.2	System Environment	15
5.3	User characteristics	16

5.4	Assumptions and dependencies	17
6	System Requirement Specification	18
6.1	External Interface Requirements	18
6.1.1	User Interfaces – GUI	18
6.1.2	Hardware and software interfaces	19
6.1.3	Communications interfaces	20
6.2	Functional Requirements	20
6.3	Non Functional Requirements	22
6.3.1	Performance Requirements	23
6.3.2	Safety Requirements	23
6.3.3	Software Quality Attribute	23
6.3.4	Reliability	23
6.3.5	Multi lingual Support	23
7	System Models	24
7.1	Use Cases	24
7.2	Use Case Diagram	35
8	System Evolution	37
9	Appendix-I	39
9.1	Level-0 Data Flow Diagram	39
9.2	Level-1 Data Flow Diagram	40

Index of Diagrams

• Figure-5.1: Block Diagram of Travel Agency Software	13
• Figure-5.2: An abstract system environment of Travel Agency Software	14
• Figure-5.3: Logical Structure of Travel Agency Software ...	15
• Figure-7.1: Use Case 01	23
• Figure-7.2: Use Case 02	24
• Figure-7.3: Use Case 03	26
• Figure-7.4: Use Case 04	27
• Figure-7.5: Use Case 05	29
• Figure-7.6: Use Case 06	30
• Figure-7.7: Use Case 07	32
• Figure-7.8: Use Case 08	33
• Figure-7.9: Use Case Diagram For Travel Agency Software	35
• Figure-9.1: Level 0 DFD of Travel Agency Software	38
• Figure-9.2: Level 1 DFD of Travel Agency Software	39

1 Overview

Preface

The Software Requirements Specification (SRS) provides a comprehensive outline of what the developers need to create for the system. It encompasses both user requirements and a thorough description of system specifications. This document starts with an overview of user needs, transitioning into detailed system requirements. As an external contractor tasked with developing the system, Efficiency Monitor, we recognize the importance of this SRS in defining the technical and non-functional requirements of the software. It ensures that we don't overlook system-wide requirements while concentrating on functional aspects. In essence, this SRS document will primarily delineate user requirements along with overarching non-functional system requirements.

Readership

This requirements document is designed for a broad audience, including senior management, software engineers, and everyone involved in the project. Its primary audience includes customers, system analysts, project managers, designers, developers, and testers. Essentially, anyone connected to the development and maintenance of the software will refer to this document.

The SRS will be reviewed by our customers to confirm that it accurately reflects the requirements gathered from user interviews. Designers will use it as a fundamental resource for the design phase. Developers and testers will rely on this document to understand their tasks and ensure they meet the specified requirements. In summary, this document serves as a critical reference point for all stakeholders involved in the software's lifecycle.

Version History

Since this is the introductory version of the "Odyssey Travels" application, there is no previous history or version to reference. In version 1.0 of Odyssey Travels, our goal is to meet all identified user requirements gathered through extensive study and customer interviews (as detailed in Chapter 4 of this document). Should users encounter any limitations or shortcomings while using our product and provide feedback or request new features, we will release updated versions of Odyssey Travels. These updates will address any reported issues, incorporate requested features, or introduce enhancements based on our own assessments.

2 Introduction

2.1 Purpose

The purpose of this document is to provide a comprehensive description of the web-based project named "Odyssey Travels," developed using Next.js. It aims to outline the system's objectives, functionalities, user interfaces, operational constraints, and how it handles external interactions. This document serves as a detailed guide for stakeholders and developers involved in the project, ensuring a clear understanding of the system's scope and requirements. It will define how "Odyssey Travels" enhances the online travel booking experience through innovative features and responsive web interfaces.

2.2 Scope

This web-based application, "Odyssey Travels," is designed to enhance user efficiency and streamline travel planning processes. It aims to empower users by providing intuitive tools to manage and prioritize travel itineraries and bookings seamlessly. Users will no longer need to rely on traditional methods like spreadsheets or multiple websites to organize their travel plans. "Odyssey Travels" will enable users to categorize and prioritize travel activities effortlessly, ensuring optimal utilization of their time and resources. By offering clear insights into itinerary management and suggesting efficient scheduling of activities, the application enhances user productivity while maintaining ease of use. Specifically, the system will guide users in making informed decisions about their travel plans, suggesting ideal times for activities to maximize efficiency throughout their journey. It will help users save valuable time by centralizing booking processes and providing real-time updates on travel arrangements. Additionally, the application will assist users in optimizing their itineraries by recommending adjustments or identifying unnecessary tasks, ensuring that their travel experiences are both productive and fulfilling. Overall, "Odyssey Travels" aims to revolutionize the travel planning experience by offering a user-friendly interface that simplifies task management and enhances productivity, thereby meeting the diverse needs of modern travelers.

2.3 Organization

This document consists of 7 chapters, each serving a distinct purpose in detailing the "Odyssey Travels" project. The following chapter, the glossary section, provides definitions for technical terms used throughout this document, aimed at enhancing readability

and understanding for all stakeholders. The third section, titled User Requirement Definition, outlines the services offered to users and encompasses non-functional system requirements. This section utilizes natural language, diagrams, and other notations that are accessible and comprehensible to our customers. Chapter 4, System Architecture, offers a high-level overview of the planned system architecture, illustrating how functions are distributed across various modules. It also highlights components that are reused within the architecture. The subsequent chapter, System Requirement Specification, delves deeper into both functional and non-functional requirements, offering detailed descriptions and specifications. Graphical system models depicting relationships between system components and their environment are included in Chapter 6, titled System Models. Chapter 7, System Evolution, outlines the core assumptions underpinning the system's design and anticipates future changes influenced by hardware advancements and evolving user needs. Appendices containing detailed, application-specific information relevant to "Odyssey Travels" are appended after the 7th section, providing supplementary details for stakeholders and developers.

2.4 References

IEEE: IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications, IEEE Computer Society, 1998.

3 Glossary

3.1 Web Framework

Next.js – A JavaScript framework designed for building server-side rendered (SSR) React applications, developed by Vercel.

3.2 Supported Devices

Next.js application – Any web application built using Next.js. In this document, synonymous with web application developed with Next.js

3.3 DFD (Data Flow Diagram)

A graphical representation that refers the system using functions performed in the system and data produced by these functions.

3.4 Interface

A specification of the attributes and operations associated with a software component. The interface is used as the means of accessing the component's functionality.

3.5 Maintenance prediction

The process in which project managers try to predict what system changes might be proposed and what parts of the system are likely to be the most difficult to maintain.

3.6 Object Model

A model of a software system that is structured and organized as a set of object classes and the relationships between these classes.

3.7 Object-Oriented (OO) Approach

An approach to software development where the fundamental abstractions in the system are independent objects.

3.8 Reliability

The ability of a system to deliver services as specified. Reliability can be specified quantitatively as a probability of failure on demand or as the rate of occurrence of failure.

3.9 Requirement, functional

A statement of some function or feature that should be implemented in a system.

3.10 Requirement, non-functional

A statement of a constraint or expected behavior that applies to a system. This constraint may refer to the emergent properties of the software that is being developed or to the development process.

3.11 Sequence Diagram

A diagram that shows the sequence of interactions required to complete some operation. In the UML, sequence diagrams may be associated with use cases.

3.12 Software Requirements Specification

A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.

3.13 Software Architecture

A model of the fundamental structure and organization of a software System.

3.14 Stakeholder

Any person with an interest in the project who is not a developer.

3.15 User

Someone who interacts with the Web Browser.

3.16 User Interface

The way in which information produced by the system is displayed and the way in which system users can access system functionality.

3.17 Unified Modeling Language (UML)

A graphical language used in object-oriented development that includes several types of system models that provide different views of a system.

3.18 Use Cases

Use cases are a requirements discovery technique that identifies the actors involved in an interaction and names the type of interaction.

4 User Requirements Definition

This section encompasses all functional and quality requirements of the system, including non-functional requirements. It provides a comprehensive description of the system and its complete feature set. To ensure clarity and avoid misunderstandings, user requirements are articulated in natural language. This version clarifies the purpose of the section while maintaining its original intent and focus.

4.1 External Interface Requirements

4.1.1 User Interface

- **Home:**
 - Serves as the main landing page.
 - Provides an overview of offerings and services.
 - Designed for intuitive navigation and accessibility.
- **About:**
 - Provides comprehensive project information.
 - Includes objectives, mission statement, team members, and background details.
 - Builds trust and transparency.
- **Packages:**
 - Users explore various travel options and service offerings.
 - Includes detailed descriptions, pricing information, and possibly images or videos.
 - Supports informed decision-making.
- **Login:**
 - Registered users access their accounts securely.
 - Provides access to booking history, saved preferences, and account settings.
 - Enhances user engagement and personalized experiences.
- **Signup:**
 - New users create accounts easily.

- Access to exclusive features and personalized recommendations.
- Facilitates seamless booking experiences.
- **Contact Us:**
 - Provides essential contact information.
 - Includes email addresses, phone numbers, and possibly a contact form.
 - Facilitates direct communication with customer support or the project team.

Design Considerations

Each section is designed with:

- User-friendliness in mind.
- Intuitive navigation and accessibility across different devices and platforms.
- Home page as the focal point for engaging users.
- About section for building trust and transparency.
- Packages section for comprehensive showcasing and informed decision-making.
- Seamless Login and Signup process for enhanced engagement and personalized experiences.
- Contact Us section for easy access to customer support and inquiries.
- Promotion of customer satisfaction and support throughout user interaction.

Hardware Interfaces

Odyssey Travels, the hardware interface consists of Android-based smartphones. There are no specific hardware requirements for the Android devices used, except that the system must be compatible with any version of Android

4.2 Functional Requirements

This section outlines the fundamental actions required for the Odyssey Travels software system:

- The system must allow users to view available travel packages and details.
- The system must provide options to filter packages based on destination, duration, and price range.
- The system must display a progress bar indicating booking completion and show the percentage of bookings done.

- The system must offer a statistical view of popular travel destinations, package categories, and user preferences.
- The system must Send Invoice Mail for important booking updates and travel reminders.
- The system must display average user ratings and reviews for each travel package.
- The system must allow users to upload images or documents (e.g., passport copies) securely for booking purposes.
- The system must include a calendar view for users to select travel dates and check availability.
- The system must provide a profile section for users to manage personal information, preferences, and past bookings.
- The system must feature interactive travel quizzes to engage users and provide insights into destinations.
- The system must maintain a history of user bookings and travel preferences for personalized recommendations.
- All functionalities can be accessed and executed using the touchscreen interface of a smartphone.

All functionalities can be accessed and executed using the touchscreen interface of a smartphone.

4.3 Non Functional Requirements

- System must provide a pleasant and user-friendly graphical interface.
- System must provide multilingual advantage, especially Bengali and English.
- System must respond within a few seconds.
- System must work with relatively long input lists.
- System mustn't alter the task list automatically.
- System must restart instantly in case of failure without any data loss.
- System must be usable to every class of users, even if they have no technical knowledge.
- System must work cooperatively with other key processes without causing any delay.

5 System Architecture

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with user and introduce the basic functionality of it. It will also describe what functionality is available for stakeholders. At last, the constraints and assumptions for the system will be presented.

5.1 Product perspective and functions

Odyssey Travels is a web-based software application designed to manage travel plans and bookings efficiently. Users can seamlessly book flights, hotels, and transportation, ensuring a streamlined travel experience. The application provides tools for organizing and managing travel itineraries, allowing users to keep track of their bookings and view relevant travel details. Additional minor functions and enhancements will be detailed later in this report. In this subsection, we will present a block diagram illustrating the main functionalities of our system. The block diagram, shown in Figure 5.1, provides a visual representation of the key features and interactions within Odyssey Travels.

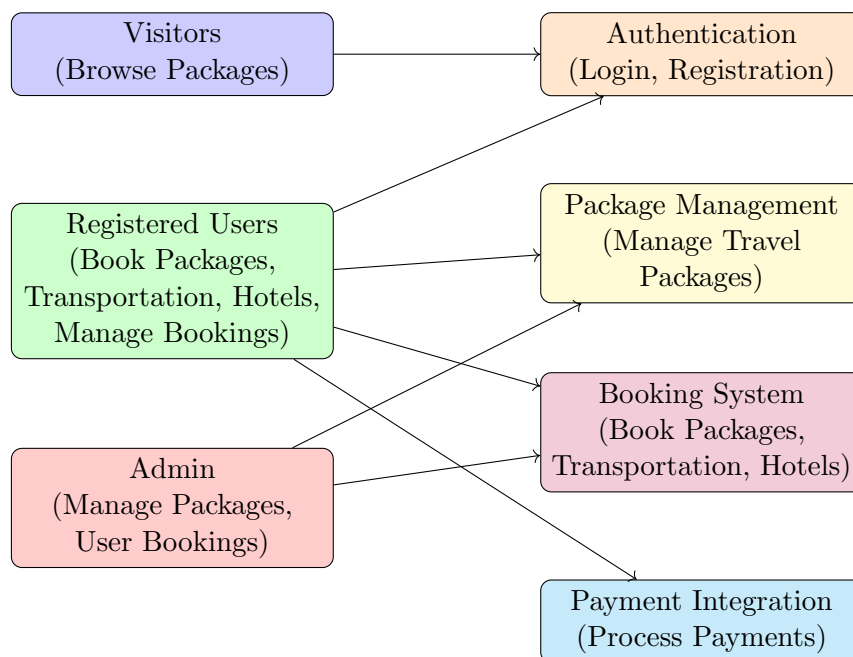


Figure-5.1: Block Diagram of Travel Agency Software

5.2 System Environment

While considering the basic functions/features of Travel Agency Software ,a sample System Environment can be imagined before analyzing the requirements in details. The abstract System Environment of our product is shown in figure-5.2

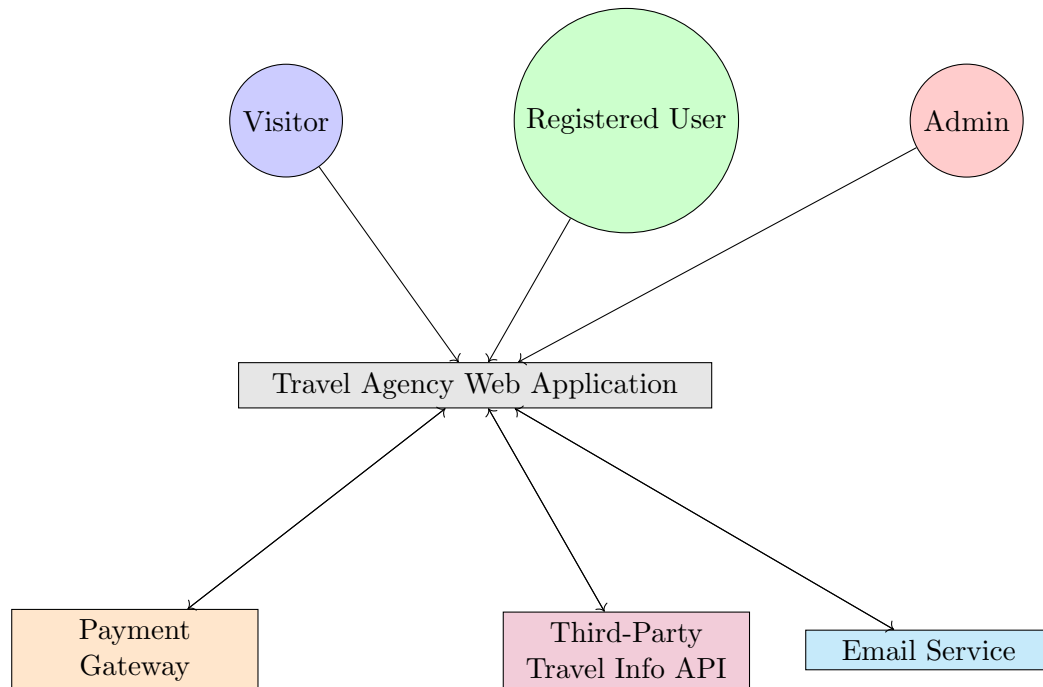


Figure-5.2: An abstract system environment of Travel Agency Software

After studying the requirements mentioned by user listed in section 4 , we have figured out a basic logic structure of Efficiency Monitor mobile application. shown in figure-5.3. That structure basically shows the primary activities, actors, features and primary relationships between them. In general language, logic diagram shows who is doing and what is doing.

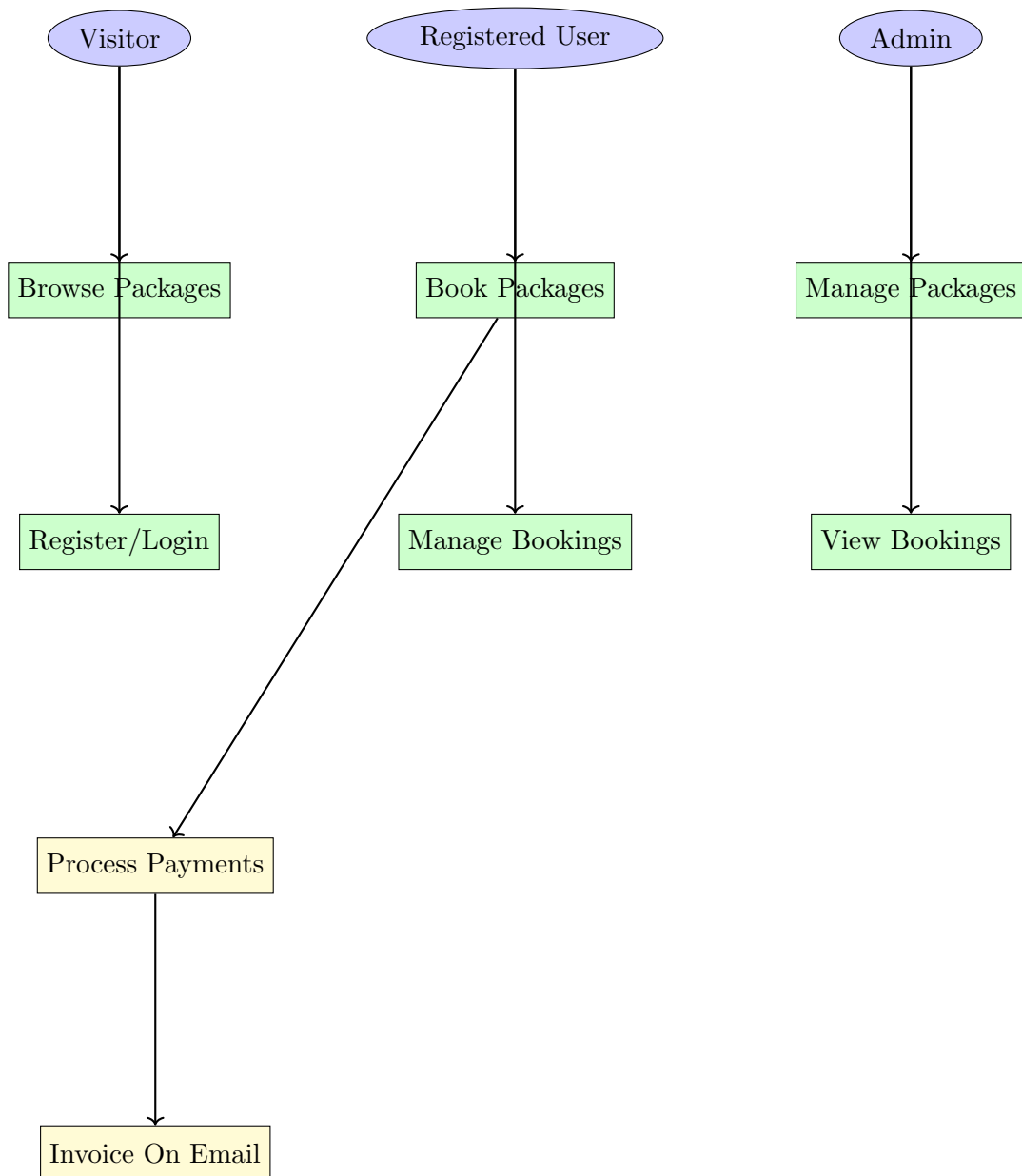


Figure-5.3:Logical Structure of Travel Agency Software

5.3 User characteristics

There is a single type of user interacting with the system: users of the Odyssey Travels web application. Users of our web app have specialized usage tailored to their travel planning and booking needs. The web application allows users to explore travel packages, manage their travel itineraries, and modify their bookings at any time. This means that

users need to be able to view available packages, add new bookings, and update their travel plans seamlessly. For optimal use of our solution, users are expected to be familiar with basic web navigation and interaction. This includes the ability to use buttons, dropdown menus, side panels, and other common web interface elements effectively.

5.4 Assumptions and dependencies

One assumption about the Odyssey Travels application is that it will be used on mobile devices with adequate performance capabilities. If the device lacks sufficient hardware resources, such as processing power, RAM, or storage space, or if these resources are heavily utilized by other applications, the performance of Odyssey Travels may be compromised. In such cases, users may experience issues where the application does not function as intended, or may not work at all.

6 System Requirement Specification

Under this headline we'll describe all the functional and non-functional requirements in details.

6.1 External Interface Requirements

This subsection provides a comprehensive description of all user interface requirements for the Odyssey Travels system. It also includes detailed descriptions of the software interfaces involved in the system's operation.

6.1.1 User Interfaces – GUI

The primary elements of Odyssey Travels' GUI are:

- **Home Page:**
 - Main landing page with navigation to sections such as 'About,' 'Packages,' 'Login,' 'Signup,' and 'Contact Us.'
 - Title bar includes a navigation menu, application logo, and quick access to notifications and user profile.
- **Packages Page:**
 - Displays available travel packages with options to filter and sort based on destination, duration, price, etc.
 - Detailed descriptions, images, and booking options for each package.
 - Users can select packages to view more details or proceed to booking.
- **Login and Signup Pages:**
 - 'Login' page for existing users to enter credentials and access their accounts.
 - 'Signup' page for new users to create accounts by providing name, email, and password.
 - Both pages support secure authentication and password recovery.
- **About Page:**
 - Provides insights into the project's mission, team, and background information.
 - Helps users understand Odyssey Travels' goals and the team behind it.

- **Contact Us Page:**
 - Contains contact information, an inquiry form, and links to support resources.
 - Facilitates direct communication with the Odyssey Travels team.
- **Profile Page:**
 - Accessible from the user icon on the Home page.
 - Options to view and edit personal details, change profile picture, and review booking history.
 - Enables convenient management of user profiles.
- **Invoices::**
 - Invoices for selected travel packages are automatically emailed to users.
 - Each email contains a detailed breakdown of the booking, including costs, services, and travel itinerary.
 - Ensures users receive timely and accurate information about their purchases and can easily access their invoices for record-keeping.
- **Interactive Features:**
 - Home page includes interactive elements like travel quizzes and engagement activities.
 - Enhances user experience with fun exploration of travel-related content and rewards or insights.

6.1.2 Hardware and software interfaces

Odyssey Travels is a mobile-friendly web application for travel planning and booking, accessible via Android (7.0+) and iOS (11.0+) devices using modern web browsers. It supports touch gestures and requires a minimum screen resolution of 720 x 1280 pixels. Push notifications keep users updated on booking details and promotions in real-time. Backend services utilize RESTful APIs for data retrieval and storage of user profiles, travel packages, and bookings, with secure authentication and payment gateways like PayPal and Stripe. Mapping services such as Google Maps or Mapbox provide location and direction functionalities. JavaScript frameworks like React and Next.js optimize frontend functionality, while Axios manages HTTP requests. Cloud storage ensures secure management of user-uploaded documents and images, and email services handle communication with users. Overall, Odyssey Travels maximizes usability and efficiency through its robust hardware and software interfaces for travel planning and booking on mobile devices.

6.1.3 Communications interfaces

Odyssey Travels utilizes HTTP/HTTPS protocols for secure data exchange between clients and servers, ensuring encrypted communication for user interactions and sensitive information. RESTful APIs facilitate seamless integration with backend services, enabling efficient retrieval of travel package data, user profiles, and booking details. Authentication services like OAuth or JWT manage secure user access and session handling, ensuring only authorized interactions within the application. Payment gateways such as PayPal and Stripe handle secure transactions, complying with PCI DSS standards for payment security. SMTP or Email APIs (e.g., SendGrid) manage transactional emails for booking confirmations and notifications, ensuring reliable communication with users. Push notification services (e.g., Firebase Cloud Messaging) deliver real-time alerts on booking updates and promotions, enhancing user engagement. Mapping APIs (e.g., Google Maps, Mapbox) provide location-based services for travel destinations and directions, enhancing the overall travel planning experience within Odyssey Travels.

6.2 Functional Requirements

- **User Authentication and Profile Management:** Secure registration, login, and profile management including personal details and preferences.
- **Travel Package Browsing and Booking:** Browse packages, view details (destinations, itineraries, pricing), and book securely.
- **Notification System:** Notify users about booking confirmations, package updates, and offers via push notifications and emails.
- **Search and Filtering:** Search packages by destination, duration, price range, and travel dates, with filtering options for tailored results.
- **Payment Gateway Integration:** Integrate with secure payment gateways for credit/debit card and other payment methods.
- **User Feedback and Reviews:** Provide feedback and reviews on travel experiences and interact with other users' reviews.
- **Admin Dashboard:** Manage travel packages, user accounts, and monitor booking transactions with an administrative dashboard.

These requirements ensure a user-friendly and comprehensive travel planning and booking experience for users of Odyssey Travels.

- **User Registration and Authentication**

- 1.1. Users can create a new account or log in using existing credentials to access personalized features.

- 1.2. The system securely stores user information, including name, contact details, and login credentials.
- 1.3. User authentication ensures secure access to sensitive information such as booking history and payment details.
- **Travel Package Management**
 - 2.1. The system presents a catalog of travel packages categorized by destination, type (e.g., adventure, cultural), and duration.
 - 2.2. Users can view detailed descriptions, itineraries, inclusions, and exclusions for each package.
 - 2.3. The system allows users to filter packages by price range, departure dates, and specific amenities.
 - 2.4. Users can add packages to a wishlist or compare multiple packages side by side for informed decision-making.
- **Booking and Reservation System**
 - 3.1. Users initiate bookings by selecting preferred travel packages and specifying travel dates.
 - 3.2. The system calculates total costs, including taxes and additional fees, and displays payment options.
 - 3.3. Users can choose payment methods (e.g., credit card, PayPal) and complete transactions securely.
 - 3.4. Confirmation emails are sent to users upon successful booking, containing booking details and itinerary information.
- **User Profile Management**
 - 4.1. Users have access to a profile dashboard where they can view and manage personal information.
 - 4.2. The system allows users to update profile details, such as contact information, travel preferences, and loyalty program memberships.
 - 4.3. Users can upload and manage profile photos or avatars to personalize their accounts.
 - 4.4. Profile settings include privacy options to control visibility of personal details and preferences.
- **Notifications and Alerts**
 - 5.1. The system sends real-time notifications to users regarding booking confirmations, itinerary changes, and special promotions.
 - 5.2. Users can customize notification preferences based on their interests and travel plans.

- 5.3. Alerts include reminders for upcoming trips, payment due dates, and recommendations for additional services.
- **Interactive Maps and Location Services**
 - 6.1. Integrated mapping APIs provide interactive maps displaying travel destinations, nearby attractions, and points of interest.
 - 6.2. Users can search for specific locations, view directions, and explore interactive maps with zoom and pan functionalities.
 - 6.3. The system offers offline access to maps and travel guides, ensuring usability even in areas with limited connectivity.
- **Customer Support and Feedback**
 - 7.1. Users access customer support options through the application, including FAQs, help guides, and contact forms.
 - 7.2. The system facilitates live chat support for immediate assistance with booking inquiries and technical issues.
 - 7.3. Users can submit feedback and reviews about their travel experiences, helping to improve service quality and user satisfaction.
- **Language and Accessibility Options**
 - 8.1. The application supports multiple languages, allowing users to select their preferred language for navigation and communication.
 - 8.2. Accessibility features ensure that the application is usable by users with disabilities, including screen readers and text-to-speech capabilities.
- **System Security and Data Privacy**
 - 9.1. The system employs encryption protocols (e.g., HTTPS) to secure data transmission and protect user information.
 - 9.2. User data, including personal details and payment information, is stored securely and compliant with data protection regulations.
 - 9.3. Regular security audits and updates are conducted to maintain the integrity and confidentiality of user data.

6.3 Non Functional Requirements

Odyssey Travels prioritizes performance with a goal of rapid response times and seamless transaction handling for up to 1000 concurrent users. It ensures high reliability through robust error handling and data recovery mechanisms, aiming for 99.9% availability. Scalability is addressed with a horizontally scalable infrastructure to accommodate peak traffic and increasing user volumes. Usability focuses on an intuitive interface across devices, employing responsive design principles. Compatibility is assured across major web browsers.

and mobile platforms (Android, iOS), ensuring consistent functionality and user experience.

Maintainability is facilitated through best coding practices and comprehensive documentation, supporting efficient system management and future enhancements. These non-functional requirements collectively ensure a secure, scalable, and user-centric travel booking experience with Odyssey Travels.

6.3.1 Performance Requirements

Odyssey Travels is an application that operates efficiently without imposing significant demands on system resources, ensuring minimal impact on other essential processes. It is designed to deliver prompt responses, providing users with a real-time experience for seamless interaction and efficient service delivery.

6.3.2 Safety Requirements

The Odyssey Travels application guarantees the integrity of user-inputted data, ensuring no alterations to these entries. It is designed to operate reliably even with extensive task lists and under various configuration settings, maintaining functionality and performance without compromise.

6.3.3 Software Quality Attribute

Odyssey Travels offers a user-friendly graphical interface designed for ease of use, featuring straightforward functionalities. Users of all levels should navigate the application effortlessly, requiring no specialized knowledge. The system prioritizes quick responsiveness to ensure a seamless user experience.

6.3.4 Reliability

The system must be reliable to users. In case of system failure it must restart instantly without any data loss.

6.3.5 Multi lingual Support

The system should be language agnostic. English and Bengali should be at least be supported because these are the native languages of users.

7 System Models

In this chapter, we will construct abstract models of the system to depict various views or perspectives using predefined graphical notations, particularly the Unified Modeling Language (UML). System modeling serves to elucidate the intended requirements to stakeholders, facilitating discussions on design proposals and providing comprehensive documentation for implementation. This section includes subsections detailing use cases and essential diagrams that illustrate the functionality and structure of our system.

7.1 Use Cases

Use Cases help us to identify the actors involved in an interaction and names the type of interaction. This subsection outlines the Use Cases for our system's users. Use cases are determined following the scenarios mentioned in section-6.

7.1.1. UC 1: Users Viewing Packages

Diagram:

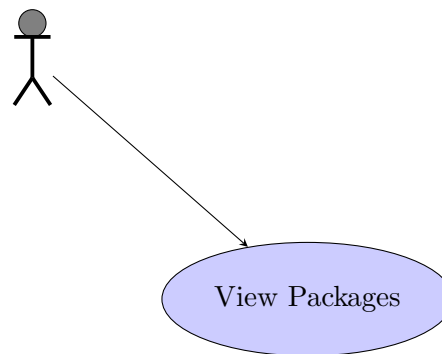


Figure-7.1: Use Case 01

Brief Description: Users view available travel packages on the Odyssey Travels platform.

Preconditions:

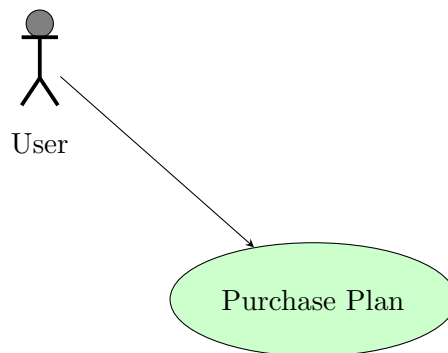
- User has a working internet connection.
- Odyssey Travels application is installed on the device.

Main Success Scenario:

1. User opens the Odyssey Travels application.
2. System displays the home screen with navigation options: Home, About, Packages, Login, Signup, Contact Us.
3. User taps on the "Packages" option from the navigation menu.
4. System shows the "Packages" page, listing various travel packages available.
5. User scrolls through the list of packages to explore details and options.
6. System displays package details including destination, duration, pricing, and amenities.
7. User selects a specific package for more detailed information.
8. System shows detailed information such as itinerary, inclusions, and exclusions.
9. User navigates back to the package list or other sections using navigation buttons.

Postcondition:

- System displays the selected package details accurately.
- User can proceed to book the selected package or explore other options.

7.1.2. UC 2: Purchase Plan**Diagram:****Figure-7.2: Use Case 02**

Brief Description: User selects and purchases a travel plan on the Odyssey Travels platform.

Preconditions:

- User has a working internet connection.
- Odyssey Travels application is installed on the device.
- User is logged into the Odyssey Travels account.

Main Success Scenario:

1. System displays the home page of the Odyssey Travels application.
2. User navigates to the "Packages" section from the navigation menu.
3. System shows a list of available travel packages.
4. User selects a specific travel package for purchase.
5. System displays detailed information about the selected package, including itinerary, pricing, and booking options.
6. User confirms the package selection and taps on the "Book Now" button.
7. System prompts the user to enter necessary details such as travel dates, number of travelers, and any additional preferences.
8. User fills in the required information and proceeds to the payment section.
9. System securely processes the payment using the chosen payment method (e.g., credit card, PayPal).
10. System confirms the successful booking and displays a confirmation message with booking details.
11. User receives a confirmation email with the booking details.

Postconditions:

- User's selected travel plan is successfully booked and confirmed.
- System displays the home page or booking summary for the user to review.

Alternative Courses:

- 5a. User explores additional details or options before making a final selection.
- 7a. User adjusts travel details or preferences before confirming the booking.
- 9a. Payment transaction fails or encounters issues; system prompts user to retry or choose an alternative payment method.

Exceptions:

- User may abandon the booking process at any step before final confirmation.

7.1.3. UC 3: Purchase Plan with Apply Coupon

Diagram:

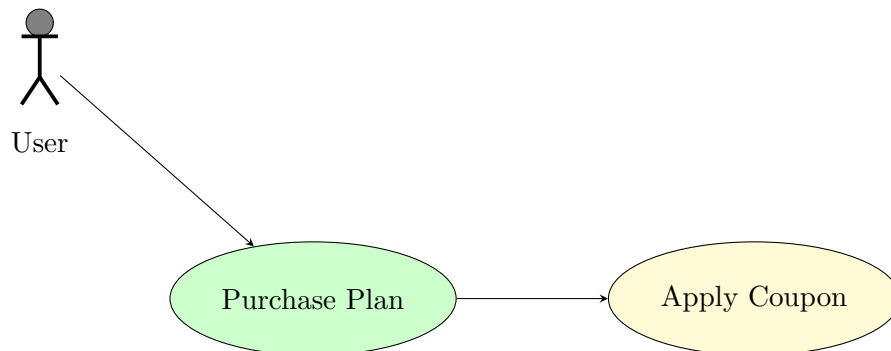


Figure-7.3: Use Case 03

Brief Description: User applies a coupon code during the booking process to avail discounts or special offers.

UC 3: Apply Coupon

Preconditions:

- User has a working internet connection.
- Odyssey Travels application is installed on the device.
- User is logged into the Odyssey Travels account.
- User has selected a travel package and proceeded to the payment section.

Main Success Scenario:

1. System displays the booking details and prompts the user to apply a coupon code.
2. User enters the coupon code in the designated field.
3. System verifies the coupon validity and applies the discount to the total booking amount.
4. System updates the payment summary to reflect the discounted price.
5. User confirms the application of the coupon code.
6. System confirms the coupon code application and adjusts the total payable amount accordingly.

Postconditions:

- System displays the updated booking summary with the discounted price.
- User proceeds to complete the booking with the discounted price.

Alternative Courses:

- 3a. User enters an invalid coupon code or expired coupon.
- 3a.01. System displays an error message indicating the issue with the coupon code.
- 3a.02. User can retry with a different coupon code or proceed without applying any coupon.

Exceptions:

- User may abandon the coupon application process at any step before confirmation.

7.1.4. UC 4: Book Flight and Transportation

Diagram:

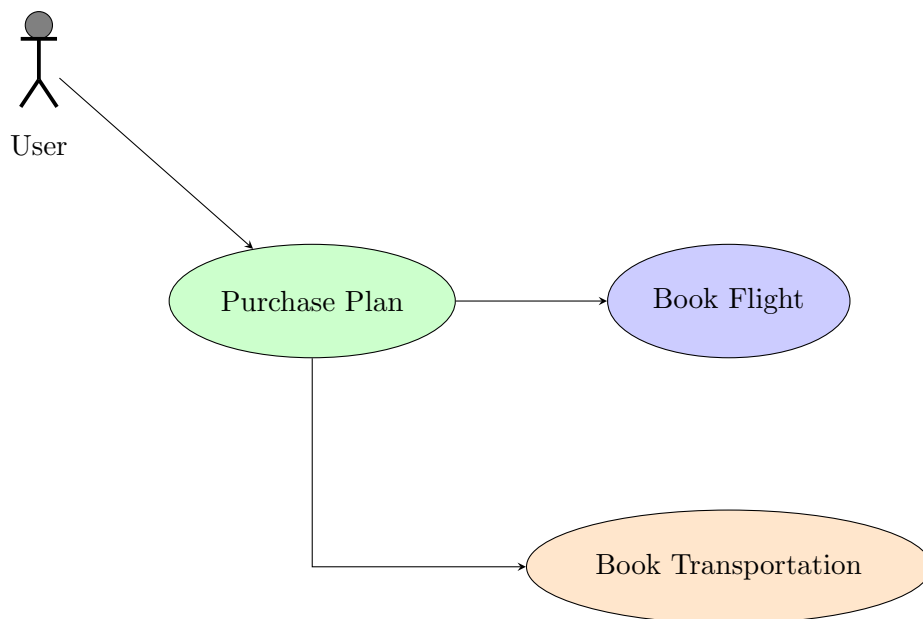


Figure-7.4: Use Case 04

Brief Description: User selects and books flights and transportation services through the Odyssey Travels application.

UC 4: Book Flight and Transportation

Preconditions:

- User has a working internet connection.
- Odyssey Travels application is installed on the device.
- User is logged into the Odyssey Travels account.
- User has navigated to the booking section of the app.

Main Success Scenario:

1. System displays the home page of the Odyssey Travels App.
2. User taps on the 'Book Flight and Transportation' option from the display screen.
3. System displays a list of available flights and transportation options.
4. User selects a flight and transportation option.
5. System prompts user to enter travel details such as departure date, return date, number of passengers, etc.
6. User fills in the required details.
7. System calculates the total fare and displays the payment summary.
8. User confirms the booking details.
9. System processes the payment transaction securely.
10. System confirms the booking with a booking ID and sends a confirmation email or notification to the user.
11. User receives the booking confirmation and can view the booking details in the app.

Postconditions:

- System updates the user's booking history with the new booking details.
- User can access the booked flight and transportation details under their account.

Alternative Courses:

- 5a. User searches for a specific flight or transportation option not listed.
 - 5a.01. System displays a message indicating no results found.
 - 5a.02. User can retry with different search criteria or contact support.

Exceptions:

- User may abandon the booking process at any step before confirming the payment.

7.1.5. UC 5: Book Hotel

Diagram:

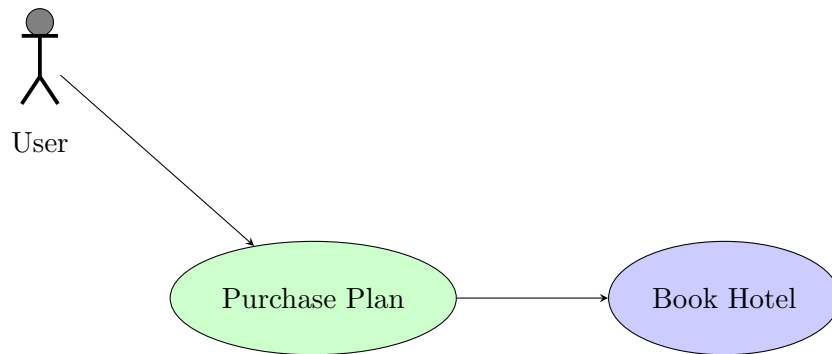


Figure-7.5: Use Case 05

Brief Description: User selects and books accommodations through the Odyssey Travels application.

UC 5: Book Hotel

Preconditions:

- User has a working internet connection.
- Odyssey Travels application is installed on the device.
- User is logged into the Odyssey Travels account.
- User has navigated to the hotel booking section of the app.

Main Success Scenario:

1. System displays the home page of the Odyssey Travels App.
2. User taps on the 'Book Hotel' option from the display screen.
3. System displays a list of available hotels based on the user's search criteria.
4. User selects a hotel from the list.
5. System prompts user to enter details such as check-in date, check-out date, number of rooms, etc.
6. User fills in the required details.
7. System calculates the total cost and displays the payment summary.

8. User confirms the booking details.
9. System processes the payment transaction securely.
10. System confirms the hotel booking with a booking ID and sends a confirmation email or notification to the user.
11. User receives the booking confirmation and can view the booking details in the app.

Postconditions:

- System updates the user's booking history with the new hotel booking details.
- User can access the booked hotel details under their account.

Alternative Courses:

- 5a. User searches for a specific hotel not listed.
 - 5a.01. System displays a message indicating no results found.
 - 5a.02. User can retry with different search criteria or contact support.

Exceptions:

- User may abandon the booking process at any step before confirming the payment.

7.1.6. UC 6: Admin View and Accept/Delete Plans

Diagram:

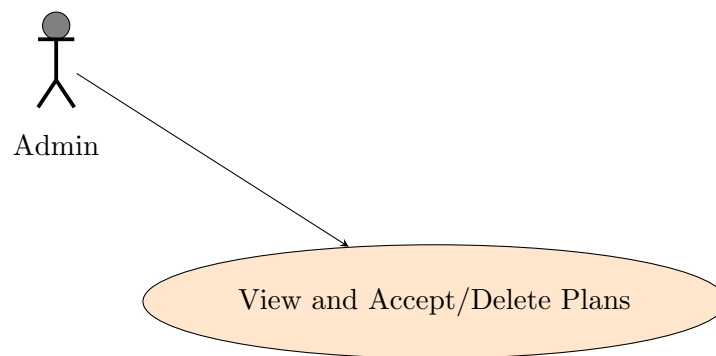


Figure-7.6: Use Case 06

Brief Description: Admin reviews and manages travel plans submitted by users, either accepting or deleting them based on predefined criteria.

Main Success Scenario:

1. Admin logs into the Odyssey Travels admin panel.
2. Navigates to "Manage Travel Plans" section.
3. Views list of submitted plans.
4. Selects a plan for detailed review.
5. Reviews plan details (itinerary, user info, pricing).
6. Evaluates plan against criteria.
7. Accepts plan: updates status, notifies user.
8. Deletes plan: removes from system, notifies user.
9. Records decision and notes.

Postconditions:

- Plan status updated in database.
- User notified of plan status.

Alternative Courses:

- Requests additional info or corrections from user.
- Communicates via internal messaging or email.

Exceptions:

- Technical issues hindering plan access.
- Admin postpones review.

7.1.7. UC 7: Admin View and Edit Booked Hotels of Users**Diagram:**

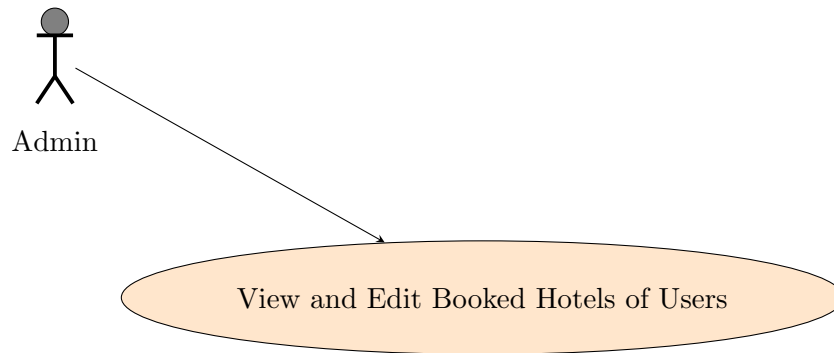


Figure-7.7: Use Case 07

Brief Description: Admin accesses and modifies hotel bookings made by users through the Odyssey Travels platform.

Main Success Scenario:

1. Admin logs into the Odyssey Travels admin dashboard.
2. Navigates to "Manage Booked Hotels" section.
3. Views list of booked hotels with user details.
4. Selects a booked hotel for editing.
5. Reviews booking details (dates, rooms, preferences).
6. Edits booking information as required.
7. Confirms changes and updates the booking.
8. Notifies user of any modifications.

Postconditions:

- Booking details updated in the system.
- User notified of changes made.

Alternative Courses:

- Requests additional information or clarifications from the user.
- Provides support or resolves issues related to the booking.

Exceptions:

- System downtime affecting access to bookings.
- Admin defers editing due to unforeseen circumstances.

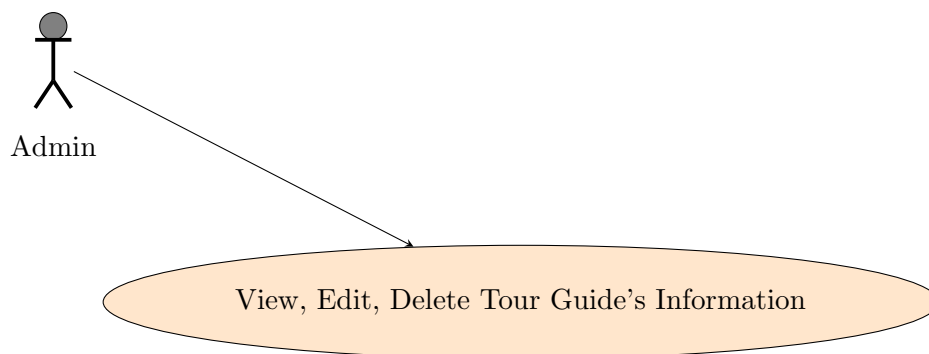
7.1.8. UC 8: Admin View, Edit, Delete Registered Tour Guide's Information**Diagram:**

Figure-7.8: Use Case 08

Brief Description: Admin manages the information of registered tour guides within the Odyssey Travels platform.

Main Success Scenario:

1. Admin logs into the Odyssey Travels admin dashboard.
2. Navigates to "Manage Tour Guides" section.
3. Views list of registered tour guides with their details.
4. Selects a tour guide for viewing or editing.
5. Reviews tour guide's information (contact details, certifications).
6. Edits tour guide's information as required.
7. Confirms changes and updates the tour guide's profile.
8. Deletes tour guide's profile if necessary, after confirmation.

Postconditions:

- Tour guide's information updated or deleted as per admin's actions.
- System reflects changes immediately.

Alternative Courses:

- Admin requests additional documentation or updates from the tour guide.
- Provides feedback or support regarding certification requirements.

Exceptions:

- Technical issues or system errors may delay updates.
- Admin decides to defer deletion due to ongoing bookings or other dependencies.

7.2 Use Case Diagram

A use case diagram is pivotal in illustrating a user's engagements with the system, depicting the correlation between the user and various use cases they engage in. This diagram serves to provide a comprehensive overview of the system and supports the detailed design process. The UML use case diagram comprises a collection of use cases that encompass all conceivable interactions outlined in the system requirements (Section 6). The use case diagram for Odyssey Travels is delineated in Figure-7.1.

UML Diagram :

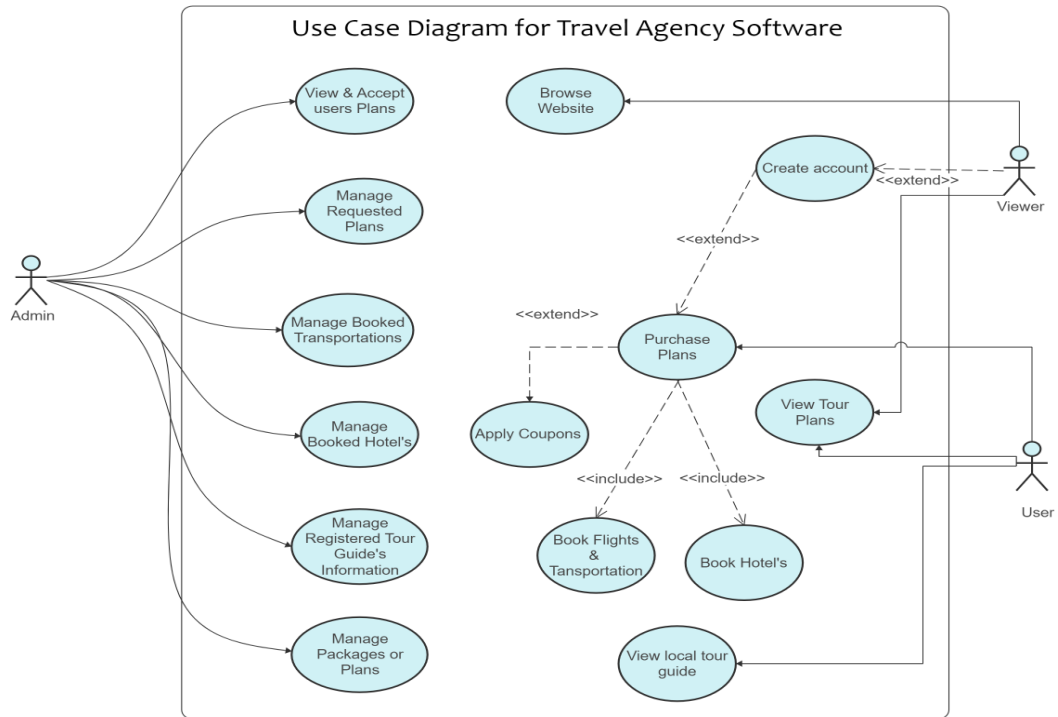


Figure - 7.9 : Use Case Diagram For Travel Agency Software

Although we adhere to the object-oriented approach (OOP) in developing our system, Data Flow Diagrams (DFDs) may occasionally be indispensable for enhanced comprehension. Therefore, we have included the DFDs of the Odyssey Travels project in Appendix-I to provide clarity where necessary.

8 System Evolution

We have formulated foundational assumptions pivotal for launching the Odyssey Travel Agency project. This section delineates these assumptions, addressing potential impacts from advancements in hardware/software, evolving user demands, and other pertinent factors.

Our software development lifecycle extends beyond initial deployment, encompassing ongoing refinement and adaptation. Post-deployment, adjustments may be necessary to address operational issues, accommodate platform updates, and enhance performance and other non-functional attributes. Given that our product is managed internally within our organization, ongoing support and evolution are integral to its lifecycle without necessitating additional contractual agreements.

The evolution process involves essential activities such as change analysis, release planning, system implementation, and customer delivery. We have identified potential post-deployment challenges and their underlying assumptions, summarized in Table-I, to proactively address and mitigate any arising issues.

Table - I: Potential reasons that can cause severe problems during the evolution process of the travel website

SN.	Reasons that can drive us to make urgent changes
1.	User encounters issues with booking confirmations not being received on time. Immediate action is required to fix the notification system to maintain customer trust.
2.	The integration with third-party services (e.g., flight or hotel booking APIs) changes their formats or access methods, causing disruptions in our service. We need to quickly adapt to these changes to keep our booking system functional.
3.	The website's performance degrades significantly under high traffic conditions, which can occur during peak travel seasons. We must improve the server scalability and optimize the codebase to handle increased loads effectively.
4.	New security vulnerabilities are discovered that could potentially expose user data. Immediate security patches and updates are necessary to protect user information and comply with data protection regulations.
5.	A major update is released for the mobile operating system that our mobile app relies on, leading to compatibility issues. We need to update the app promptly to ensure a seamless experience for our users.

Table-II: Maintenance Predictions

SN.	Assumptions	Reengineering Technique
1.	Interface updates will be necessary to accommodate new travel packages and user preferences.	Source code translation
2.	Certain requirements may change due to new regulations or changes in third-party services (e.g., new APIs for hotel bookings).	Program structure improvement
3.	As a new version is rolled out, a significant number of bug and failure reports can be expected due to the high variability in user environments.	Code refactoring
4.	Users may request specific feature enhancements such as multi-language support or personalized itinerary suggestions without needing a complete overhaul of the system.	Program modularization
5.	Integration with new social media platforms for sharing travel plans could require substantial backend modifications.	Integration and migration

Maintenance Predictions for our travel website involve strategic planning to address potential challenges that could impact our system’s functionality and user experience. One critical aspect is **source code translation**, where older programming languages are updated to newer versions or entirely different languages using automated tools, ensuring compatibility and performance enhancements. **Program structure improvement** is another vital strategy, focusing on analyzing and refining the control structures within our software to enhance readability and maintainability, often requiring a balance of automated tools and manual interventions. **Program modularization** plays a key role by organizing related components to streamline functionality and reduce redundancy, fostering easier updates and scalability. These techniques anticipate future needs, such as accommodating new travel packages and user preferences through interface updates. Additionally, they prepare us for regulatory changes or updates in third-party services, ensuring seamless integration and compliance. As we roll out new versions, addressing potential bugs and user environment variability becomes paramount, necessitating thorough **code refactoring** processes to optimize performance and stability. Finally, integrating with emerging social media platforms necessitates backend modifications, highlighting the importance of **integration and migration** strategies in our maintenance predictions. Overall, these proactive measures ensure our travel website remains robust, adaptable, and responsive to evolving customer demands and technological advancements.

9 Appendix-I

DFD (Data Flow Diagram) helps us understand the how the data is flowing across the system and what is the relation between the functions of the system. Level 0 DFD and Level 1 DFD of Efficiency Monitor are shown in figure-9.1 and figure-9.2 respectively.

9.1 Level-0 Data Flow Diagram

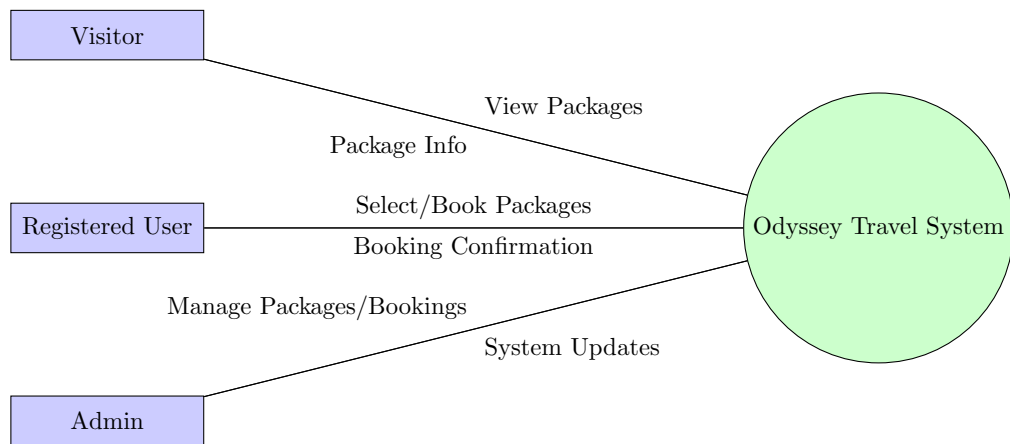


Figure-9.1: Level 0 DFD of Travel Agency Software

9.2 Level-1 Data Flow Diagram

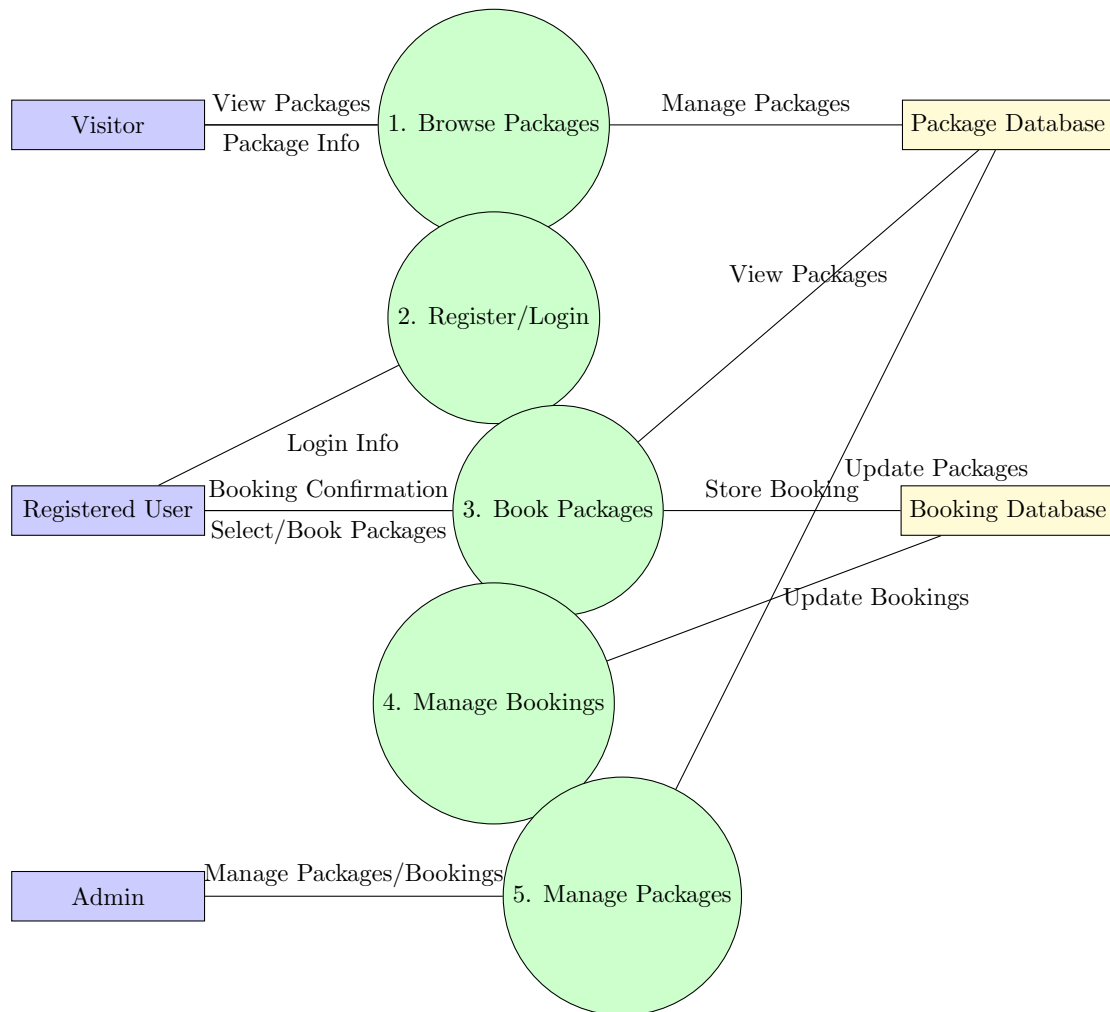


Figure-9.2: Level 1 DFD of Travel Agency Software

Conclusion

The Odyssey Travel Agency project is poised to The Odyssey Travel Agency project aims to simplify and enhance the way people plan and manage their travels. Our platform makes it easy for users to book flights, hotels, and tour guides through a user-friendly interface, ensuring a smooth and enjoyable experience. With real-time updates and secure transactions, users can confidently organize their trips, whether it's for a quick getaway or an extended adventure. We also provide personalized recommendations and support for applying discounts, making travel planning more affordable and efficient. As we continue to develop our platform. Our goal is to make Odyssey Travel Agency a reliable and helpful tool that travelers can depend on for all their journey needs.