**A PROPOSED OFFERING OF A on TRAVEL BOOKING SYSTEM FOR RARE EXPLORER TRAVEL&TOURS AT DATAMEX COLLEGE OF SAINT ADELINE VALENZUELA BRANCH**

A Project Proposal Presented to the

Faculty of Datamex College of Saint Adeline, Inc.

In Partial Fulfillment of the Requirements for the

Degree of Bachelor of Science in Information Technology

By:

Alelojo,Red Hernie

Wong,Chrisdelyn S.

September 2025

**DESIGN DOCUMENT**

**INTRODUCTION**

The purpose of this document is to clearly state the design features and specification of the Rare Explorer Travel & Tour's web-based travel booking system. It is intended as a guide for all parties to the project: developers, designers, testers, and team and client members alike, all of whom must share a common understanding of how the system should function and what it is to achieve. Such an arrangement will bring clarity to and organize the development process by checking against confusion arising during that phase. This in turn guarantees that all requirements are catered for from the onset.

The client is introduced clearly to the views of the finished system and the positive impacts on the business and the customers. Some of the key features covered in the document include the flow of the system, the general look and feel of the interface, and the overall architecture of the platform. This section gives a transparent view with regard to the design decisions involved in the making of this project; thus, it is trust-building that the client's perspective is kept in mind while making decisions for the system.

**Overview**

The travel booking system is under development to modernize and make the system a better way for Rare Explorer Travel and Tours to conduct booking. The company currently adopts a very tedious, time-consuming manual booking processing, making the whole scenario prone to errors and management difficulties with the growing number of customers. With this new online-based system, the whole booking process will become faster and reliable, and much easier for customers and staff.

For customers, the new system is a simple yet user-friendly online platform that allows them to sign up, log in securely and explore all travel packages on offer. Customers will be able to search for trips based on their preferences, access information such as itinerary details, scheduling, charges, and subsequently make a booking in just a few easy steps. After confirming the reservation, there are secure online payment options to which they will receive instant confirmation receipts and electronic itineraries or vouchers. It keeps customers updated with a personal dashboard where they can manage

bookings, view past transactions, and update personal account information any time they want.

There will be an admin dashboard for the company employees that will manage everything when it comes to making the process more efficient. The users would be able to access features such as updating travel packages, checking booking records and payments, and replying to customer inquiries. It automatically generates reports like sales summaries, booking histories, and payment records, which would be useful in making better decisions and tracking business performance. Since it is web-based, the system is accessible through various devices, including desktop, laptop, tablet, or even mobile phone, thus making it the most flexible and easy system to use from anywhere at any time with an internet connection.

**SCOPE OF THE DESIGN**

The travel reservation system is designed to provide a seamless and efficient booking experience for both customers and administrators. It will include core functionality such as account creation, travel booking, secure payment processing, booking confirmations, automated notifications, and report generation. For administrators, the system will offer tools to manage travel packages, monitor payment statuses, oversee customer bookings, and generate analytical reports to support business decisions.

The user interface will prioritize simplicity and clarity, ensuring that customers can easily navigate the platform without technical expertise. The design will be fully responsive, allowing smooth access across desktops, tablets, and smart phones. Accessibility features will also be considered to support users with varying needs.

Security is a central component of the design, with encrypted data transmission, secure login protocols, and role-based access control to protect sensitive customer and financial information. The system will also include audit trails and activity logs to monitor usage and detect anomalies.

Scalability is built into the architecture, allowing the system to handle increased traffic, expanded travel offerings, and additional administrative users as Rare Explorer grows. The back end will support modular upgrades, such as adding loyalty programs, muti-language support, or integration with third-party APIs for flights and accommodations.

Additionally, the system will feature a dashboard for both users and admins, real-time availability tracking, and customization booking work flows to adapt to different types of travel services. These design choices ensure the platform remains flexible, secure, and user-friendly while meeting the operational needs of Rare Explorer Travel and Tours.

**SYSTEM ARCHITECTURE**

The Rarexplorer Travel and Tours booking system is structured under a client-server paradigm in order to provide availability, maintainability, and high availability of the system. The unique combination of cloud services in the deployment grants flexibility for scaling and resource management depending on demand. The system has several communication protocols and interfaces to ensure the smooth interaction of its components, external services, and end-users.

**High-Level Components**

These modules, or system elements, are key components of the entire application. Each module serves a specific purpose and function, and work together to facilitate proper system management.The key modules of the system are:

* **Package Management** – Keeps all travel packages in one place (destinations, schedules, prices, and details).
* **Customer Management** – Stores information about customers like their profiles, preferences, and booking history.
* **Booking Management** – Manages reservations, cancellations, and booking status (pending, confirmed, canceled).
* **Payment Recording** – Records payments, payment types (cash, bank transfer), and gives receipts.
* **Itinerary Generator** – Creates a detailed travel plan with hotel, transport, and activity schedules.
* **Document Management** – Saves digital copies of travel documents (passport, insurance, e-tickets).
* **Reports** – Provides summaries such as daily bookings, finances, and upcoming trips.
* **Staff Access Management** – Controls login access for administrators, agents, and staff.
* **Notification System** – Sends reminders, booking updates, and confirmations through email or app.
* **Reports & Analytic** – Shows data insights like booking trends, payments, and exports reports (PDF/CSV).

**Deployment Architecture**

Deployment architecture describes how the software and hardware environments operate the system. It describes the placement of the application, database, and other components, as well as their inter connectivity. Thus, its an assurance that the system really runs smoothly, securely, and efficiently with users.

* Web-based application built with React / Angular / Flutter Web hosted on Fire-base Hosting.
* Server less back end using Fire base Cloud Functions to handle business logic (e.g., booking, cancellations, payments).
* Data stored in:
* **Cloud Firestore** – for dynamic data (users, bookings, packages
* **Firebase Storage** – for media files and document uploads (e.g., itineraries, customer documents)
* **Fire base** - Authentication manages secure user login and access roles.
* Real-time updates are available through Firestore’s real-time database listeners.
* Accessible via browser or mobile devices, with optional Progressive Web App (PWA) features for offline access.

**Communication and Protocol**

Communication and protocols define how different parts of the system communicate and share data. Protocols are like rules that help information on one end get safely, clearly, and agreed-upon meaning by the other end.

· **Reports -** Generated in PDF and printable formats, ensuring compatibility with existing office documentation practices.

· **Offline Functionality -** The system operates entirely offline, meaning no internet protocols are required for daily operations.

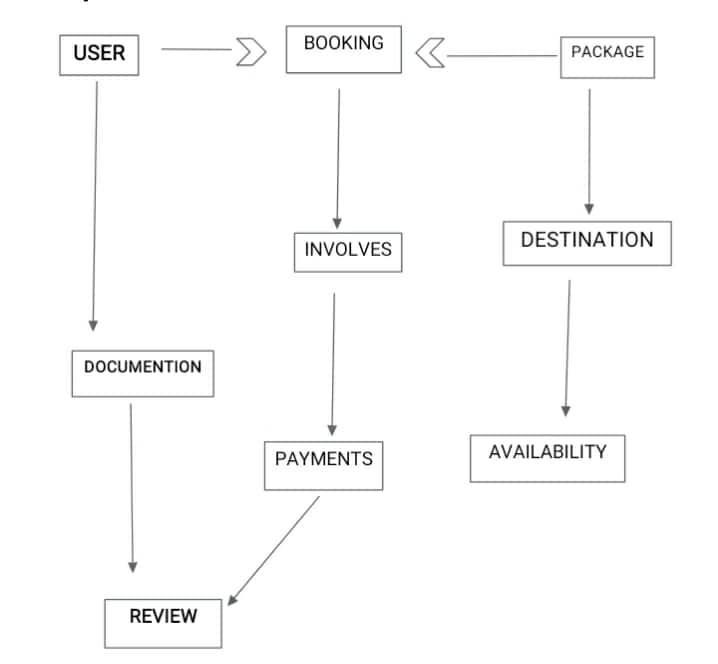
**DATABASE DESIGN**

Database design shows how information is arranged and stored in the system. It organizes data into tables so it is easy to save, find, and manage.

The Entity Relationship Diagram (ERD) for the Travel Booking System shows how data are stored and connected inside the system. The main parts include Users, Packages, Reservations, Payments, and Notifications. Each part keeps information about customers, travel options, bookings, and payments. The ERD also shows how these parts are linked .

Entity Relationship Diagram (ERD) consist of these core entities:

* **User**: Represents the traveler or customer using the system
* **Booking**: Represents the booking made by the user
* **Package**: Represents a travel package available for booking
* **Destination**: Represents the travel destination associated with a package
* **Availability**: Represents the availability of a package for booking.
* **Payment**: Represents payment transactions made by the user for booking.
* **Documentation**: Represents the documents associated with the booking.
* **Review**: Represents the reviews given by users after completing the booking



*“Figure 1. Entity Relationship Diagram (ERD) of the Travel Booking System”*

**Description of database tables, fields, and relationships**

* **USER -** The customer or client who is initiating the travel booking. They provide necessary information and documents.
* **BOOKING -** The core transaction where the user reserves a travel package. It links the user's choice with payment and package details.
* **PACKAGE -** A pre-defined set of travel services (e.g., flights, hotel, tours) that the user can choose from. It is directly linked to a specific destination.
* **DESTINATION -** The geographical location or venue included in the travel package.
* **AVAILABILITY -** Represents the status of the destination, checking if slots, seats, or rooms are available for the requested dates. This is a critical check before a booking can be confirmed.
* **INVOLVES -** This is an action or relationship showing that a booking process includes or leads to the payment step.
* **PAYMENTS -** Refers to the financial transaction stage where the user pays for the selected package. This is a required part of completing the booking.
* **DOCUMENTATION -** The necessary paperwork and information provided by the user, such as passports, visas, or identification, which are required for the travel arrangements.
* **REVIEW -** The final stage of the process where the submitted documentation and the confirmed payment are verified to finalize and approve the user's booking.

**Data Normalization**

The RareExplorer Travel Tours data design consists of the main four tables: Customers, Packages, Bookings, and Payments. Each data point is indexed and stored in its most reduced form, making the system easy to update and maintain. The separation of logically related pieces of information into individual tables instead of lumping them together in one table gives the travel booking system higher performance, accurate results, and better organized structure.

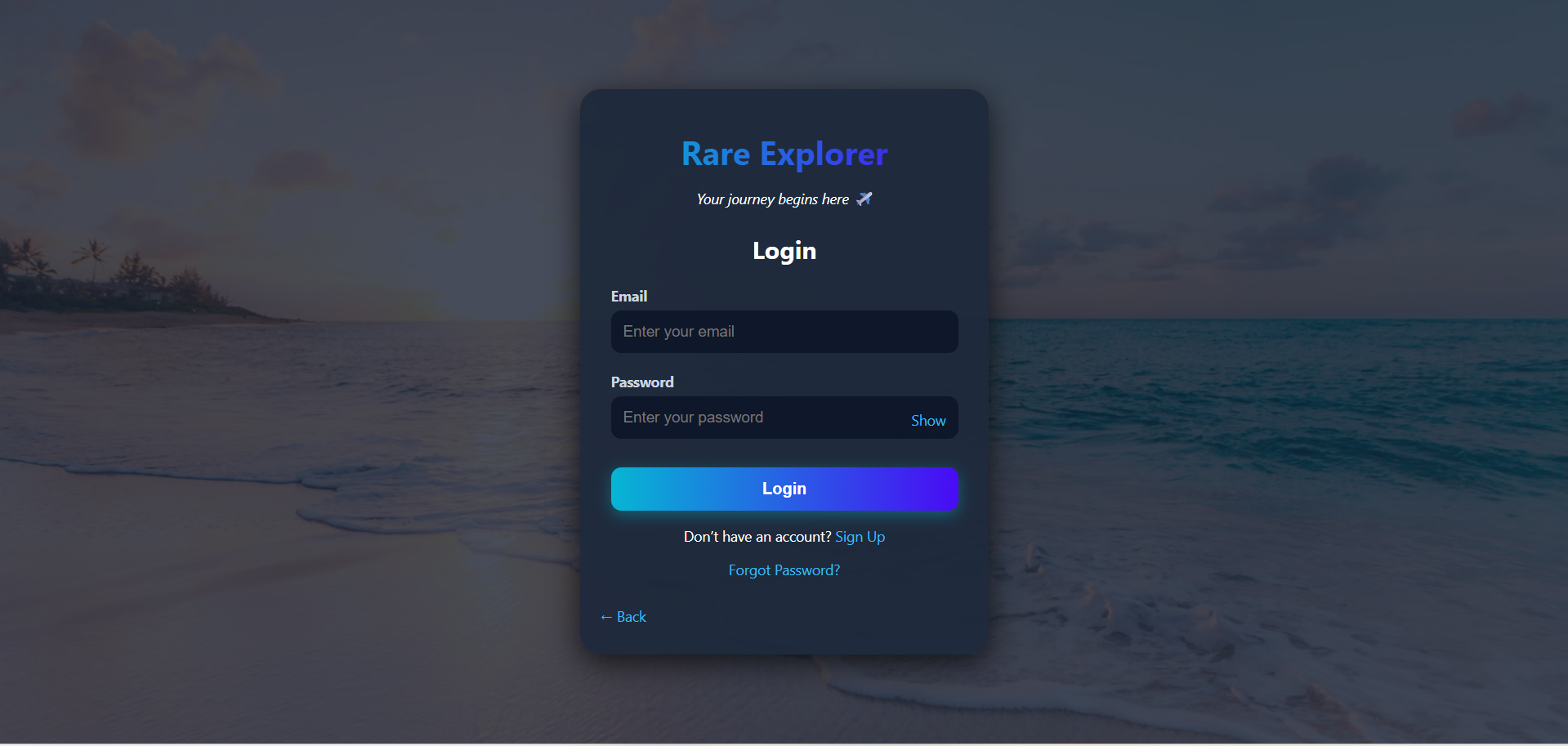
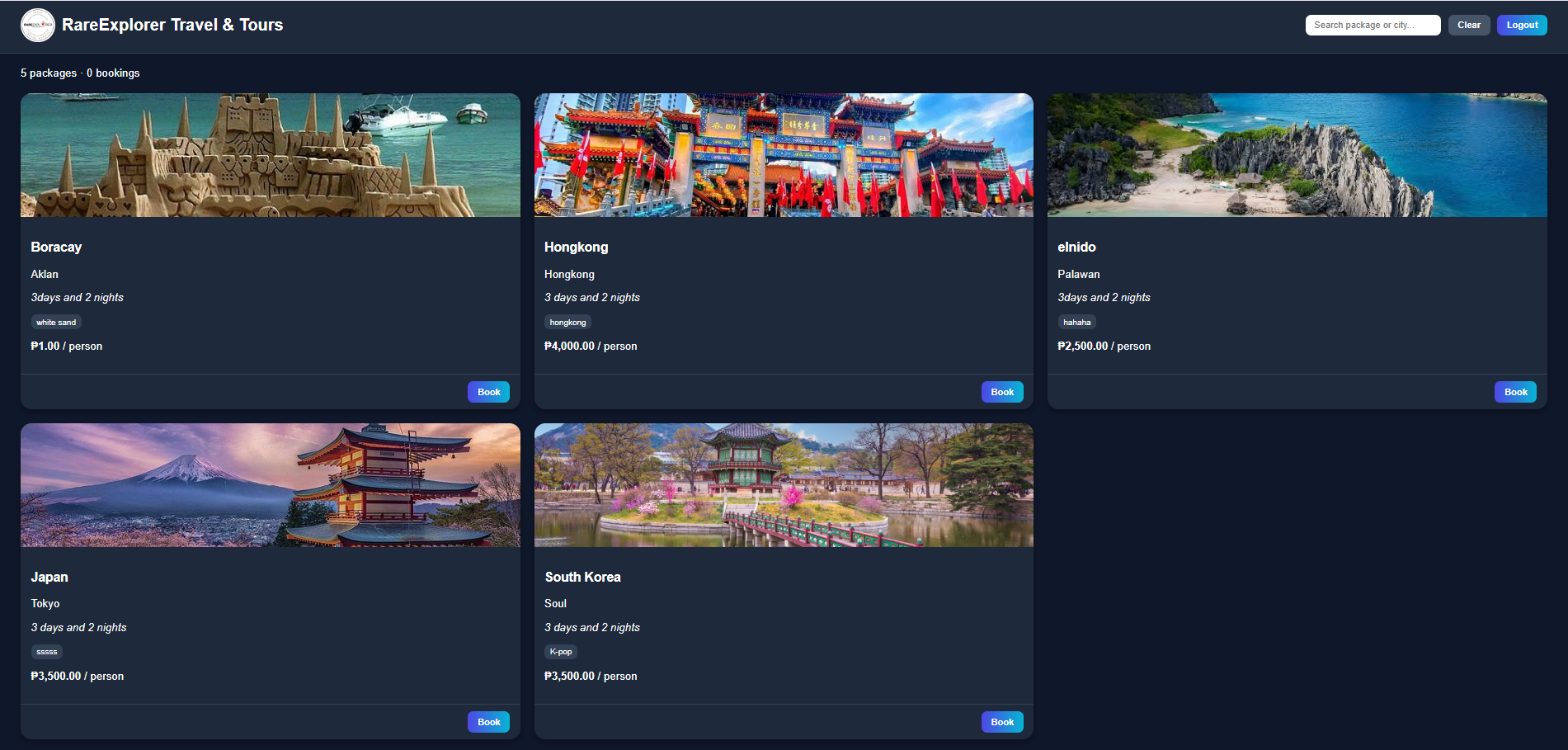
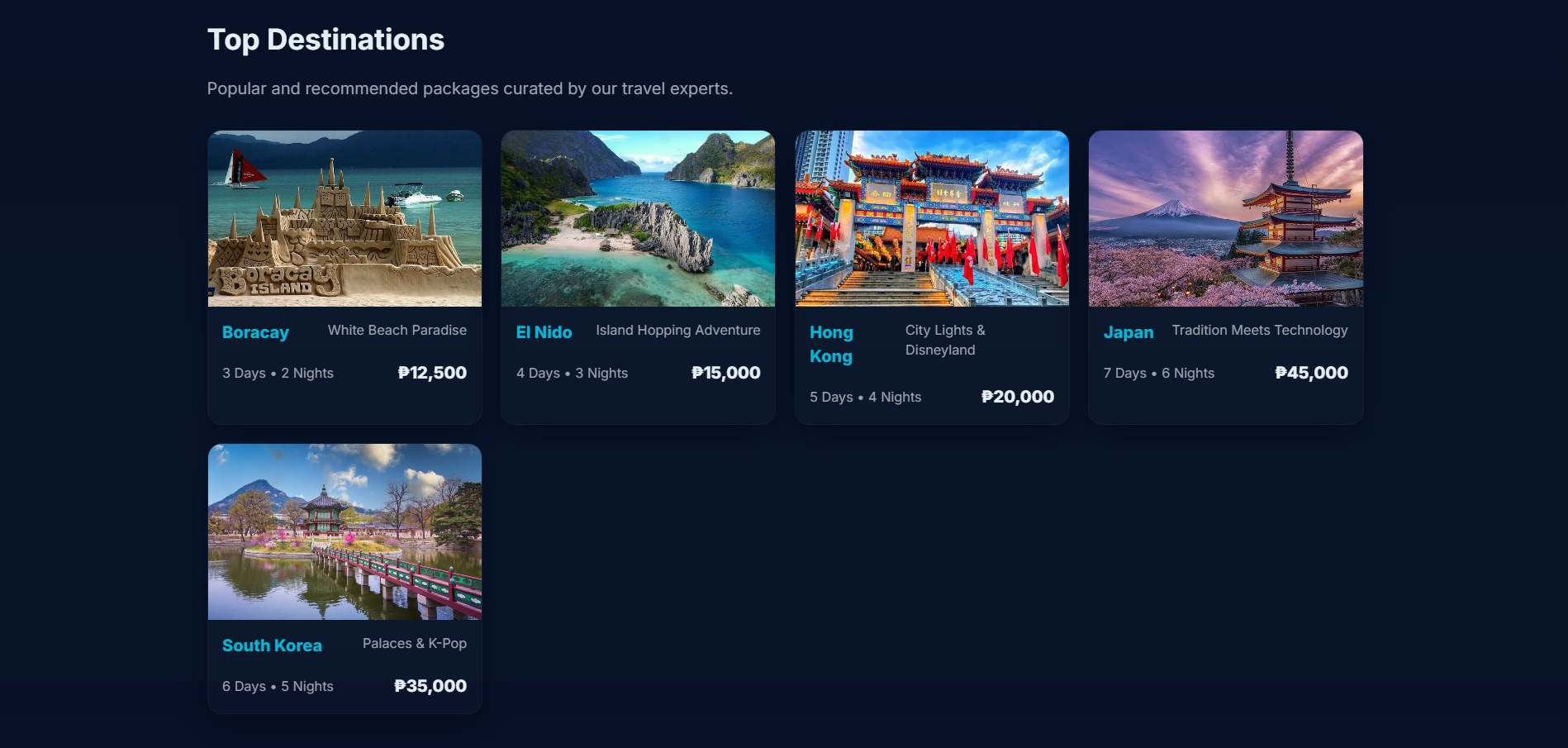
**USER INTERFACE DESIGN**

Figure 2. *User Interface Design of the Travel Booking System (Sign Up Page For User)*

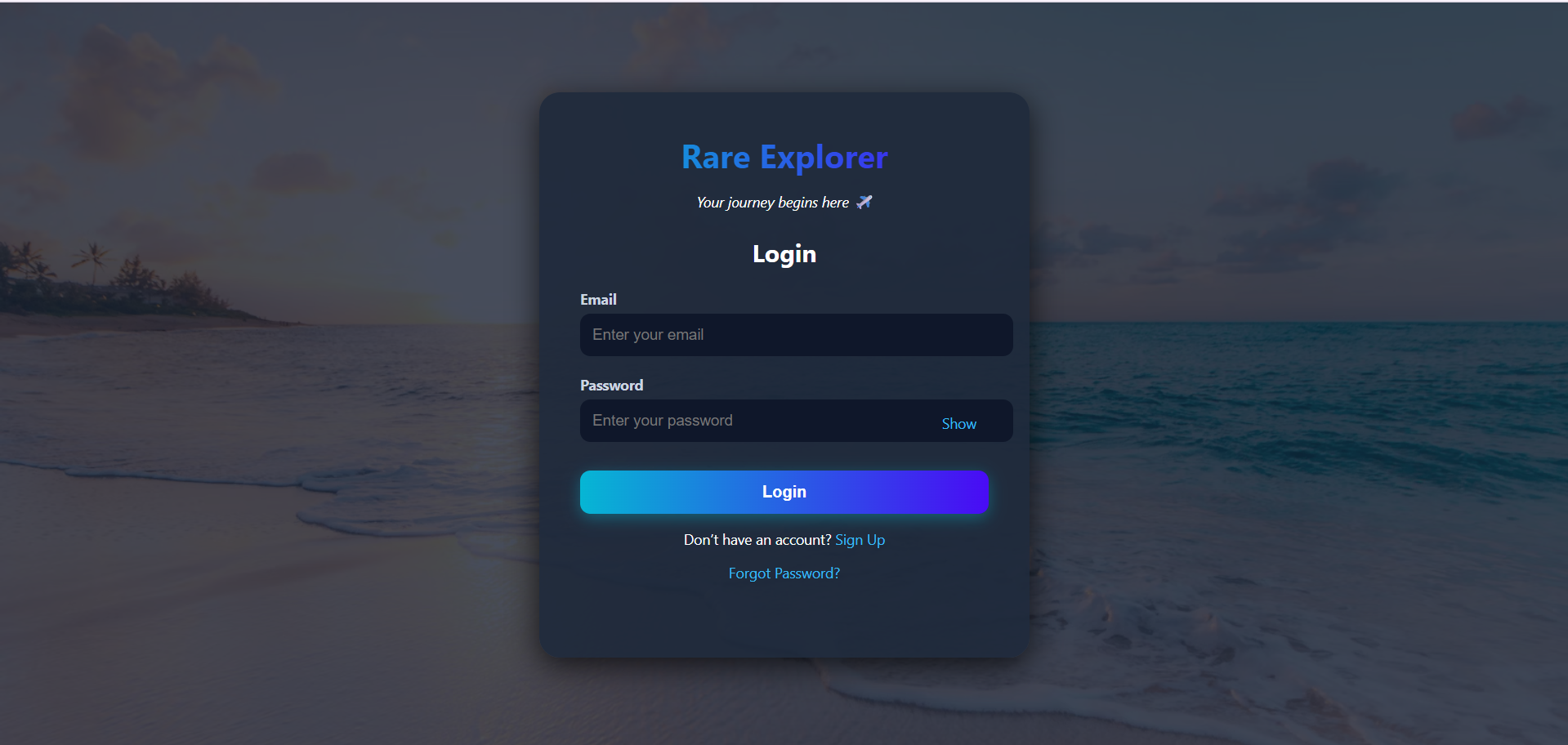
Figure 3. *User Interface Design of the Travel Booking System (User Page)*

**

*Figure 4. User Interface Design of the Travel Booking System (About Page for User)*



*Figure 5. User Interface Design For Travel Booking System(Top Destinations)*



*Figure 6. User Interface Design of the Travel Booking System (Admin Sign Up Page)*



*Figure 7. User Interface Design of the Travel Booking System (Admin Page)*

**COMPONENT DESIGN**

Data normalization is the process of arranging data in such a way that it is clean and clear in the database. It eliminates redundant data, stores data with minimum redundancy, and separates related data into other tables such as Customers, Packages, Bookings, and Payments. This gives the system ease of updating, accuracy, and neatness.

* **User Management Module** – Lets users sign up, log in, and edit their profile. It also connects users with their bookings.
* **Tour Management Module** – Handles tour packages, schedules, and availability. It checks if tours are available when users book.
* **Booking Management Module** – Manages bookings (add, change, or cancel). It works with the Tour and Payment modules to confirm reservations.
* **Payment Module** – Processes payments through services like Gcash or bank transfer. It confirms bookings once payment is successful.
* **Customer Support Module** – Helps customers through chat or support tickets. It uses booking details to answer questions and fix problems.
* **Admin Module** – Allows admins to manage tours, bookings, payments, and users. It connects with all modules and can create reports.

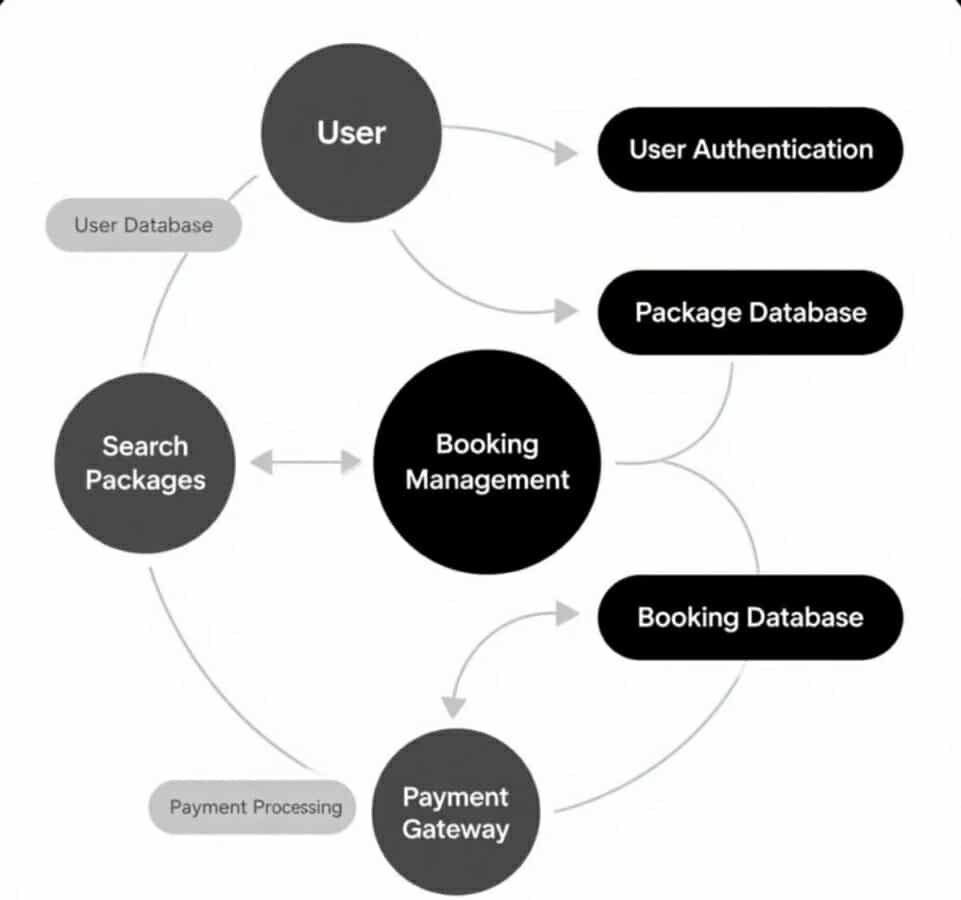
**Dependencies**

Dependencies are the essentials for the operation of a system. These may be software, hardware, or services which the system depends on. For the travel booking system, a database to save records a payment service for payment processing an internet connection for users to gain access to the system; these are needed to support the system. These basic requirements, the system will not function properly.

1. **User Management Module** – Manages authentication and authorization, ensuring only valid users can access the system.
2. **Scheduling Module** – Provides schedule and availability data needed for reservations.
3. **Reservation Management Module** – Handles bookings while coordinating with payment and notification modules.
4. **Payment Processing Module** – Processes transactions through external gateways and updates payment status internally.
5. **Notification Module** – Sends confirmations, alerts, and updates to users via messaging services.

**DATAFLOW DIAGRAM**

A Data Flow Diagram shows how data moves inside the system. It explains where the data comes from, how it is used, and where it goes. It also shows how the system saves and gets data to keep everything running smoothly.

******

*Figure 8. Data Flow Diagram of the Travel Booking System*

**SECURITY DESIGN**

The booking system of the RareExplorer Travel and Tours uses Fire base, which contains more security aspects and the customer data may not be accessed by pretending a real owner. It uses Fire base Authentication for secure log in and sign up, with a range of supported authentication flows including email-password, phone authorization and social login, and role-based feature access control meaning only signed users can use certain features. To manage data protection, Fire base and Real time Database deploy security rules that limit who can read and write data. Sensitive data or transactions are protected with Firebase out-of-the-box encryption, as well.

**PERFORMANCE DESIGN**

The RareExplorer Travel and Tours booking system utilizes the Fire base platform to ensure that the booking system works quickly, smoothly, and reliably for every customer. Fire base Hosting provides pages and content around the world to speed up loading time, so that the customer doesn't need to wait for the system to load regardless of location. The Firestone database is properly defined by way of rules and searches are easy for the most efficient discovery and booking experience with the inherent in memory storage of Fire base that will sit idle similar requests to avoid duplicates or concurrent requests.

Google Cloud helps facilitate user traffic and scales automatically once too many users are booking at once. Each platform is tested by Fire base Test Lab to check the booking system for performance on various devices and internet speeds. Fire base Performance Monitoring helps measure speed, lag time, and detected issues in real time. Using each of these continual indicators will allow the system to provide the customer with a fast, responsive, and reliable booking experience.

**ERROR AND HANDLING**

The RareExplorer Travel and Tours booking system uses Fire base for error management and to log system activity in a systematic way. Centralized error handling manages all errors consistently such as incorrect login details, double bookings, or network errors while not preventing minor issues from being retried or resolved in an alternative way. Fire base logs all activity and any errors to Firestore , structured to include time stamps, severity levels, module names and error messages. All logs are aggregated centrally to allow administrators to monitor system performance and identify potential unintended activity.

Critical errors are monitored with real time data from Fire base allowing issues to be detected and resolved promptly, contributing to a smooth, safe and trustworthy booking transaction for customers.To increase reliability, Disturbances (critical errors) are monitored on a real-time basis, which provides a way for any problems to be identified and resolved quickly. Fire base allows alerts to be sent to administrators so they know if serious problems arise right away. Thus, customers are assured a consistent, safe, and reliable booking experience.

**THIRD PARTY INTEGRATION**

The booking system for RareExplorer Travel and Tours, which is hosted on the Google Cloud platform, leverages other services to optimize the system for users and improve customer experience. For example, payment services such as Gcash provide safe mechanisms for online payments; email and text services such as G mail API allow the system to send booking confirmations, booking reminders and updates; and Google Calendar can be connected to have trips or tours automatically populate the calendars of customers and staff.

These services are used by the booking system by sharing information using simple data exchange formats such as JSON or XML, which allow for messages to be formatted uniformly and send as safe messages to and from respective services. The use of these services assists in ensuring reliability, convenience, and user-friendliness of the booking system.

**DEPLOYMENT PLAN**

**Overview of the Deployment Process**

The web-based travel booking system (which is in phase III of development) will begin deployment on the production environment after assessing the hotel reservations application. The deployment process consists of configuring the production environment, installing the software components, configuring the system, and post deployment testing of the application on the production environment and for end users in the hosted environment.

The goal is to help ensure a smooth transition from development to production, with minimal downtime and disruption to users. The deployment team will follow a prescribed series of events, including configuring the environment, installing the software, migrating any necessarily service data, and validating the system is working as expected prior to going live.

**Hardware and Software Requirements For Deployment**

* You’ll need a good server computer to host the system with enough memory and storage.
* The server should run a stable operating system like Linux or Windows Server.
* Software needed includes a database to store bookings, a web server to run the site, and the programming environment the system is built on.
* Users will just need a modern web browser to use the system from their computers or phones.

**Configuration Management And Version Control**

* Configuration files for the system will be managed centrally and version-controlled using industry-standard tools such as Git. This includes environment-specific settings (database connections, API keys, server URLs).
* Documentation will include step-by-step deployment guides, rollback procedures, and contact points for support.

**MAINTENANCE AND SUPPORT**

The RareExplorer Travel and Tours booking system will need ongoing management and an obvious ongoing support process to keep it active over time. This maintenance and support section defines the principles for keeping the booking system doing the right thing over time, the method for applying software updates, patches and bug fixes, and the escalation process to help identify, manage, and leave notes regarding issues that cannot be solved immediately. All of these items help secure the system reliability and ensure that the planning and booking system stays as user friendly as possible for the customer and staff.

**System Maintenance and Support Guidelines**

* **Regular Backups -** To protect reservation and customer data, daily incremental backups and full weekly database backups should be performed. Monitoring of server performance, storage, and network usage is essential to ensure the system runs smoothly without interruptions.
* **Security Management -** Regular checks for vulnerabilities, implementation of protective measures, and audits of user permissions should be done to keep the system secure and follow company security policies
* **Log Management -** System logs, error logs, and user access logs should be kept to help with auditing, troubleshooting, and maintaining records of system activity.
* **User Support -** Support channels such as help desks, ticketing systems, and live chats should be available to assist users with technical or booking-related issues.
* **Software Update, Patch, and Bug Fix Procedures**
* **Updates -** System updates should be carried out during off-peak hours to avoid disrupting users.
* **Testing -** All updates, patches, and bug fixes must first be tested in a safe environment to ensure they work properly and do not cause issues.
* **Deployment -** Once validated, changes are applied to the live system using proper version control and rollback procedures in case problems occur.
* **Notification -** Users should be informed ahead of scheduled maintenance, downtime, or significant system changes.
* **Documentation -** All updates, patches, and fixes should be recorded for tracking and future reference.

**REVISION HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description of Changes** | **Authors()** |
| Aug 28, 2025 | **1.0** | First draft of the requirements prepared. | Project Team |
| Sept. 5, 2025 | **1.1** | Added more details to requirements and the format. | Project Team |
| Sept.08,2025 | **1.2** | Fix the format and add a info | Project Team |

*Table 1 : Revision Table*

**APPENDIX**

*Wiegers,Karl(2023,August 18) SRS Document for Online Travel Booking System: Complete Specifications. Dr. Vishwanath Karad MIT World Peace University.*

*<https://www.studocu.com/in/document/dr-vishwanath-karad-mit-world-peace-university/computer-science-engineering/srs/78628350>*

*GeeksforGeeks. (2020, June 13). System design of airline management system.*

*<https://www.geeksforgeeks.org/system-design/system-design-of-airline-management-system/>*

*GeeksforGeeks. (2020, July 23). How to design a database for booking and reservation systems.*

*<https://www.geeksforgeeks.org/dbms/how-to-design-a-database-for-booking-and-reservation-systems/>*