**Introduction**

In the modern world, the demand for seamless and efficient travel booking platforms has grown exponentially. With the rise of digital transformation, travelers expect a comprehensive solution that caters to all their travel needs, from booking flights and hotels to renting cars and planning itineraries. This project aims to develop a robust travel booking platform that integrates various travel services into a single, user-friendly interface.

The travel booking platform is designed to simplify the process of organizing trips by providing a centralized system where customers can manage all aspects of their travel plans. The key features of the platform include customer management, booking management, trip planning, payment processing, hotel reservations, flight bookings, car rentals, reviews, destination information, and itinerary management.

**Abstract**

* This project report details the development of a comprehensive Travel Booking Platform designed to facilitate the seamless management of travel-related services for users. The platform integrates several key modules, including customer management, booking management, trip planning, payment processing, hotel reservations, flight bookings, car rentals, review submissions, destination information, and itinerary management.
* The database design, which is central to the platform, is structured to efficiently store and retrieve data across these interconnected modules, ensuring a cohesive user experience. The functional requirements outline the essential features that the platform must support, while the non-functional requirements emphasize performance, security, usability, and reliability, ensuring the platform meets high standards of quality and user satisfaction.
* The implementation plan is broken down into phases, starting from planning and requirement analysis, through design, development, testing, deployment, and ongoing maintenance. Each phase is carefully outlined to ensure a systematic approach to building the platform.
* This report aims to provide a clear roadmap for developing a robust, secure, and user-friendly Travel Booking Platform that enhances the travel booking experience for users by integrating all necessary travel services into a single, cohesive system.

**Functional Requirements**

Customer Management:

Registration: Allow customers to register with their details.

Profile Management: Enable customers to update their profiles.

Booking Management:

Create Booking: Allow customers to book trips, flights, hotels, and car rentals.

View Booking: Enable customers to view their bookings.

Cancel Booking: Allow customers to cancel bookings.

Trip Management:

Create Trip: Enable the creation of trips with start and end dates, and destination.

View Trip: Allow customers to view trip details.

Payment Processing:

Make Payment: Allow customers to make payments for their bookings.

View Payment History: Enable customers to view past payments.

Hotel Management:

Book Hotel: Allow customers to book hotels as part of their trip.

View Hotel Details: Enable customers to view details of booked hotels.

Flight Management:

Book Flight: Allow customers to book flights.

View Flight Details: Enable customers to view flight details.

Car Rental Management:

Rent Car: Allow customers to rent cars for their trips.

View Rental Details: Enable customers to view details of rented cars.

Review Management:

Submit Review: Allow customers to submit reviews for their trips.

View Reviews: Enable customers to view reviews of other customers.

Destination Information:

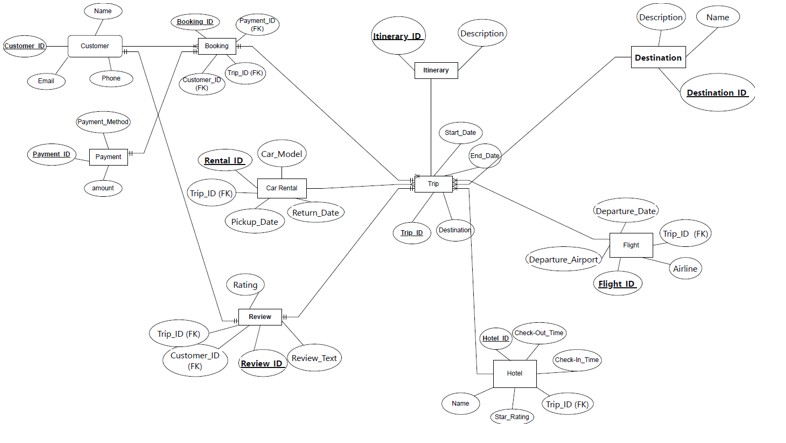
View Destinations: Allow customers to view information about various destinations.

Itinerary Management:

Create Itinerary: Enable the creation of itineraries for trips.

View Itinerary: Allow customers to view their trip itineraries.

**ER DIAGRAM:**



**MY SQL QUERIES :**

* CREATING THE DATABASE:

create database travel;

* USING THE DATABASE:

use travel;

* QUEREY TO SHOW THE TABLE:

Select\*from tablename;

* CREATE TABLE FOR CUSTOMER:

CREATE TABLE Customer (

customer\_id INT PRIMARY KEY,

c\_name VARCHAR(255),

email VARCHAR(255),

phone VARCHAR(10)

);

* CREATE TABLE DESTINATION:

CREATE TABLE Destination (

Destination\_ID INT PRIMARY KEY,

Name VARCHAR(100),

Description VARCHAR(255)

);

* CREATE TABLE FOR TRIP:

CREATE TABLE Trip (

Trip\_ID INT PRIMARY KEY,

Itinerary\_ID INT,

Start\_Date DATE,

End\_Date DATE,

Destination\_ID INT,

FOREIGN KEY (Itinerary\_ID) REFERENCES Itinerary(Itinerary\_ID),

FOREIGN KEY (Destination\_ID) REFERENCES Destination(Destination\_ID)

);

* CREATE TABLE FOR BOOKING:

CREATE TABLE Booking (

Booking\_ID INT PRIMARY KEY,

Customer\_ID INT,

Payment\_ID INT,

Trip\_ID INT,

FOREIGN KEY (Customer\_ID) REFERENCES Customer(Customer\_ID),

FOREIGN KEY (Payment\_ID) REFERENCES Payment(Payment\_ID),

FOREIGN KEY (Trip\_ID) REFERENCES Trip(Trip\_ID)

);

* CREATE TABLE FOR FLIGHT:

CREATE TABLE Flight (

Flight\_ID INT PRIMARY KEY,

Trip\_ID INT,

Departure\_Date DATE,

Departure\_Airport VARCHAR(100),

Airline VARCHAR(100),

FOREIGN KEY (Trip\_ID) REFERENCES Trip(Trip\_ID)

);

* CREATE TABLE FOR CAR RENTAL:

CREATE TABLE Car\_Rental (

Rental\_ID INT PRIMARY KEY,

Trip\_ID INT,

Car\_Model VARCHAR(100),

Pickup\_Date DATE,

Return\_Date DATE,

FOREIGN KEY (Trip\_ID) REFERENCES Trip(Trip\_ID)

);

* CREATE TABLE FOR REVIEW:

CREATE TABLE Review (

Review\_ID INT PRIMARY KEY,

Trip\_ID INT,

Customer\_ID INT,

Rating INT,

Review\_Text TEXT,

FOREIGN KEY (Trip\_ID) REFERENCES Trip(Trip\_ID),

FOREIGN KEY (Customer\_ID) REFERENCECustomer(Customer\_ID)

);

* CREATE TABLE FOR HOTEL:

CREATE TABLE Hotel (

Hotel\_ID INT PRIMARY KEY,

Trip\_ID INT,

h\_Name VARCHAR(100),

h\_Check\_In\_Time TIME,

h\_Check\_Out\_Time TIME,

Star\_Rating INT,

FOREIGN KEY (Trip\_ID) REFERENCES Trip(Trip\_ID)

);

* CREATE TABLE FOR PAYMENT:

CREATE TABLE Payment (

Payment\_ID INT PRIMARY KEY,

Payment\_Method VARCHAR(50),

Amount DECIMAL(10, 2)

);

* CREATE TABLE FOR ITINERARY:

CREATE TABLE Itinerary (

Itinerary\_ID INT PRIMARY KEY,

Description VARCHAR(255)

);

**CLASS / UML DIAGRAM:**

|  |
| --- |
| CUSTOMER |
| Attributes  -Customer\_ID:int  -Name:int  -Email:string  -Phone:long |
| Operations()  +addCustomerDetails():void  +updateCustomerDetails():void |

|  |
| --- |
| BOOKING |
| Attributes  -Booking\_ID:int  -Trip\_ID:int  -Payment\_ID:long |
| Operations()  +bookingStatus():void  +payment\_status():void |

|  |
| --- |
| TRIP |
| Attributes  -Start\_Date:string  -End\_Date:string  -Trip\_ID:int |
| Operations()  +updatestartDate():void  +updateendDate():void |

|  |
| --- |
| PAYMENT |
| Attributes  -Payment\_ID:long  -Payment\_Method:string  -Amount:int |
| Operations()  +processPayment():void  +updatePaymentDetails():void |

|  |
| --- |
| HOTEL |
| Attributes  -Hotel\_id:int  -name:string  -star\_rating:int  -checkin\_time:string  -checkout\_time:string  -trip\_id:int |
| Operators() |
| +addCheckintime():void  +addCheckouttime():void |

|  |
| --- |
| Flight |
| Attributes  -detature\_date:string  -depature\_airport:string  -flight\_id:long  -airline:string  -trip\_id:int |
| Operations()  +addDepatureDate():void  +addDepatureAirport():void |

|  |
| --- |
| CAR RENTAL |
| Attributes  -rental\_id:int  -car\_model:int  -pickup\_date:string  -return\_date:string  -Trip\_id:int |
| Operations()  +addPickupDate():void  +addReturnDate():void |

**CHALLENGES LIST:**

Teamwork Issues: Effective teamwork is essential for project success, but challenges such as conflicts, miscommunication, and lack of collaboration can hinder team performance. Building strong team dynamics, fostering trust and respect among team members, and promoting clear roles and responsibilities can help overcome teamwork issues and promote collaboration.

Lack of Information Access: Ensuring that all team members have access to relevant project information, data, and resources is essential for effective communication. Challenges may arise if certain team members are unable to access critical information due to technical limitations or restricted permissions.

**Scalability: Ensuring that the inventory management system can handle an increasing volume of products and transactions as the business grows.**