TRAVEL SITE DATABASE DESIGN

&

ACTION PLANS

---

1. User Management (**Authentication & Authorization**)

- **Users**
  - `UserID` INT AUTO_INCREMENT PRIMARY KEY
  - `Username` VARCHAR(50) UNIQUE NOT NULL
  - `PasswordHash` VARCHAR(255) NOT NULL
  - `Email` VARCHAR(100) UNIQUE NOT NULL
  - `FirstName` VARCHAR(50) NOT NULL
  - `LastName` VARCHAR(50) NOT NULL
  - `PhoneNumber` VARCHAR(15)
  - `RoleID` INT NOT NULL (Foreign Key referencing Roles table)
  - `CreatedAt` TIMESTAMP DEFAULT CURRENT_TIMESTAMP
- **Roles**
  - `RoleID` INT AUTO_INCREMENT PRIMARY KEY
  - `RoleName` VARCHAR(50) UNIQUE NOT NULL

2. Master Data Management

- **CountryMaster**
  - `CountryID` INT AUTO_INCREMENT PRIMARY KEY
  - `CountryName` VARCHAR(100) UNIQUE NOT NULL
  - `CountryCode` VARCHAR(5) UNIQUE NOT NULL
- **StateMaster**
  - `StateID` INT AUTO_INCREMENT PRIMARY KEY
  - `StateName` VARCHAR(100) NOT NULL
  - `CountryID` INT NOT NULL (Foreign Key referencing CountryMaster table)
- **CityMaster**
  - `CityID` INT AUTO_INCREMENT PRIMARY KEY
  - `CityName` VARCHAR(100) NOT NULL
  - `StateID` INT NOT NULL (Foreign Key referencing StateMaster table)
- **CategoryMaster**
  - `CategoryID` INT AUTO_INCREMENT PRIMARY KEY
  - `CategoryName` VARCHAR(100) UNIQUE NOT NULL
- **TagMaster**
  - `TagID` INT AUTO_INCREMENT PRIMARY KEY
  - `TagName` VARCHAR(100) UNIQUE NOT NULL
- **PackageTypeMaster**
  - `PackageTypeID` INT AUTO_INCREMENT PRIMARY KEY
  - `PackageTypeName` VARCHAR(100) UNIQUE NOT NULL
- **TourIncludesMaster**
  - `TourIncludeID` INT AUTO_INCREMENT PRIMARY KEY

o `IncludeDescription` TEXT NOT NULL

3. Package Management

- **Packages**
  - o `PackageID` INT AUTO_INCREMENT PRIMARY KEY
  - o `PackageName` VARCHAR(255) NOT NULL
  - o `Description` TEXT
  - o `PackageListImageURL` VARCHAR(255)
  - o `PackageDetailsImageURL` VARCHAR(255)
  - o `DurationDays` INT NOT NULL
  - o `DurationNights` INT NOT NULL
  - o `Price` DECIMAL(10, 2) NOT NULL
  - o `PackageTypeID` INT NOT NULL (Foreign Key referencing PackageTypeMaster table)
  - o `CategoryID` INT NOT NULL (Foreign Key referencing CategoryMaster table)
  - o `IsActive` BOOLEAN DEFAULT TRUE
  - o `CreatedAt` TIMESTAMP DEFAULT CURRENT_TIMESTAMP
- **PackageTourDetails**
  - o `PackageTourDetailID` INT AUTO_INCREMENT PRIMARY KEY
  - o `PackageID` INT NOT NULL (Foreign Key referencing Packages table)
  - o `CountryID` INT NOT NULL (Foreign Key referencing CountryMaster table)
  - o `StateID` INT NOT NULL (Foreign Key referencing StateMaster table)
  - o `CityID` INT NOT NULL (Foreign Key referencing CityMaster table)
- **PackageTags**
  - o `PackageTagID` INT AUTO_INCREMENT PRIMARY KEY
  - o `PackageID` INT NOT NULL (Foreign Key referencing Packages table)
  - o `TagID` INT NOT NULL (Foreign Key referencing TagMaster table)
- **PackageTourIncludes**
  - o `PackageTourIncludeID` INT AUTO_INCREMENT PRIMARY KEY
  - o `PackageID` INT NOT NULL (Foreign Key referencing Packages table)
  - o `TourIncludeID` INT NOT NULL (Foreign Key referencing TourIncludesMaster table)

4. Ratings and Reviews

- **TotalRatings**
  - o `RatingID` INT AUTO_INCREMENT PRIMARY KEY
  - o `PackageID` INT NOT NULL (Foreign Key referencing Packages table)
  - o `Stars` INT CHECK (Stars BETWEEN 1 AND 5) NOT NULL
  - o `NumberOfRatings` INT
- **Ratings**
  - o `RatingID` INT AUTO_INCREMENT PRIMARY KEY
  - o `PackageID` INT NOT NULL (Foreign Key referencing Packages table)
  - o `UserID` INT NOT NULL (Foreign Key referencing Users table)
  - o `Stars` INT CHECK (Stars BETWEEN 1 AND 5) NOT NULL

- o Comment TEXT
- o RatingDate TIMESTAMP DEFAULT CURRENT_TIMESTAMP

5. Booking Management

- **Bookings**
  - o BookingID INT AUTO_INCREMENT PRIMARY KEY
  - o UserID INT NOT NULL (Foreign Key referencing Users table)
  - o PackageID INT NOT NULL (Foreign Key referencing Packages table)
  - o BookingDate TIMESTAMP DEFAULT CURRENT_TIMESTAMP
  - o TravelDate DATE NOT NULL
  - o TotalAmount DECIMAL(10, 2) NOT NULL
  - o PaymentStatus ENUM('Pending', 'Completed', 'Cancelled') NOT NULL
  - o BookingStatus ENUM('Confirmed', 'Cancelled') NOT NULL
  - o
- **PaymentDetails**
  - o PaymentID INT AUTO_INCREMENT PRIMARY KEY
  - o BookingID INT NOT NULL (Foreign Key referencing Bookings table)
  - o PaymentMethod ENUM('Credit Card', 'PayPal', 'Bank Transfer') NOT NULL
  - o PaymentDate TIMESTAMP DEFAULT CURRENT_TIMESTAMP
  - o AmountPaid DECIMAL(10, 2) NOT NULL
  - o TransactionID VARCHAR(100) UNIQUE
  - o PaymentStatus ENUM('Successful', 'Failed') NOT NULL

6. Tour Itinerary

- **Itineraries**
  - o ItineraryID INT AUTO_INCREMENT PRIMARY KEY
  - o PackageID INT NOT NULL (Foreign Key referencing Packages table)
  - o DayNumber INT NOT NULL
  - o ActivityDescription TEXT NOT NULL

7. Tour Inclusion & Exclusion

- **Inclusions**
  - o InclusionID INT AUTO_INCREMENT PRIMARY KEY
  - o PackageID INT NOT NULL (Foreign Key referencing Packages table)
  - o InclusionDescription TEXT NOT NULL
- **Exclusions**
  - o ExclusionID INT AUTO_INCREMENT PRIMARY KEY
  - o PackageID INT NOT NULL (Foreign Key referencing Packages table)
  - o ExclusionDescription TEXT NOT NULL

8. Departure Information

- **Departures**
  - o DepartureID INT AUTO_INCREMENT PRIMARY KEY

- o `PackageID` INT NOT NULL (Foreign Key referencing Packages table)
- o `DepartureCityID` INT NOT NULL (Foreign Key referencing CityMaster table)
- o `DepartureDate` DATE NOT NULL

9. Contact and Support

- **ContactRequests**
  - o `RequestID` INT AUTO_INCREMENT PRIMARY KEY
  - o `UserID` INT (Foreign Key referencing Users table, Nullable if not logged in)
  - o `Name` VARCHAR(100) NOT NULL
  - o `Email` VARCHAR(100) NOT NULL
  - o `Phone` VARCHAR(15)
  - o `Message` TEXT NOT NULL
  - o `RequestDate` TIMESTAMP DEFAULT CURRENT_TIMESTAMP
  - o `Status` ENUM('Pending', 'Resolved') DEFAULT 'Pending'

ER Diagram :

Entities and Their Attributes

1. **Users**
   - UserID (Primary Key)
   - Username
   - PasswordHash
   - Email
   - FirstName
   - LastName
   - PhoneNumber
   - RoleID (Foreign Key)
   - CreatedAt
2. **Roles**
   - RoleID (Primary Key)
   - RoleName
3. **CountryMaster**
   - CountryID (Primary Key)
   - CountryName
   - CountryCode
4. **StateMaster**
   - StateID (Primary Key)
   - StateName
   - CountryID (Foreign Key)
5. **CityMaster**
   - CityID (Primary Key)
   - CityName
   - StateID (Foreign Key)
6. **CategoryMaster**
   - CategoryID (Primary Key)
   - CategoryName
7. **TagMaster**
   - TagID (Primary Key)
   - TagName
8. **PackageTypeMaster**
   - PackageTypeID (Primary Key)
   - PackageTypeName
9. **TourIncludesMaster**
   - TourIncludeID (Primary Key)
   - IncludeDescription
10. **Packages**
    - PackageID (Primary Key)
    - PackageName
    - Description
    - PackageListImageURL
    - PackageDetailsImageURL
    - DurationDays
    - DurationNights
    - Price

- o PackageTypeID (Foreign Key)
- o CategoryID (Foreign Key)
- o IsActive
- o CreatedAt

11. **PackageTourDetails**
- o PackageTourDetailID (Primary Key)
- o PackageID (Foreign Key)
- o CountryID (Foreign Key)
- o StateID (Foreign Key)
- o CityID (Foreign Key)

12. **PackageTags**
- o PackageTagID (Primary Key)
- o PackageID (Foreign Key)
- o TagID (Foreign Key)

13. **PackageTourIncludes**
- o PackageTourIncludeID (Primary Key)
- o PackageID (Foreign Key)
- o TourIncludeID (Foreign Key)

14. **TotalRatings**
- o RatingID (Primary Key)
- o PackageID (Foreign Key)
- o Stars
- o NumberOfRatings

15. **Ratings**
- o RatingID (Primary Key)
- o PackageID (Foreign Key)
- o UserID (Foreign Key)
- o Stars
- o Comment
- o RatingDate

16. **Bookings**
- o BookingID (Primary Key)
- o UserID (Foreign Key)
- o PackageID (Foreign Key)
- o BookingDate
- o TravelDate
- o TotalAmount
- o PaymentStatus
- o BookingStatus

17. **PaymentDetails**
- o PaymentID (Primary Key)
- o BookingID (Foreign Key)
- o PaymentMethod
- o PaymentDate
- o AmountPaid
- o TransactionID
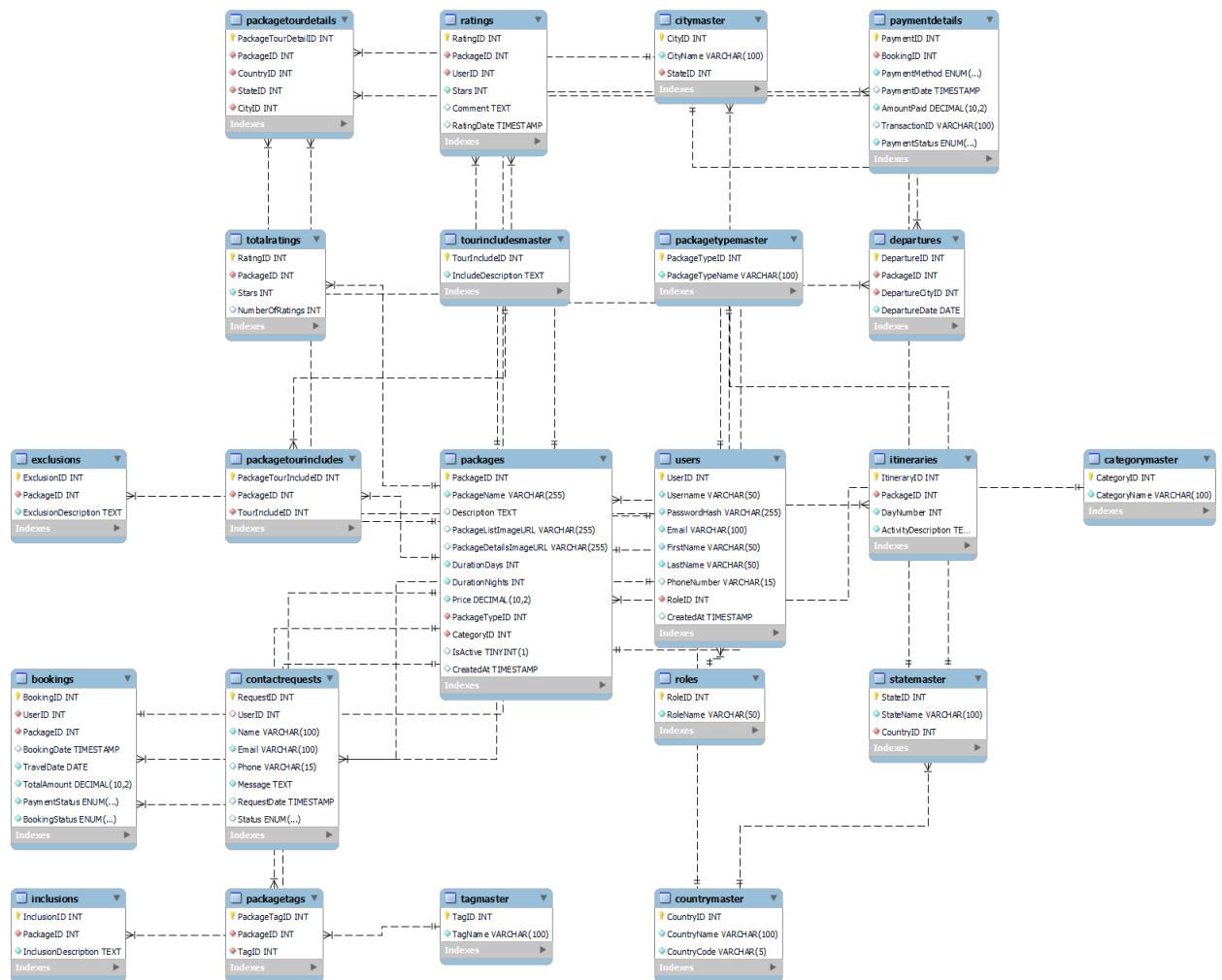- o PaymentStatus

18. **Itineraries**
- o ItineraryID (Primary Key)
- o PackageID (Foreign Key)
- o DayNumber

o ActivityDescription
19. **Inclusions**
   o InclusionID (Primary Key)
   o PackageID (Foreign Key)
   o InclusionDescription
20. **Exclusions**
   o ExclusionID (Primary Key)
   o PackageID (Foreign Key)
   o ExclusionDescription
21. **Departures**
   o DepartureID (Primary Key)
   o PackageID (Foreign Key)
   o DepartureCityID (Foreign Key)
   o DepartureDate
22. **ContactRequests**
   o RequestID (Primary Key)
   o UserID (Foreign Key, Nullable)
   o Name
   o Email
   o Phone
   o Message
   o RequestDate
   o Status

Relationships

- **Users** to **Roles**: Many-to-One (Many Users have One Role)
- **StateMaster** to **CountryMaster**: Many-to-One (Many States belong to One Country)
- **CityMaster** to **StateMaster**: Many-to-One (Many Cities belong to One State)
- **Packages** to **PackageTypeMaster**: Many-to-One (Many Packages belong to One PackageType)
- **Packages** to **CategoryMaster**: Many-to-One (Many Packages belong to One Category)
- **PackageTourDetails** to **Packages**: Many-to-One (Many Package Tour Details for One Package)
- **PackageTourDetails** to **CountryMaster**, **StateMaster**, **CityMaster**: Many-to-One (Many Details for One Country, State, City)
- **PackageTags** to **Packages** and **TagMaster**: Many-to-One (Many Tags for One Package, Many Tags belong to One Tag)
- **PackageTourIncludes** to **Packages** and **TourIncludesMaster**: Many-to-One (Many Includes for One Package, Many Includes belong to One Include)
- **Ratings** to **Packages** and **Users**: Many-to-One (Many Ratings for One Package, Many Ratings belong to One User)
- **Bookings** to **Users** and **Packages**: Many-to-One (Many Bookings for One User and One Package)
- **PaymentDetails** to **Bookings**: Many-to-One (Many Payments for One Booking)
- **Itineraries** to **Packages**: Many-to-One (Many Itineraries for One Package)
- **Inclusions** and **Exclusions** to **Packages**: Many-to-One (Many Inclusions and Exclusions for One Package)

- **Departures** to **Packages** and **CityMaster**: Many-to-One (Many Departures for One Package, Many Departures belong to One City)
- **ContactRequests** to **Users**: Many-to-One (Many Requests for One User, Nullable)

Mysql Queries :

```sql
CREATE TABLE Roles (

    RoleID INT AUTO_INCREMENT PRIMARY KEY,

    RoleName VARCHAR(50) UNIQUE NOT NULL

);


CREATE TABLE Users (

    UserID INT AUTO_INCREMENT PRIMARY KEY,

    Username VARCHAR(50) UNIQUE NOT NULL,

    PasswordHash VARCHAR(255) NOT NULL,

    Email VARCHAR(100) UNIQUE NOT NULL,

    FirstName VARCHAR(50) NOT NULL,

    LastName VARCHAR(50) NOT NULL,

    PhoneNumber VARCHAR(15),

    RoleID INT NOT NULL,

    CreatedAt TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    FOREIGN KEY (RoleID) REFERENCES Roles(RoleID)

);


CREATE TABLE CountryMaster (

    CountryID INT AUTO_INCREMENT PRIMARY KEY,

    CountryName VARCHAR(100) UNIQUE NOT NULL,

    CountryCode VARCHAR(5) UNIQUE NOT NULL

);
```

```sql
CREATE TABLE StateMaster (

    StateID INT AUTO_INCREMENT PRIMARY KEY,

    StateName VARCHAR(100) NOT NULL,

    CountryID INT NOT NULL,

    FOREIGN KEY (CountryID) REFERENCES CountryMaster(CountryID)

);


CREATE TABLE CityMaster (

    CityID INT AUTO_INCREMENT PRIMARY KEY,

    CityName VARCHAR(100) NOT NULL,

    StateID INT NOT NULL,

    FOREIGN KEY (StateID) REFERENCES StateMaster(StateID)

);


CREATE TABLE CategoryMaster (

    CategoryID INT AUTO_INCREMENT PRIMARY KEY,

    CategoryName VARCHAR(100) UNIQUE NOT NULL

);


CREATE TABLE TagMaster (

    TagID INT AUTO_INCREMENT PRIMARY KEY,

    TagName VARCHAR(100) UNIQUE NOT NULL

);
```

```sql
CREATE TABLE PackageTypeMaster (

    PackageTypeID INT AUTO_INCREMENT PRIMARY KEY,

    PackageTypeName VARCHAR(100) UNIQUE NOT NULL

);


CREATE TABLE TourIncludesMaster (

    TourIncludeID INT AUTO_INCREMENT PRIMARY KEY,

    IncludeDescription TEXT NOT NULL

);

CREATE TABLE Packages (

    PackageID INT AUTO_INCREMENT PRIMARY KEY,

    PackageName VARCHAR(255) NOT NULL,

    Description TEXT,

    PackageListImageURL VARCHAR(255),

    PackageDetailsImageURL VARCHAR(255),

    DurationDays INT NOT NULL,

    DurationNights INT NOT NULL,

    Price DECIMAL(10, 2) NOT NULL,

    PackageTypeID INT NOT NULL,

    CategoryID INT NOT NULL,

    IsActive BOOLEAN DEFAULT TRUE,

    CreatedAt TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    FOREIGN KEY (PackageTypeID) REFERENCES PackageTypeMaster(PackageTypeID),

    FOREIGN KEY (CategoryID) REFERENCES CategoryMaster(CategoryID)

);
```

```sql
CREATE TABLE PackageTourDetails (

    PackageTourDetailID INT AUTO_INCREMENT PRIMARY KEY,

    PackageID INT NOT NULL,

    CountryID INT NOT NULL,

    StateID INT NOT NULL,

    CityID INT NOT NULL,

    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID),

    FOREIGN KEY (CountryID) REFERENCES CountryMaster(CountryID),

    FOREIGN KEY (StateID) REFERENCES StateMaster(StateID),

    FOREIGN KEY (CityID) REFERENCES CityMaster(CityID)

);


CREATE TABLE PackageTags (

    PackageTagID INT AUTO_INCREMENT PRIMARY KEY,

    PackageID INT NOT NULL,

    TagID INT NOT NULL,

    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID),

    FOREIGN KEY (TagID) REFERENCES TagMaster(TagID)

);


CREATE TABLE PackageTourIncludes (

    PackageTourIncludeID INT AUTO_INCREMENT PRIMARY KEY,

    PackageID INT NOT NULL,

    TourIncludeID INT NOT NULL,
```

```sql
    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID),

    FOREIGN KEY (TourIncludeID) REFERENCES TourIncludesMaster(TourIncludeID)

);

CREATE TABLE TotalRatings (

    RatingID INT AUTO_INCREMENT PRIMARY KEY,

    PackageID INT NOT NULL,

    Stars INT CHECK (Stars BETWEEN 1 AND 5) NOT NULL,

    NumberOfRatings INT,

    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID)

);


CREATE TABLE Ratings (

    RatingID INT AUTO_INCREMENT PRIMARY KEY,

    PackageID INT NOT NULL,

    UserID INT NOT NULL,

    Stars INT CHECK (Stars BETWEEN 1 AND 5) NOT NULL,

    Comment TEXT,

    RatingDate TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID),

    FOREIGN KEY (UserID) REFERENCES Users(UserID)

);

CREATE TABLE Bookings (

    BookingID INT AUTO_INCREMENT PRIMARY KEY,

    UserID INT NOT NULL,

    PackageID INT NOT NULL,
```

```sql
    BookingDate TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    TravelDate DATE NOT NULL,

    TotalAmount DECIMAL(10, 2) NOT NULL,

    PaymentStatus ENUM('Pending', 'Completed', 'Cancelled') NOT NULL,

    BookingStatus ENUM('Confirmed', 'Cancelled') NOT NULL,

    FOREIGN KEY (UserID) REFERENCES Users(UserID),

    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID)

);

CREATE TABLE PaymentDetails (

    PaymentID INT AUTO_INCREMENT PRIMARY KEY,

    BookingID INT NOT NULL,

    PaymentMethod ENUM('Credit Card', 'PayPal', 'Bank Transfer') NOT NULL,

    PaymentDate TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    AmountPaid DECIMAL(10, 2) NOT NULL,

    TransactionID VARCHAR(100) UNIQUE,

    PaymentStatus ENUM('Successful', 'Failed') NOT NULL,

    FOREIGN KEY (BookingID) REFERENCES Bookings(BookingID)

);

CREATE TABLE Itineraries (

    ItineraryID INT AUTO_INCREMENT PRIMARY KEY,

    PackageID INT NOT NULL,

    DayNumber INT NOT NULL,

    ActivityDescription TEXT NOT NULL,

    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID)

);
```

```sql
CREATE TABLE Inclusions (

    InclusionID INT AUTO_INCREMENT PRIMARY KEY,

    PackageID INT NOT NULL,

    InclusionDescription TEXT NOT NULL,

    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID)

);


CREATE TABLE Exclusions (

    ExclusionID INT AUTO_INCREMENT PRIMARY KEY,

    PackageID INT NOT NULL,

    ExclusionDescription TEXT NOT NULL,

    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID)

);

CREATE TABLE Departures (

    DepartureID INT AUTO_INCREMENT PRIMARY KEY,

    PackageID INT NOT NULL,

    DepartureCityID INT NOT NULL,

    DepartureDate DATE NOT NULL,

    FOREIGN KEY (PackageID) REFERENCES Packages(PackageID),

    FOREIGN KEY (DepartureCityID) REFERENCES CityMaster(CityID)

);
```

```sql
CREATE TABLE ContactRequests (

    RequestID INT AUTO_INCREMENT PRIMARY KEY,

    UserID INT,

    Name VARCHAR(100) NOT NULL,

    Email VARCHAR(100) NOT NULL,

    Phone VARCHAR(15),

    Message TEXT NOT NULL,

    RequestDate TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    Status ENUM('Pending', 'Resolved') DEFAULT 'Pending',

    FOREIGN KEY (UserID) REFERENCES Users(UserID)

);
```

ACTION PLANS

#	ACTIONS	DATE	DURATION	REMARK
1	Database Design			
2	API Design using NodeJS			
3	Admin Template Desing using React			
4	Linking Backend with Front End			
5	Testing			
6	Deployment			
7	UAT			