

## **Specification:** Database modeling for Airbnb platform.

### **Abstract:**

This project aims to develop a comprehensive database for an online booking platform like Airbnb, following three essential phases: Conception, Development, and Finalization. Airbnb is a global platform that connects hosts offering accommodations with guests looking for temporary lodging. Guests can search for available properties based on various criteria such as location, price, and dates, and make reservations. Payment is processed through credit or debit cards. Each user, whether a host or a guest, has a profile page. Hosts are required to upload a picture and phone number, while guests provide additional information. Users can link their profiles to social networks like Facebook and Instagram. Both hosts and guests can receive discounts via coupons and leave reviews, ratings, and complaints about their experiences.

### **Roles and actions:**

This system supports several roles with distinct permissions: Guests: Users who can browse and book listings for temporary accommodations, applying filters to refine their search. Hosts: Users who offer accommodations, can publish and manage listings, and communicate with guests. Employees: Users who can add, edit, or remove data records and manage access and permissions.

### **Required Data and Functions:**

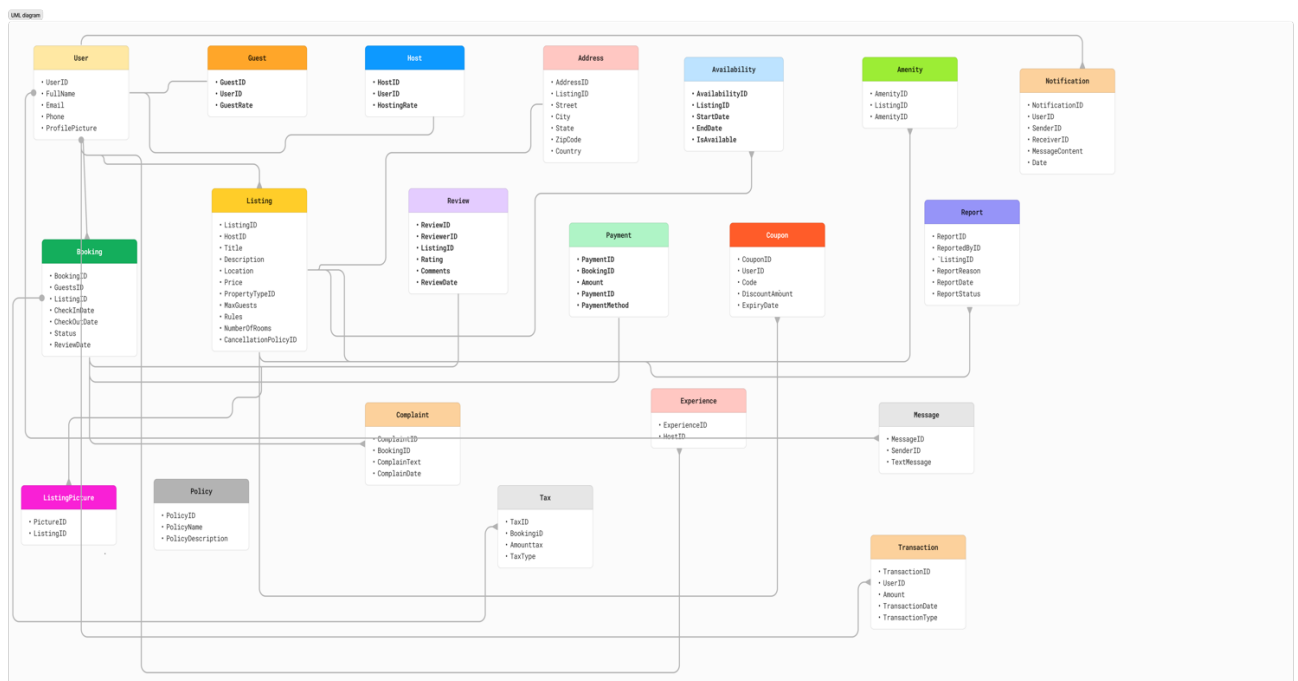
**User data:** UserID, Name, Email, Phone, ProfilePicture.

**Listing data:** ListingID, HostID, Title, Description.

**Booking data:** BookingID, GuestID, ListingID, BookingDates, Status, Payment.

**Review data:** ReviewID, ReviewerID, ListingID, Rating, Comments.

## Entity Relationship Model (ERM):



## Data Dictionary:

**\*\*User Table:\*\***

- `UserID` (Primary Key, INT): Unique identifier for each user.
- `Name` (VARCHAR(100)): User's name.

- `Email` (VARCHAR(250)): User's email address.
- `Phone` (TEXT): User's phone number.
- `ProfilePicture` (VARCHAR(250)): URL to the user's profile picture.
- `VerifiedStatus` (VARCHAR(50)): Indicates whether the user is verified.

**\*\*Host Table:\*\***

- `HostID` (Primary Key, INT): Unique identifier for each host.
- `UserID` (Foreign Key, INT): References `UserID` in the Users table.
- `HostRating` (FLOAT): Rating of the host.

**\*\*Guest Table:\*\***

- `GuestID` (Primary Key, INT): Unique identifier for each guest.
- `UserID` (Foreign Key, INT): References `UserID` in the Users table.
- `GuestRating` (FLOAT): Rating of the guest.

**\*\*Listing Table:\*\***

- `ListingID` (Primary Key, INT): Unique identifier for each listing.
- `HostID` (Foreign Key, INT): References `HostID` in the Hosts table.
- `Title` (VARCHAR(100)): Title of the listing.
- `Description` (VARCHAR(250)): Description of the listing.
- `Location` (VARCHAR(200)): Location of the listing.
- `Price` (FLOAT): Nightly price of the listing.
- `PropertyID` (Foreign Key, INT): References `PropertyTypeID` in the PropertyType table.
- `MaxGuests` (INT): Maximum number of guests allowed.
- `NumberOfBedrooms` (INT): Number of bedrooms.

- `Rules` (TEXT): Rules for the listing.
- `CancellationPolicyID` (Foreign Key, INT): References `PolicyID` in the Policy table.

**\*\*PropertyType Table:\*\***

- `PropertyTypeID` (Primary Key, INT): Unique identifier for each property type.
- `TypeName` (VARCHAR(50)): Name of the property type.

**\*\*Booking Table:\*\***

- `BookingID` (Primary Key, INT): Unique identifier for each booking.
- `GuestID` (Foreign Key, INT): References `GuestID` in the Guests table.
- `ListingID` (Foreign Key, INT): References `ListingID` in the Listings table.
- `CheckInDate` (DATE): Check-in date.
- `CheckOutDate` (DATE): Check-out date.
- `Status` (VARCHAR(50)): Booking status.
- `PaymentID` (Foreign Key, INT): References `PaymentID` in the Payments table.

**\*\*Review Table:\*\***

- `ReviewID` (Primary Key, INT): Unique identifier for each review.
- `ReviewerID` (Foreign Key, INT): References `UserID` in the Users table.
- `ListingID` (Foreign Key, INT): References `ListingID` in the Listings table.
- `Rating` (FLOAT): Rating given by the reviewer.
- `Comments` (TEXT): Comments provided by the reviewer.
- `ReviewDate` (TIMESTAMP): Date of the review.

**\*\*Payment Table:\*\***

- `PaymentID` (Primary Key, INT): Unique identifier for each payment.
- `BookingID` (Foreign Key, INT): References `BookingID` in the Bookings table.
- `Amount` (DECIMAL(10, 2)): Amount paid.
- `PaymentDate` (DATE): Date of the payment.
- `PaymentMethod` (VARCHAR(50)): Method of payment.

**\*\*Complaint Table:\*\***

- `ComplaintID` (Primary Key, INT): Unique identifier for each complaint.
- `BookingID` (Foreign Key, INT): References `BookingID` in the Bookings table.
- `ComplaintText` (TEXT): Text of the complaint.
- `ComplaintDate` (DATE): Date the complaint was made.

**\*\*Discount Table:\*\***

- `DiscountID` (Primary Key, INT): Unique identifier for each discount.
- `Code` (VARCHAR(50)): Discount code.
- `Percentage` (FLOAT): Discount percentage.
- `ExpirationDate` (DATE): Expiry date of the discount code.