

Summary:

The project is to layout and put into impact a database schema for an Airbnb-like platform that allows the renting and booking of inns. The platform needs to control users, their profiles, listings, bookings, evaluations, bills, social community integrations, messages, availability of listings, amenities, towns, host and guest management, reservations, renovation, list pix, and coupon manipulate. The database needs to aid numerous functionalities which include consumer registration, list advent, reserving management, assessment submission, charge processing, and greater. Ensuring statistics integrity, inexperienced querying, and retaining relationships among various entities are critical annoying conditions.

Requirement Specification:

Roles (Person/User Groups)

1. Admin:

- Manage user accounts (create, update, delete, deactivate).
- Approve or reject property listings.
- Oversee bookings and handle disputes.
- Generate reports on platform activity.
- Manage and update platform policies.
- Oversee payment transactions and handle refunds.
- Perform system maintenance and updates.

2. Host:

- Create and manage property listings.
- Set pricing, availability, and house rules for properties.
- Respond to booking requests and messages from guests.
- Manage bookings (approve, reject, or modify bookings).
- Review and rate guests.
- Receive payments for bookings.
- Update profile information and verification details.

3. Guest:

- Browse property listings.
- Filter and search for properties based on preferences.
- Request or confirm bookings.
- Make payments for bookings.
- Communicate with hosts via messaging.

- Leave reviews and ratings for properties and hosts.
- Manage personal profile and preferences.

4. User (Non-Registered Visitor):

- Browse available property listings.
- Search and filter properties without booking.
- Register or sign up to become a host or guest.

5. System (Automated Processes):

- Send automated notifications and reminders to users.
- Process payments and refunds.
- Update availability status of listings.
- Monitor for fraudulent activities and flag suspicious behavior.
- Handle automated data backups and system updates.

Data and Functions Required

Data:

1. **User Data:** User ID, name, email, password, role (Admin, Host, Guest), contact details, profile picture, verification status.
2. **Profile Data:** Bio, preferences, social media links, verification documents, reviews received, ratings.
3. **Listing Data:** Listing ID, property title, description, location, amenities, pricing, availability, images, house rules, host ID.
4. **Booking Data:** Booking ID, guest ID, host ID, listing ID, booking dates, status (pending, confirmed, canceled), total amount, payment status.
5. **Review Data:** Review ID, listing ID, guest ID, host ID, rating, review text, date of review.
6. **Payment Data:** Payment ID, booking ID, guest ID, host ID, amount, payment method, status, date of transaction.
7. **Messaging Data:** Message ID, sender ID, receiver ID, content, timestamp, status (read/unread).
8. **Social Network Data:** Links to social media profiles, friend lists, shared content.
9. **Maintenance Data:** Maintenance request ID, listing ID, issue description, status, maintenance schedule.

Functions

1. User Management

- Functions for registering, logging in, updating profile, changing passwords, and managing user roles.

2. Listing Management

- Functions for creating, updating, deleting, searching property listings. Also includes functions to manage pricing, availability, and house rules.

3. Booking Management

- Functions for browsing listings, making bookings, confirming or canceling bookings, and handling payments.

4. Review and Rating

- Functions to submit, view, update, and delete reviews and ratings for properties, guests, and hosts.

5. Messaging

- Functions to send, receive, and manage messages between guests and hosts.

6. Admin Management

- Functions for overseeing user activities, managing listings, handling disputes, and generating reports.

7. Payment Processing

- Functions to handle payments, process refunds, manage transactions, and generate payment receipts.

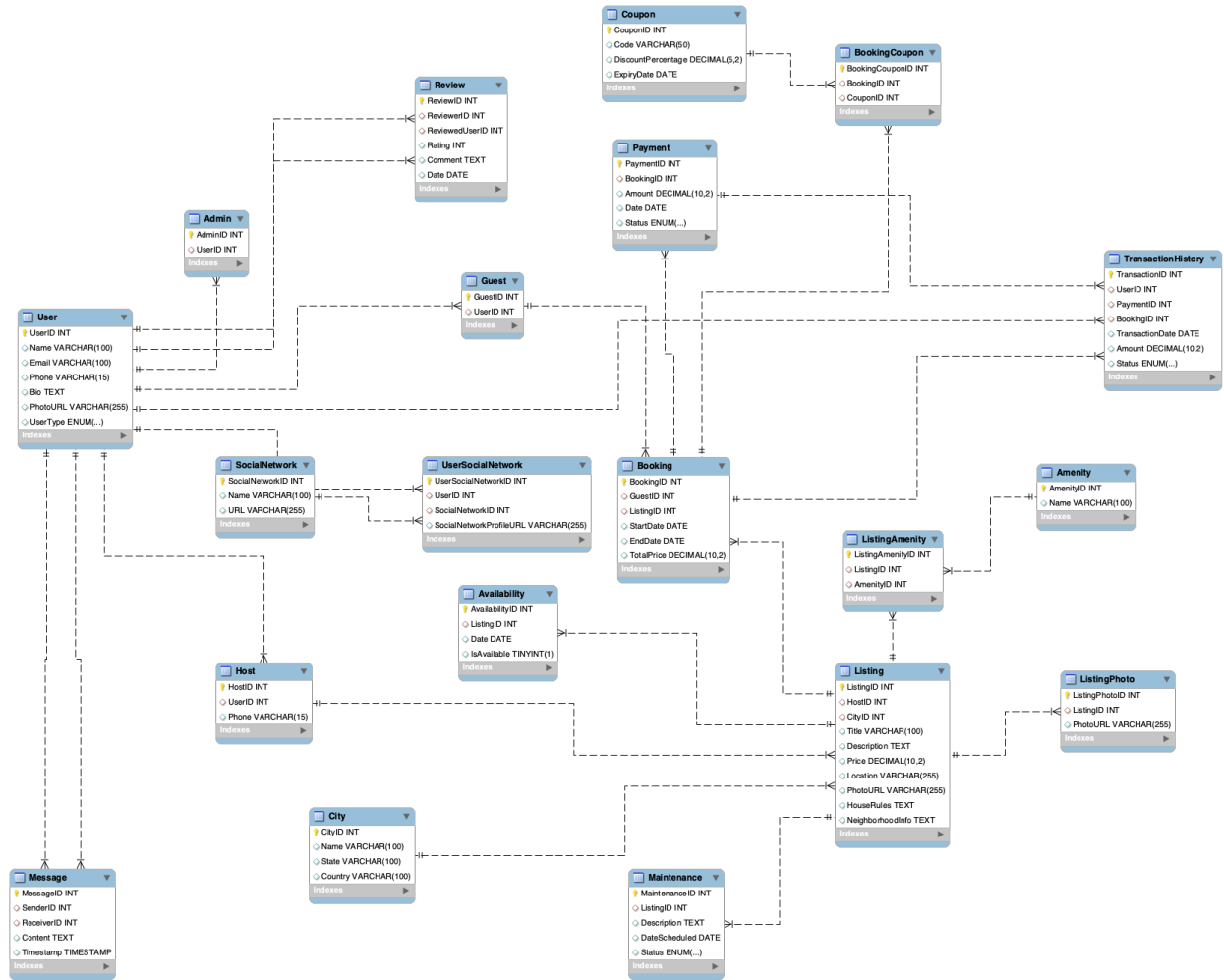
8. Notification System

- Functions for sending notifications about booking confirmations, cancellations, payment reminders, and system updates.

9. Maintenance Management:

- Functions to report maintenance issues, schedule repairs, and update maintenance status.

Updated ERM-Diagram



Updated Data Dictionary

1. User Table:

Column Name	Data Type	Description
UserID	INT	Primary key: Unique identifier for each user.
Name	VARCHAR(100)	Name of the user
Email	VARCHAR(100)	Unique email address of the user
Phone	VARCHAR(15)	Contact number of the user
Bio	TEXT	Unique email address of the user
PhotoURL	VARCHAR(255)	Contact number of the user
UserType	ENUM('Host', 'Guest', 'Admin')	Defines the role of the user in the system

2. Host Table:

Column Name	Data Type	Description
HostID	INT	Primary key; Unique identifier for each host.
UserID	INT	Foreign key referencing User(UserID); Identifies the user who is a host.
Phone	VARCHAR(15)	Contact number of the host.

3. Guest Table:

Column Name	Data Type	Description
GuestID	INT	Primary key; Unique identifier for each guest.
UserID	INT	Foreign key referencing User(UserID); Identifies the user who is a guest.

4. Admin Table:

Column Name	Data Type	Description
-------------	-----------	-------------

AdminID	INT	Primary key; Unique identifier for each admin.
UserID	INT	Foreign key referencing User(UserID); Identifies the user who is an admin.

5. City Table:

Column Name	Data Type	Description
CityID	INT	Primary key; Unique identifier for each city.
Name	VARCHAR(100)	Name of the city.
State	VARCHAR(100)	State where the city is located.
Country	VARCHAR(100)	Country where the city is located.

6. Listing Table:

Column Name	Data Type	Description
ListingID	INT	Primary key: Unique identifier for each listing.
HostID	INT	Foreign key referencing User(UserID); Identifies the host.
Title	VARCHAR(100)	Title of the property listing
Description	TEXT	Detailed description of the property.
Price	DECIMAL(10, 2)	Price per night for the listing.
Location	VARCHAR(255)	Address or location of the property.
PhotoURL	VARCHAR(255)	URL to the primary photo of the listing.
HouseRules	TEXT	House rules mentioned by the owner
NeighborhoodInfo	TEXT	Neighborhood information including do's and dont's

7. Availability Table:

Column Name	Data Type	Description
AvailabilityID	INT	Primary key; Unique identifier for each availability entry.
ListingID	INT	Foreign key referencing Listing(ListingID); Identifies the listing.
Date	DATE	Date for which availability is set.
IsAvailable	BOOLEAN	Indicates whether the listing is available on the given date.

8. Booking Table:

Column Name	Data Type	Description
BookingID	INT	Primary key: Unique identifier for each booking.
GuestID	INT	Foreign key referencing User(UserID); Identifies the guest.
ListingID	INT	Foreign key referencing Listing(ListingID); Identifies the listing.
StartDate	DATE	Date when the booking starts.
EndDate	DATE	Date when the booking ends.
TotalPrice	DECIMAL(10, 2)	Total price for the booking period.

9. Review Table:

Column Name	Data Type	Description
ReviewID	INT	Primary key: Unique identifier for each review.

ReviewerID	INT	Foreign key referencing User(UserID); Identifies the reviewer.
ReviewedUserID	INT	Foreign key referencing User(UserID); Identifies the reviewed user.
Rating	INT	Rating given by the reviewer (1 to 5).
Comment	TEXT	Comments or feedback left by the reviewer.
Date	DATE	Date when the review was submitted.

10. Payment Table:

Column Name	Data Type	Description
PaymentID	INT	Primary key: Unique identifier for each payment.
BookingID	INT	Foreign key referencing Booking(BookingID); Identifies the associated booking.
Amount	DECIMAL(10, 2)	Amount paid for the booking.
Date	DATE	Date when the payment was made.
Status	ENUM('Pending', 'Completed', 'Failed')	Status of the payment transaction.

11. SocialNetwork Table:

Column Name	Data Type	Description
SocialNetworkID	INT	Primary key: Unique identifier for each social network.
Name	VARCHAR(100)	Name of the social network (e.g., Facebook, Twitter).
URL	VARCHAR(255)	URL to the social network's homepage.

12. UserSocialNetwork Table:

Column Name	Data Type	Description
UserSocialNetworkID	INT	Primary key; Unique identifier for each user-social network relationship.
UserID	INT	Foreign key referencing User(UserID); Identifies the user.
SocialNetworkID	INT	Foreign key referencing SocialNetwork(SocialNetworkID); Identifies the social network.
SocialNetworkProfileURL	VARCHAR(255)	URL to the user's profile on the social network.

13. Message Table:

Column Name	Data Type	Description
MessageID	INT	Primary key; Unique identifier for each message.
SenderID	INT	Foreign key referencing User(UserID); Identifies the sender.
ReceiverID	INT	Foreign key referencing User(UserID); Identifies the receiver.
Content	TEXT	Content of the message.
Timestamp	TIMESTAMP	Time when the message was sent.

14. Amenity Table:

Column Name	Data Type	Description
AmenityID	INT	Primary key; Unique identifier for each amenity.
Name	VARCHAR(100)	Name of the amenity (e.g., WiFi, Pool, Parking).

15. ListingAmenity Table:

Column Name	Data Type	Description
ListingAmenityID	INT	Primary key; Unique identifier for each listing-amenity relationship.
ListingID	INT	Foreign key referencing Listing(ListingID); Identifies the listing.
AmenityID	INT	Foreign key referencing Amenity(AmenityID); Identifies the amenity.

16. Maintenance Table:

Column Name	Data Type	Description
MaintenanceID	INT	Primary key; Unique identifier for each maintenance record.
ListingID	INT	Foreign key referencing Listing(ListingID); Identifies the listing.
Description	TEXT	Description of the maintenance issue.
DateScheduled	DATE	Date when maintenance is scheduled.
Status	ENUM('Pending', 'Completed')	Status of the maintenance request.

17. ListingPhoto Table:

Column Name	Data Type	Description
ListingPhotoID	INT	Primary key; Unique identifier for each listing photo.

ListingID	INT	Foreign key referencing Listing(ListingID); Identifies the listing.
PhotoURL	VARCHAR(255)	URL to the photo of the listing.

18. Coupon Table:

Column Name	Data Type	Description
CouponID	INT	Primary key; Unique identifier for each coupon.
Code	VARCHAR(50)	Coupon code used by guests.
DiscountPercentage	DECIMAL(5, 2)	Percentage discount offered by the coupon.
ExpiryDate	DATE	Expiration date of the coupon.

19. BookingCoupon Table:

Column Name	Data Type	Description
BookingCouponID	INT	Primary key; Unique identifier for each booking-coupon relationship.
BookingID	INT	Foreign key referencing Booking(BookingID); Identifies the booking.
CouponID	INT	Foreign key referencing Coupon(CouponID); Identifies the coupon used.

20. Reservation Table:

Column Name	Data Type	Description
TransactionID	INT	Primary key; Unique identifier for each transaction.

UserID	INT	Foreign key; References User(UserID). Identifies the user associated with the transaction.
PaymentID	INT	Foreign key; References Payment(PaymentID). Identifies the payment associated with the transaction.
BookingID	INT	Foreign key; References Booking(BookingID). Identifies the booking associated with the transaction.
TransactionDate	DATE	The date when the transaction occurred.
Amount	DECIMAL(10, 2)	The amount of money involved in the transaction.
Status	ENUM('Pending', 'Completed', 'Failed')	The current status of the transaction.