

Our Airbnb platform is built upon a comprehensive database schema that supports a wide range of functionality and features. The database schema is designed to store and manage data related to reservations, payment processing, user profiles, and reviews made by guests and hosts.

The database schema is made up of 23 tables, each of which serves a specific purpose and contains data related to a particular aspect of the platform's functionality. These tables include the following:

- `guest_profile`: The table stores information about guests who have signed up on the platform.
- `host_profile`: The table stores information about hosts who have signed up on the platform.
- `login_guest`: The table stores login credentials for guests.
- `login_host`: The table stores login credentials for hosts.
- `guest_about`: The table stores additional information about the guest's profile, such as their work and language preferences.
- `guest_card_details`: The table stores the guest's credit card details for payment processing.
- `identity_verification`: The table stores the guest's government ID verification information.
- `emergency_contact`: The table stores the guest's emergency contact information.
- `host_place`: The table stores information about the places that hosts have listed on the platform.
- `rooms_spaces`: The table stores information about the rooms and spaces in the place listed by the host.
- `place_location`: The table stores the location information for the place listed by the host.
- `place_highlight`: The table stores the highlights or unique features of the place listed by the host.
- `property_guests`: The table stores information about the property guests.
- `place_offer`: The table stores information about the amenities offered by the property.
- `Reservation`: The table stores information about the reservations made by guests.
- `guest_payment_type`: The table stores information about the payment methods available to the guest.
- `setup_payouts`: The table stores information about the payout setup for the host.
- `transaction_approval`: The table stores information about the approval of a transaction by the host.
- `host_payment_type`: The table stores information about the payment methods available to the host.
- `payout_methods`: The table stores information about the payout methods for a transaction.
- `price_calculator`: The table stores the calculated prices for a reservation.
- `review_guest`: The table stores the reviews made by guests after their stay.

- `review_host`: The table stores the reviews made by hosts after a guest's stay.

The database schema uses foreign keys to establish relationships between tables, allowing the platform to retrieve and display data from multiple tables as needed. For example, the `reservation` table includes foreign keys referencing the `guest_profile`, `host_place`, and `host_profile` tables, allowing the platform to retrieve information about the reservation, guest, place, and host in a single query.

The metadata stored in the system includes 20 entries for each table, with the number of entries depending on the amount of data stored in the system. For example, the `reservation` table will have entries for each reservation made by guests, while the `review_guest` and `review_host` tables will have entries for each review made by guests and hosts, respectively.

As for the size of the database (rough estimate): 928.0 KiB of data stored in the system. However, we have implemented various measures to optimize the database's performance and ensure that it can scale to handle large volumes of data. These measures include indexing, query optimization, and partitioning.

In conclusion, our reservation and payment processing platform's database schema is designed to support a wide range of functionality and features, including reservation management, payment processing, and guest and host reviews. The database schema includes 23 tables, each serving a specific purpose, and uses foreign keys to establish relationships between tables. The metadata stored in the system includes 20 entries for each table, with the size of the database depending on the amount of data stored. We are confident that our database schema can scale to handle large volumes of data and support the continued growth and success of our platform.