

Airbnb Database model requirements analysis – Tom Schmaeling

In this short paper we will analyze the web application Airbnb to try and imitate the database model it uses for its many functionalities.

When imitating the Airbnb database model, it is crucial to provide robust support scalability, ensuring the system can handle vast amounts of data, user profiles, and transactions. Additionally, flexibility in schema design is vital to accommodate diverse property types, amenities, allowing seamless modifications to adapt to changing business requirements and user needs.

To achieve this, we will start by studying and researching the Airbnb website as well as the assignment text, as it also contains some relevant information about the application. While doing so we will note down relevant user groups and dynamic information that is used by the applications' functionalities.

By first walking through the most common use cases of the website we can identify multiple entities right away, the following list is the result after a simple "walk-through" of the website and noting all relevant entities in a mostly chronological order:

- Users
- Hosts
- Guests
- Addresses
- Emergency Contacts
- Property listings
- Property type
- Property categories
- Property amenities
- Reviews
- Ratings
- Images
- Wishlist
- Trip History
- Reservations / Bookings
- Chats
- Messages

What role some of these entities play in the application and what attributes might be relevant is listed in more detail below.

Once these entities have been noted down, it is essential to figure out how these entities might relate to each other so we can document their relationships as well as start thinking about the cardinalities of said relationships.

As the process is an iterative one, changes to entities or relationships are quite common during this step. The result should be cohesive, easily understandable and provide the template for implementation is the second phase.

The main problem this database is facing is how we structure the property listings and its different elements as well as the users and bookings and transactions. These are the most important entities in the database and will form the basis for the other tables.

Roles / User Groups

As noted in the assignment there are two main categories of users, hosts and guests. These two groups differ somewhat, in that they share most of their attributes but the requirements (NOT NULL) of such some attributes change depending on which type of user it is.

Generally, a Host is required to upload a profile image, personal information, a description and bank information to receive payments.

Guests, on the other hand, need to upload a government ID and credit card information. Important to note here is that a booking can only have one guest. Other guests are still handled through the booking guest user and are handled by the “number of guests” attribute.

We will use the joined subclass table strategy for this purpose. The reasoning for the “duplicate” attributes in guest and host is the difference in requirements.

Property-Listings

The term property-listings refers to the many different properties offered from hosts to guests on the website. Looking at the filters, we can see that a property has a type, such as house, apartment, guesthouse, or hotel and may also have multiple categories, such as cabin, tiny homes, mansions, etc.

When looking at a listing we see that there are images, a name, an address, prices per night (including cleaning fee, Airbnb service fee and taxes), reviews, descriptions, amenities and a calendar that displays the available times for booking. The host is also displayed on the property listing page.

Bookings

A booking, also called reservation lists the nightly and total price, the number of guests, a confirmation code, the timeframe (start and end date), etc. More specific payment information as well as other relevant information is stored in the transaction entity.

Addresses

The addresses in Airbnb are made up of country, state or region, and city. The street / house number are also part of the address but are not relevant when first filtering through the property listings.

