A short description of the purpose of the project. A description of the conceptual problems encountered and the justification of your solutions.

Purpose

The purpose of this project is to understand the process of designing a database application that supports typical queries and operations expected of a large application such as AirBnB.

Problems Encountered

- How can we efficiently get past bookings for a renter/host (to be used for reviewing and such)? In addition, how can we only show availabilities for a booking that are not before the current date?

    Solution & Justification: Initially we thought to check the date of each booking/availability against the current date. However, this is a costly check to perform each time. Instead, we maintain a "status" field in both the Booked/Availabilities tables.

    When a user goes to see their upcoming bookings, we first filter all the bookings for the user, and update the status of any bookings that have an endDate before the current date, to 'past'. Then, we get all relevant bookings with status 'booked'. A similar process occurs when a user chooses to see availabilities for a specific listing. This ensures that the status is only updated once and only when a user cares to view the information.

- How can we easily display information about max cancellations for a renter/host?

    Solution & Justification: Again, maintaining a status on our Bookings table with staus 'cancelled' (instead of deleting the row from the table entirely) allowed us to easily retrieve this information.

- How can we maintain the idea of a "logged-in user" without asking for a SIN for each query or operation (as SIN is the primary key for a user)?

    Solution & Justification: Ensure that a username is unique for each user and allow them to log in using their username and password. Their SIN is stored in the backend for all subsequent queries. This hides sensitive details about the user from the console and doesn't require re-entering of the same info.

- There can be 3 types of Comments and Rating. We originally wanted an entity for Comments and an entity for Ratings, each of which would have relationships from Renters to Hosts, Renters to Listings, and Hosts to Renters.. However, it is difficult to distinguish each type of rating/comment, especially for Renters review Hosts and Hosts review Renters because they comment on/rate each other.

Solution & Justification: Instead of Rating and Comment entities, we created relationship tables called RentersReviewListings, RentersReviewHosts, HostsReviewRenters with SINs/ListIDs as the foreign keys and attributes Review and Rating. This makes it much easier to distinguist whether the HostSIN is reviewing RenterSIN or vice versa based on the table names.

- How can we have a unique address when different units of a hotel/apartment are only distinguished by unit number, whereas houses/guesthouses usually do not have a unit number?

    Solution & Justification: Use a default for the unitNumber field in the Addresses table, the default of 0 is used for houses and guesthouses. At the application level, users are not prompted to enter a unit number if they select their listing type as house or guesthouse. This allows us to maintain a relevant primary key for the Addresses table, while only showing relevant information to the user.

- How to implement a full search when there will be multiple choices made by the user which lead to many different queries?

    Solution & Justification: Instead of calling the database once with a query that combines every choice made by the user (whether to filter by searching by postal code, a set of amenities, time window of availability, and/or a price range), the filter was divided into steps. For each choice, a call was made to the database with a different query depending on whether the user chose to filter with the given option (such as postal code). The result for the query regardless of whether the user chose yes or no was stored in a view (ex. postalView). Then, the subsequent query based on the next choice the user makes is performed on the previous view which contains the Listings that are results of the choices thus far.

## The assumptions you made (if any).

Storing Data and Constraints

- We assume a renter can only leave 1 review for a host and 1 review per listing. The same applies for a host leaving a review for a renter.

- We assume a listing can only be associated with 1 host, and a booking can only be associated with 1 renter.

- When a host cancels a booking, the listing does not become available for booking again. However, when a renter cancels a booking, it becomes available.

Assumptions for Queries & Reports

- We generate reports only for listings that are currently on the site; deleted listings, and their associated bookings do not contribute to the reports.

- "We are only interested in ranking those renters that have made at least two bookings in the year." We assume "in the year" means renters who have made 2 bookings in the same year as the year of the start date the user is filtering by.

- When displaying max cancellations within a year, we assume the startdate of the booking must be within the current year.

- Reports about number of bookings (by city/postal or ranking renters) only count bookings that have not been cancelled

- Whenever price is involved in filtering/reports/suggesting prices, only listings that have availabilities are shown.

- When filtering by availabilities, we only return listings that are available for the whole range given. When filtering bookings reports by date range, we assume the entire booking must be within the date range.

An ER diagram of the enterprise, specifying attributes, entity sets, relationships and primary keys.
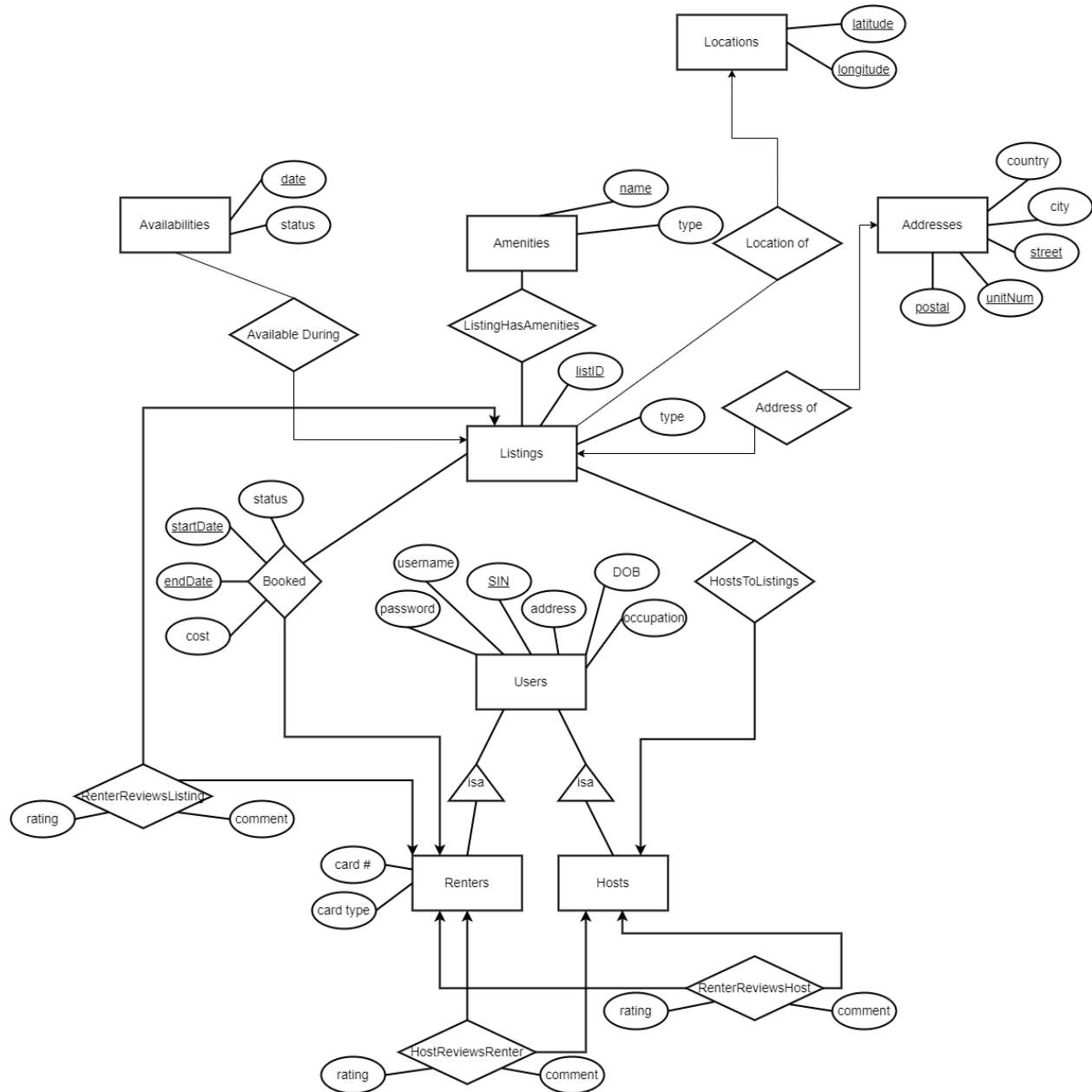
Schemas

Users(<u>SIN</u>, username, password, address, DOB, occupation)
Hosts(<u>SIN</u>)
Renters(<u>SIN</u>, cardNum, cardType)
Listings(<u>listID</u>, type)
Amenities(<u>name</u>, type)
Addresses(country, city, <u>street, postal, unitNum</u>)
LocationsOf(<u>latitude, longitude</u>)
Availabilities(<u>date</u>, status)

LocationsOf(latitude, longitude, <u>listID</u>)
HostsReviewRenters(rating, review, <u>renterSIN, hostSIN</u>)
RentersReviewHosts(rating, review, <u>renterSIN, hostSIN</u>)
RentersReviewListings(rating, review, <u>renterSIN, listID</u>)
Booked(<u>listID</u>, renterSIN, <u>startDate, endDate</u>, cost, <u>status</u>)
ListingsHaveAmenities(<u>listID, amenityName</u>)
HostsToListings(<u>listID</u>, hostSIN)

You should also include and hand in a user manual and a description of the system limitations and possibility for improvement.

Here is a sample workflow for a user on MyBnB. Upon starting the application, you are a logged out user.

<u>**As a Logged Out User**</u> the main menu provides options to:
- Sign up as a renter or a host

    - When signing up, you will be prompted to provide general information such as a unique username, password (hidden on the console for security), and SIN

    - You have the option to select if you are signing up as a renter/host. If you are a renter, you must provide additional information such as card type and number.

- Log in as a renter or a host to view options specific to the type of user

    - Provide your username and password to log in.

- View and Search through various listings, to see if you wish to sign up for/log in to MyBnB to book

Upon login, we will determine if your username and password is associated with a renter or host account.

As a Logged in Renter

- Log out or delete your account

- View, Search, or Filter through various listings

  Selecting this option will take you into **(renter) listings mode**, providing more options about listings, including:

  - Viewing all listings available on MyBnB, which provides information such as the listID, type of listing, and address

  - Check availabilities for a listing, which requires providing the ID of the listing. Use the previous options for viewing and filtering listings to retrieve the relevant ID.

  - Book a listing. Based on your queries about listings and availabilities, enter the listID and available start and end date to book a listing. If any date within the given range is not available, you cannot book.

  - Search and filter listings. There are various search options. Selecting this option will take you into **search mode**.

    Search Mode Options include:

    - Searching nearby location, which allows you to specify latitude and longitude. Enter a specific radius in km, or we default to 50km. You can choose to sort the returned results by distance, or price (in ascending or descending order)

    - Search nearby postal codes, we look at postal codes that only differ in the last letter/digit from what you have specified.

    - Find a listing based on specific address, including street name, postal, and unit # (only applicable for hotels or apartments).

    - Find listings by availabilities. Note that the entire range of availabilities entered must be available in order for a listing to show up.

    - You can sort all listings by price

    - Finally, you can filter fully by all the options mentioned above, selecting yes or no for each option to aggregate the results.

    We display results including the listID, type and address of the listing, and cost per night to provide all information you would need to decide if you wish to book.

- View and Manage Your Bookings

Selecting this option will take you into **(renter) bookings mode**, providing more options about your bookings including:

- Viewing and/or cancelling upcoming bookings. When viewing bookings (past or upcoming), we provide information including the listID, start and end dates of the booking, the address of the listing, as well as the name of the host.

  As a renter, cancelling an upcoming booking makes it available again for other renters to book.

- Viewing past bookings, and leaving reviews about your past hosts or listings. To leave a review about a host you have booked from in the past, type in their username(displayed when looking at past bookings), and provide a review and integer rating from 1-5.

- View reviews hosts have left about you, to see how you can improve as a renter.

As a Logged in Host

- Log out or delete your account

- View and manage your active listings

Selecting this option will take you into **(host) listings mode**, providing more options about listings such as:

- Viewing your listings including the listID, type and address

- Viewing and modifying availabilities for a listing, providing the listID to do so. Modifying can include adding or removing availabilities, and modifying the price of a listing during the available time slots. Note that we do not allow you to modify the price of an availability that is already booked for some time in the future. Please cancel the booking first.

- Delete a listing. This will delete all associated data for the listing such as availabilities, reviews, bookings, etc.

- View and manage your bookings

Selecting this option will take you into **(host) bookings mode**, providing options about bookings such as:

- Viewing and/or cancelling upcoming bookings. When viewing bookings (past or upcoming), we provide information including the listID, start and end dates of the booking, the address of the listing, as well as the name of the renter.

  As a host, cancelling an upcoming booking does not make it available for other renters.

- Viewing past bookings, and leaving reviews about your past renters. To leave a review about a past renter, type in their username(displayed when looking at past bookings), and provide a review and integer rating from 1-5.

- View reviews renters have left about you, to see how you can improve as a host.

- Add a Listing

MyBnB provides a host toolkit to aid in the process when you add a listing.

1. First, provide generic information about your listing such as the type (one of house, guesthouse, apartment, hotel), the location (latitude and logitude), and address.

2. We then provide our top 3 recommendations for amenities you may consider adding to your listing, based on popularity. This is based on the percentage of available listings on MyBnB that offer the amenity.

3. Based on this information, select an initial list of amenities you wish to offer (from a predetermined list).

4. Once you select amenities, we will suggest a price (per night) for your listing and explain how we have done so. We attempt to be as specific as possible, looking first at the average price per night of listings with the same type, country, city, and address, as well as amenities.

   If we cannot find listings with the exact same details as yours, we gradually decrease specificity. Note that for amenities, we look at other listings that offer at least the same amenities as you.

5. We also suggest, for each amenity that you have not included that could potentially increase the price of your listing, how much the suggested price would increase if you chose to include it. This is based on the average price per night, of listings that offer all the amenities you offer, as well as the added amenity mentioned.

6. Based on this new info, we give you the option to modify the list of amenities you offer.

7. After successfully creating a listing, you can add availabilities for it.

All Users (Logged In or Logged Out) can view reports about MyBnB.
Reports include

- Reports about Number of Bookings

Number of bookings by city (with the option to further refine by zip/postal code)
Ranking renters by number of bookings

Hosts and renters with most cancelled bookings within the current year

- Reports about Number of Listings

  Number of listings by country (with the option to further filter by city/postal)
  Ranking hosts by listings per country (with the option to further filter by city)
  Possible commercial hosts by country/city (hosts with > 10% of listings in that
  country/city)

- Word cloud

  A noun phrase frequency chart is displayed for each Listing that has at least 1 review.
  Only the noun phrases that appear more than once for that Listing are displayed and in
  order of frequency.

General Tips and Tricks:

To exit from any mode (such as listings, search or bookings mode), simply type 0 to exit and
return to the main menu. Type 0 from the main menu to exit the application.

System Limitations

- A user can only leave a review for another particular user or listing only once. For
  example, if a Renter reviews a Host, then rents from them again, the Renter cannot
  update their Rating/Commentor or leave a new review.

  Possible Improvement: Associate a review not only with a Host and Renter or Listing
  and Renter but also a booking. That way, a Host or a Renter will be able to leave new
  reviews about their new experiences.

- In the Booked table, the key is listID, startDate, endDate, status. This means there can
  be at most one row with status 'canceled' for the same listID, startDate, endDate. So, if a
  particular listing was booked, then canceled, then booked and so on with the same
  startDate and endDate, only information about the first cancellation is recorded. This can
  warp our Max Cancellation reports because it shows that the listing was cancelled only
  once even if it was cancelled multiple times and the cancellation does not reflect on any
  Renters other than the first one who cancelled.

  Possible Improvement: make RenterSIN also part of the key so that the different
  Renters cancelling the same listing during the same date range is accurately
  represented in the reports.

- To accurately determine the increase of the suggested price using amenities the sample
  data would have to be very large with many iterations of the possible amenities for all 4

Listing types in many different streets/ postal codes/cities/countries. This means we had to simplify our suggestion in certain scenarios.
- When a host is adding a listing, the suggested price is based on the amenities they have added. However, if they add no amenities, the suggested price is calculated based on other similar listings (ex. In the same postal code, city, and/or type of listing) and does not take into account that the new listing doesn't have any amenities.
- We suggest a price based on Listings that have all of the amenities the new Listing has but these listings may have other amenities as well, making the suggested price slightly higher than the average for Listings that have only the chosen amenities.

Possible Improvement: Many more data points with multiple Listings for the each combination of the amenities. Then, we would refine our Suggest Price queries to only include information about Listings that have exactly the same amenities.