Airbnb Report

CS 4347.002

Team #6

Alexandria Andrade, Lucas Noack, Joshua Brown

STEP 1

Low level Overview of Functional Data Requirements

Entities:

* Property
* Reservation
* PropertyCategory
* User
* Host
* Amenities
* Reviews

Attributes:

* Property: Beds, Bathrooms, Max\_guests, Nightly\_fee, Cleaning\_fee, Address, Host\_account#, Property\_type
* Reservation: Reservation#, CheckInDate, CheckOutDate, Cost, Address, Renter\_Acc\_No, Host\_Acc\_No
* PropertyCategory: CategoryLabel, CategoryDesc
* Amenities: Name, Description
* User: username, Account#, Email, FName, Lname
* Host: user\_Acc\_No, rating, SuperHost, languages
* Review: Reservation#, Title, content, rating, date\_written

Relationships:

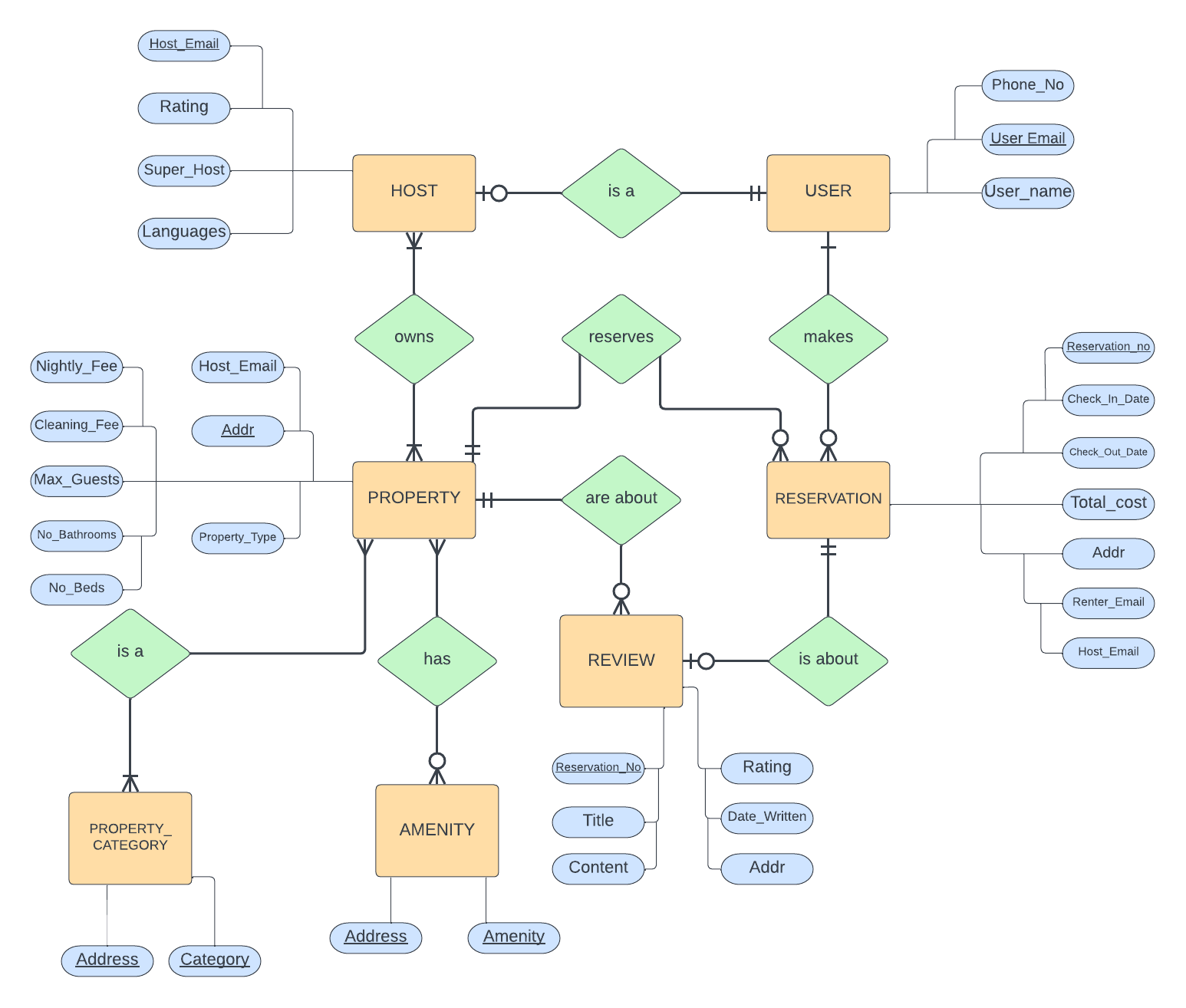
* Many Hosts can have Many Properties
* One User can make Many Reservations
* Many Properties can have Many PropertyCategories
* Many Properties can have Many Amenities
* One Property can have many reservations
* One Property has many reviews
* One Review has one Reservation
* One Host is one user

High Level Overview of the functional data requirements.

* Airbnb has different PROPERTY listings for a USER to reserve through a RESERVATION. Each PROPERTY specifies the number of Beds, Bathrooms, and Max\_guests allowed to stay at the property. Each PROPERTY can also have a Cleaning\_fee and/or Nightly\_fee amount that is charged for staying here. A unique address is used to identify each PROPERTY. A PROPERTY is always owned by one or more HOSTs and displays the Host\_account# for owner identification under a PROPERTY listing. Lastly, a PROPERTY has a Property\_Type (House, Apartment, GuestHouse).
* A PROPERTY is considered to be one or more different PROPERTY\_CATEGORYs and a PROPERTY\_CATEGORY will have several different properties. A PROPERTY\_CATEGORY has a Description of the category and unique Label (Lakefront, Cabins, OMG!, Luxe, etc). PROPERTY\_CATEGORY is different from a property type in that categories are based on several different aspects about the property like its location, popularity, aesthetics, etc, whereas a property type identifies the actual structure of the property (house, apartment, etc.).
* A PROPERTY may or may not have an AMENITY or it may have several different ones. An AMENITY has some Description of what the amenity is and a different Name to identify each AMENITY.
* To stay at a PROPERTY, or list your PROPERTY on Airbnb, a person must sign up and be an official USER. A USER will always have a name (Fname, Lname), Email, and a Phone Number  associated with them. A USER can also be a HOST that owns PROPERTY. If a USER wants to stay at a PROPERTY on airbnb, they must make a RESERVATION.
* A HOST is always a USER that will always own one or more properties listed on Airbnb. You must be a HOST to list a property on airbnb (a USER cannot list PROPERTY if they are not also a HOST). Additionally, a host may or may not have a Rating or be a SuperHost. A Host will list the Languages they speak.
* To reserve a PROPERTY, a USER must make a RESERVATION. USERs can make multiple RESERVATIONs to reserve properties. A PROPERTY can have several RESERVATIONs, but a RESERVATION made is only to reserve one PROPERTY. Every reservation has a Check\_in\_date and Check\_out\_date available based on whether or not there already exists a reservation for those dates. The Total\_cost of the stay, the Host Email Address, and Address of the property being reserved are also part of the reservation. Finally, every RESERVATION has a different Reservation\_no. For every reservation, there may or may not be a REVIEW made.
* A REVIEW can be made only if a RESERVATION is made. The REVIEW is about the PROPERTY stayed at during the RESERVATION. A REVIEW is made up of a Title, Content, and Rating, and the date the review was written (Date\_written) and the Address of the property. Each review also has a reservation\_no since a review can only be made about a RESERVATION.

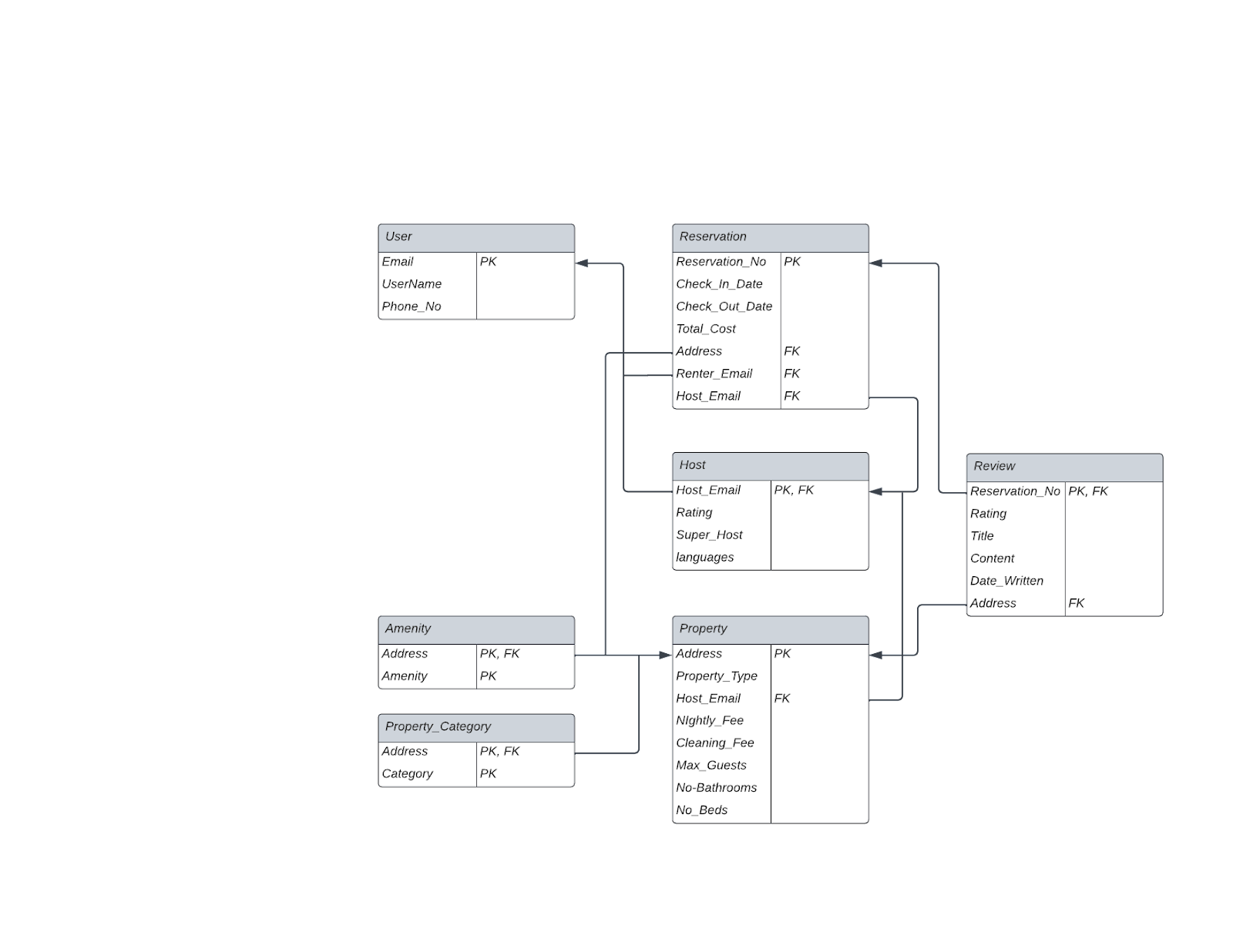
STEP 2

ER Diagram



STEP 3

Initial Relational Schema

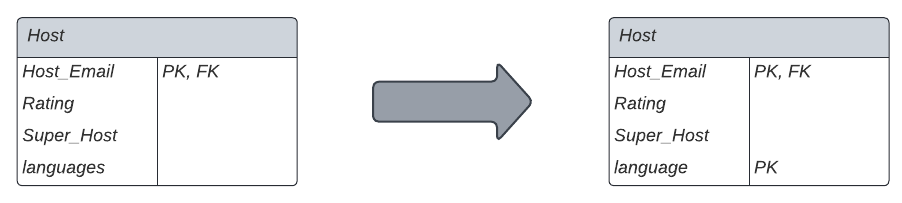


STEP 4

Normalization

First Normal Form

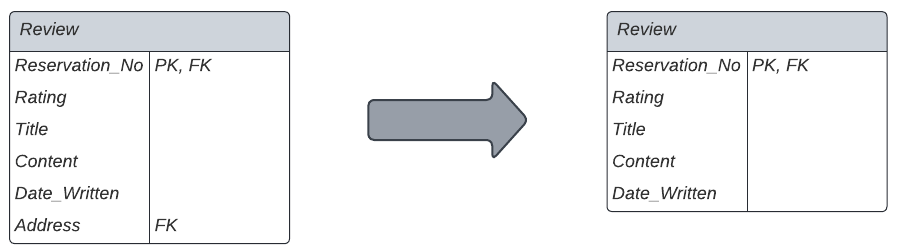
* The Languages attribute in the Host table is a composite attribute. Changing it to ‘language’ and making it a primary key will put Host into 1NF.



* All other tables are in 1NF.

Second Normal Form

* The Address attribute is not functionally dependent on the primary key of ‘Review’. This attribute is unnecessary in this table so removing it will put the table into 2NF.



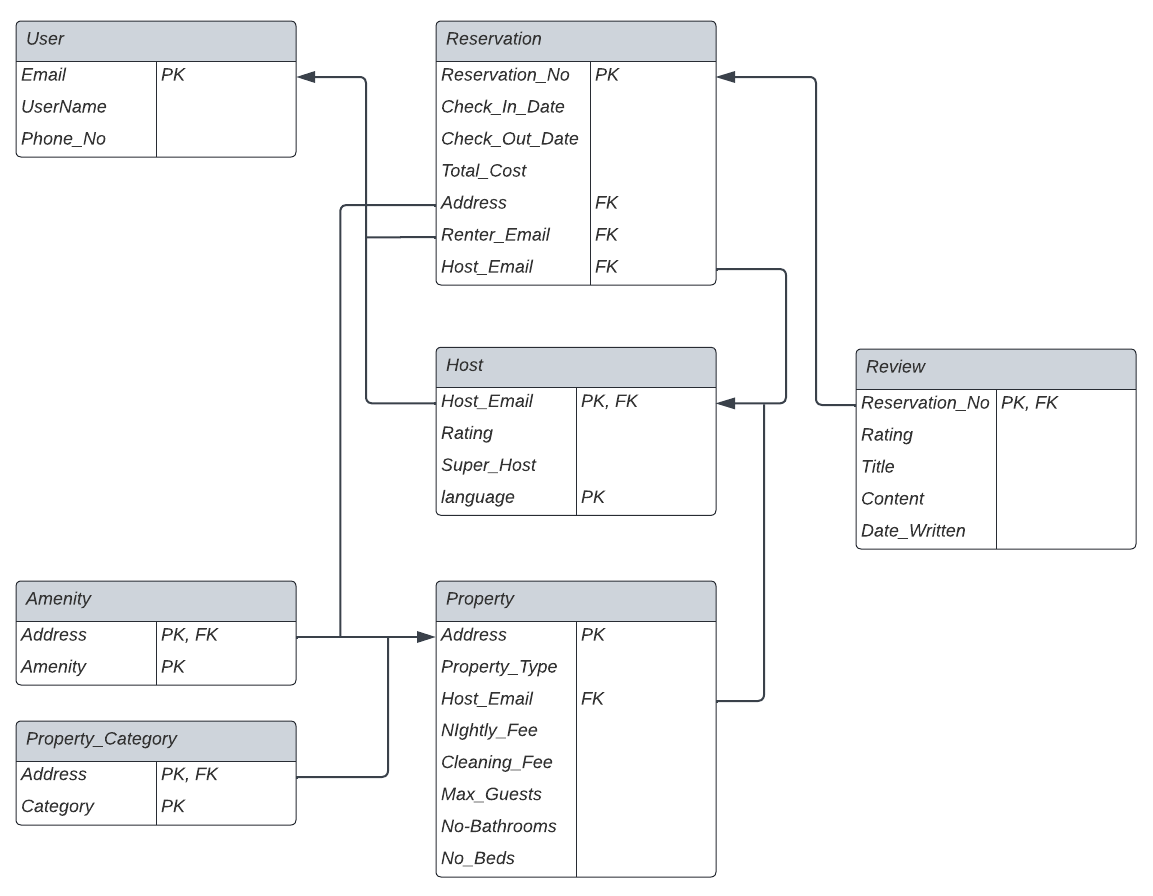
* All other tables are in 2NF.

Third Normal Form

* All tables are in 3NF.

STEP 5

Initial Relational Schema



STEP 6

SQL Commands

**create schema airbnb;**

**use airbnb;**

**CREATE TABLE User (**

**Email    varchar(50),**

**UserName      varchar(25) not null,**

**Phone\_No char(10) not null,**

**PRIMARY KEY (Email));**

**CREATE TABLE Property (**

**Address    varchar(50),**

**Property\_Type      varchar(25),**

**Host\_Email   varchar(50),**

**Nightly\_Fee    float,**

**Cleaning\_Fee      float,**

**Max\_Guests      int,**

**No\_Bathrooms      int,**

**No\_Beds      int,**

**PRIMARY KEY (Address)**

**);**

**CREATE TABLE Host (**

**Host\_Email  varchar(50),**

**Rating      decimal(1,1) default 0,**

**Super\_Host   bool  default false,**

**Language    varchar(25),**

**PRIMARY KEY (Host\_Email, Language),**

**FOREIGN KEY (Host\_Email) REFERENCES User (Email)**

**ON UPDATE CASCADE ON DELETE CASCADE);**

**CREATE TABLE Reservation (**

**Reservation\_No    char(15),**

**Check\_In\_Date      date,**

**Check\_Out\_Date      date,**

**Total\_Cost   float,**

**Address      varchar(50),**

**Renter\_Email      varchar(50),**

**Host\_Email       varchar(50),**

**PRIMARY KEY (Reservation\_No),**

**FOREIGN KEY (Renter\_Email) REFERENCES User (Email)**

**ON UPDATE CASCADE ON DELETE CASCADE,**

**FOREIGN KEY (Host\_Email) REFERENCES Host (Host\_Email)**

**ON UPDATE CASCADE ON DELETE CASCADE);**

**CREATE TABLE Review (**

**Reservation\_No    char(15),**

**Rating      decimal(1,1) default 0,**

**Title   varchar(20)  not null,**

**Content    varchar(1000) not null,**

**Date\_Written      date not null,**

**PRIMARY KEY (Reservation\_No),**

**FOREIGN KEY (Reservation\_No) REFERENCES Reservation (Reservation\_No) ON DELETE CASCADE**

**);**

**CREATE TABLE Property\_Category (**

**Address    varchar(50),**

**Category varchar(25),**

**PRIMARY KEY (Address, Category),**

**FOREIGN KEY (Address) REFERENCES Property (Address)**

**ON DELETE CASCADE ON UPDATE CASCADE);**

**CREATE TABLE Amenity (**

**Address    varchar(50),**

**Amenity varchar(25),**

**PRIMARY KEY (Address, Amenity),**

**FOREIGN KEY (Address) REFERENCES Property (Address)**

**ON DELETE CASCADE ON UPDATE CASCADE);**

**ALTER TABLE Property**

**ADD FOREIGN KEY (Host\_Email) REFERENCES Host (Host\_Email) ON DELETE CASCADE ON UPDATE CASCADE;**