

account = (email, fname, lname, password)

client = (email[fk21], phone\_number)

fk21: email -> account.email

admin = (email[fk2])

fk2: email -> account.email

customer = (email[fk22], credit\_card, cvv, exp\_date, current\_location)

fk22: email -> client.email

owner = (email[fk3])

fk3: email -> client.email

airline = (name, rating)

airport = (airport\_id, name, time\_zone, street, city, state, zip)

flight = (airline\_name[fk1], flight\_num, departure\_time, arrival\_time, date, cost\_per\_seat, capacity, from\_airport[fk5], to\_airport[fk6])

fk1: airline\_name -> airline.name

fk5: from\_airport -> airport.airport\_id

fk6: to\_airport -> airport.airport\_id

property = (owner\_email[fk4], property\_name, description, street, city, state, zip, cost\_per\_night\_per\_person, capacity)

fk4: owner\_email -> owner.email

isCloseTo = (airport\_id[fk7], owner\_email, property\_name[fk8], distance)

fk7: airport\_id -> airport.airport\_id

fk8: (owner\_email, property\_name) -> (property.owner\_email, property.property\_name)

reserve = (customer\_email[fk9], owner\_email, property\_name[fk10], start\_date, end\_date, num\_guests)

fk9: customer\_email -> customer.email

fk10: (owner\_email, property\_name) -> (property.owner\_email, property.property\_name)

review = (customer\_email#[fk11], owner\_email, property\_name[fk12], content, score)

fk11: customer\_email -> customer.email

fk12: (owner\_email, property\_name) -> (property.owner\_email, property.property\_name)

book = (customer\_email[fk13], airline\_name, flight\_num[fk14], num\_seats)

fk13: customer\_email -> customer.email

fk14: (airline\_name, flight\_num -> (flight.airline\_name, flight.flight\_num)

customerRateOwner = (owner\_email[fk15], customer\_email[fk16], score)

fk15: owner\_email -> owner.email

fk16: customer\_email -> customer.email

ownerRateCustomer = (owner\_email[fk17], customer\_email[fk18], score)

fk17: owner\_email -> owner.email

fk18: customer\_email -> customer.email

amenities = (owner\_email, property\_name[fk19], amenity\_name)

fk19: (owner\_email, property\_name) -> (property.owner\_email, property.property\_name)

attractions = (airport\_id[fk20], attraction\_name)

fk20: airport\_id -> airport.airport\_id

### Unhandled Constraints:

- Every flight must come from one airport and go to one airport.
- Every flight must be associated with one airline.
- Every property must have an owner.
- Customers can only book flights/properties that have enough capacity for them(ie, capacity > guests). We will be able to do this with a check constraint.
- We did not handle derived attributes such as average rating, seats left, total cost for property, total cost for flight, cancellation fee for property, cancellation fee for flight, and total income for owner.
- Additionally, cancelation fee is a derived attribute. The cancelation fee states that the customer is charged 20 percent of the flight booking, but the rest of the money is returned to them. We did not handle the refund of money if the customer cancels a flight due to it being part of a derived attribute.
- We did not handle the ability to track the total income of an owner up to a given date, due to this being a derived attribute.

### Assumptions

- We are assuming that credit card information has to be not null as every customer needs to input their credit card number, cvv, expiration date in order to book flights and reserve properties.
- Since the airline ratings are pre-populated into the database we are assuming they can be null since the third party resource might not have ratings according to the project description.
- We are assuming that departure time, arrival time, date, cost per seat, and capacity should be not null since customers book their flights based on this flight information.
- We are assuming that when the project description states that the attribute should be recorded, this attribute should be not null. An example of this is the description stating that the system should store the current location of every customer.

- The description does not explicitly state that the time zone will be given in three characters, but we have made an assumption that the time zone will be three characters based on looking at the data.
- We are assuming that the distance between airport and property is not null, since this distance will help customers better plan their trip.

