Database Design Spec

Fall 2021

Objective: Below is an overview of a **hotel reservation system**. Please review the processes below to gain an understanding of the organizational context for which we are designing this database

Background: The Sour Apple Hotel, a South Austin boutique hotel, acquired funding a year ago has already begun expanded its operations into a diverse set of properties. Not having a single source for storing its data, it is now ready to build a single centralize reservation system that will manage all its diverse locations, customers, and reservations. Presently, they're managing each of their three locations' data separately and having to merge it all in QuickBooks for accounting/reporting, and this is not sustainable especially if they plan to keep expanding. Their current locations include their original hotel location on South Congress, the East 7th Lofts, and their cabins near the Balcones Canyonlands which are actually west of Austin in the city, Marble Falls. Like any other hotel, you can reserve a single room to stay or a "block" of rooms. Locations can share similar features like "Free Breakfast", but due to their diverse nature, can have features unique to that specific location. The system you make should be able to handle all key data such as the various locations and the customer reservations of rooms at these locations. Once this database is built, it will support internal operations, and the plan will be to integrate it to their website to allow for online viewing of room features but also customer account setup and room reservations.

Customer Account Signup process: When a new customer is reserving a room or checking in, the Sour Apple staff will setup an account for that person. This requires a first and last name, email, address, and a single phone number. This account is used to automatically track their stay credits earned. Each night stay (per room) earns a single credit. 10 credits can be redeemed for a free night's stay in one room. The team tracks total credits earned and credits used to understand how many credits a person has left to use currently while retaining a simple history of how many they've earned in a lifetime. Each customer will have a single credit card on file since Sour Apple keeps things simple and doesn't want to track multiple payments for customers.

Room Reservation process: When a customer wants to make a reservation currently, it's done by calling the location they want to stay with or making the reservation at the front desk of that location. Once Sour Apple has a centralized system, their hope is that any location can take reservations for another location or you can reserve rooms at any location online. When a customer requests a reservation their customer_id is tied to the reservation if it's known. If it's a new customer, staff creates a customer account first and then assigns the id. A check-in date and anticipated check-out date are captured and then a number of rooms is requested to figure out if there are enough rooms at the location to fulfill the reservation. If they can't fulfill the reservation we won't create it. If the customer has a discount code, that can be applied to the reservation at that time otherwise it's left blank. Once the reservation is created and the rooms are attached to it, a Confirmation number is randomly generated and stored on the reservation along with the date it was created. A single character status is also assigned to track what status the reservation is in (see more detail below on this). For each room on the reservation, the number of anticipated guests is logged since each room has a max capacity. After the stay, the customer is asked to provide a rating of 1 to 5 stars that can be captured and stored on the reservation if they complete the rating survey.

Technical Details: NOTE: Each of these could be separate entities.

- **Customers** – Sour Apple also wants to track birthdates of customers so they can email them a birthday card and discount code each year. On top of the personal info and contact details, each patron should have their one and only primary credit card assigned to their account but note that the team wants to store this credit card info in a separate table to allow for it to be encrypted and better secured against unwanted access. To process a credit card, you must have the person's full name, including middle if they have one. You then provide the card type which is a 4-character code like "VISA", "MSTR" for MasterCard, or "AMEX" for American Express. You also capture the card number which is 15-16 digits, the expiration date, the 3-digit code on the back called the CC_ID, and the full billing address attached to that card since it doesn't always have to match the mailing address attached to the customer's account.

- **Location** – We want to track all the locations with a specific location_ID and name and allow for the company to scale up to more locations or expand the number of rooms for each location if they choose to build on later. We will need to track the full address, phone, and URL of the location since they will drive search on the website and reporting for management. Each location has a set of features that can either be unique or shared between locations such as “Free Wi-Fi”, “Free Breakfast”, “Health Center”, etc. A location can have a number of features. Features should be assigned just their own ID and name. The system should allow adding new features. Each location has a set number of rooms.
- **Reservations** – Once the reservation is made, a reservation total is saved on the reservation which is the number of nights times the rate of each room added up. The confirmation number has to be unique to easily find a reservation. It’s a random collection of numbers and letters and looks something like “G2JD8J3”. The status that is attached to the reservation is one of the following preset single characters codes (e.g. U for *upcoming*, I for *in progress*, C for *completed*, and N for *no show*, R for *refunded*). Discount codes are usually a max of 12 characters and are a combo of numbers and letters. Customer rating is just a number of 1 to 5 so we can capture averages over time by location and customer. For each reservation we also will track a “notes” column that is not to exceed 500 characters that can be used to track random things like customer requests or special celebrations and more.
- **Rooms** – Each room will be assigned a “floor” even if the location has only one floor in case that location adds another story latter. Each room gets a room number of that starts with 1 if it’s on the first floor, 2 if it’s on the 2nd floor, etc. Rooms are assigned a single character code to designate the type of room it is like (D for double beds, Q for single queen, K for single king, S for suite that has two rooms and some form of kitchen, C for cabin). This can expand as needed. We also need to track the square footage of the room and a max number of people allowed in the room. Sour Apple has chosen to keep their pricing the same all year round including gamedays and holidays so each room has a weekday and weekend rate.

Deliverable: The database model should include table names, field names, and identify which fields are primary and foreign keys. Relationships between tables should use the Crow’s Foot method.

Formatting Tips/Requirements:

- When you connect your tables, be sure that the relationship lines are attached and connecting to the specific columns on the table. This way, you are showing that a PK or FK on one table is connected to corresponding PK/FK on the other table. The feet shouldn't be touching random parts of the table.
- When ordering the columns in your entities, it's best to list the PKs as the first columns, the FKs as the next ones, and then the rest of your non-key columns below that. This keeps it neat and makes it easy to quickly identify what the table is about.
- Remember that a column can be an FK and also part of a composite PK. If that’s the case, you can use "PK,FK" for any columns that are both part of a composite PK and are also an FK.