Hotel Management Database System Design Document

Database Specification: Purpose, Business Problems Addressed and Business Rules

Submitted by Team - 7:

Parth Shah, Aditi Krishna, Stuti Singh, Abhiram Desai, Shiwangani Jaiswal

Database Purpose:

The purpose of the database is to maintain the data used to track and report on hotel management activities for an international hotel chain. It will be used by the hotel's administrative staff only and will not duplicate information.

Business Problems Addressed:

- Allow a hotel's departments to generate reports on customer, employees, hotel locations, event reservations etc.
- Provide information to enhance or improve marketing efforts to specific targets like room bookings and restaurant bookings, and keeping a track of customer details.
- Gain insight for revenue/budgeting reports based on customer habits and seasons (via dates booked)
- Allow financial employees to look at investors, their status based on investment and offer benefits and profits accordingly
- Permit inventory management department and facilities department to maintain inventory and coordinate with suppliers
- Allow Chains to keep track of the performance of their hotels
- Let ChainHotels maintain a record of rooms, amenities, restaurants and their booking data

Business Rules:

- Each parent organization (chain) will have one or many hotels
- Each parent organization (chain) will one or many investors
- Each parent organization (chain) will one or many employees
- Each investor can invest in multiple chain
- Each hotel will have one or many rooms
- Each hotel will have zero or many restaurants
- Each hotel will have one or many suppliers
- Each hotel will have zero or many amenities
- Each hotel will have one or many employees
- Each hotel will have zero or many event reservations
- Each hotel will have zero or many customer feedback (optional to all staying customer)
- Each employee will have one department they are working for, however each department will have one or many employees

- Each customer will have zero or one membership
- Each customer will have zero or many room reservations
- Each customer will have zero or many restaurant reservations
- Each customer will have zero or many event reservations
- Each customer will have one or many customer feedback (for each stay)
- Each event reservation will only have one event invoice
- Each event reservation will have only zero or more amenities
- Each room will have zero or many room reservations for a given date
- Each reservation will have one or many roo
- Each restaurant will have zero or many restaurant reservations
- Each department has one or many employees
- Each room reservation will generate one invoice
- Each supplier only supplies to one hotel under the chain
- Each customer should be above 18 years of age
- Each employee should be above 18 years of age
- All CheckOutDate is greater than CheckInDate
- All room prices are greater than 0
- All AmenityCharges are greater than 0
- Number of guests in an EventReservation should be less than 1000
- All MembershipStartDate should be greater than MembershipEndDate

ERD Design Requirements (Credit to Professor Simon Wang)

- Using Crow's Foot notation
- Specify primary key and foreign keys fields in each table by specifying PK and FK beside the field name
- Specify identifying and non-identifying relationships between entity tables by drawing dotted or solid lines
- Specify multiplicity between entities

Design Decisions:

Entity Name	Why Entity is Included	How the Entity Related to Other Entities
Chain	The chain entity includes information about the parent organization which has several hotels under their belt. It contains information about the organization, such as, name, address, type of hotel chain (luxury, premium, etc.) and website details, to name a few.	The chain, or the parent organization has a one-to-many relationship with ChainHotel, Employee, and Investor entities. We ensure a business rule where only each investor only invests in one Chain, however, one Chain can have many investors. The Chain would then decide how to distribute

		the investment amongst their different hotels.
ChainHotel	The primary purpose of the database is to track and store information corresponding to all hotels under the parent organization. The entity contains information about the essential attributes of a specific hotel, such as floor count, room count, etc.	As the core entity in the database, the ChainHotel entity key, hotelID, relates it to Chain, Room, Supplier, Employee, Restaurant, Amenity, EventReservation, and CustomerFeedback. These help us gain insight. There are one-to-many relationships with these entities.
Room	Contains information about the room booked corresponding to hotel ID (based on what hotel the room is booked in) along with other information like room number, price, capacity, type and whether or not it is pet friendly.	The Room entity is related to the ChainHotel and the RoomReservation entities as a crucial factor in room reservation. The Room entity is related to the ChainHotel entity through a many-to-one relationship. One Hotel can have many rooms in it. It is also related to the RoomReservation entity through a one-to-many relationship. One room can have multiple reservations from different customers on different check in and check out dates.
Restaurant	Stores information about one or many restaurants within a hotel.	The restaurant entity is associated with the ChainHotel and RestaurantReservation entities. A Restaurant can have zero or many reservations for a given date.
Amenity	Stores information about one or many amenities offered by a hotel.	The amenity entity is related to the ChainHotel and EventReservation tables. A hotel can have zero or many amenities they offer. Similarly, for each EventReservation, the customer can request for certain amenities to be included with their reservation.
Customer	Includes information about customers who make reservations at hotels/restaurants.	The customer entity is a very crucial entity to the database. It is related to Membership, RestaurantReservation, RoomReservation, EventReservation and CustomerFeedback. There's a one-to-zero Or Many relationship

		between the Customer entity and RestaurantReservation, RoomReservation, EventReservation and CustomerFeedback entities, while the Customer and Membership entity have a one-to-zero or one relationship. This is because a customer who is a Marriott member would have access to all chains under the parent company.
Membership	Each customer has a membership with the hotel, whose details are stored in the membership entity.	The membership entity has a one-to-one relationship with the customer. This entity stores points and other membership specific details.
Employee	Contains information about employees who are in charge of tethering to customers' needs by making reservations for them at hotels/restaurants.	The Employee entity is related to the Chain, ChainHotel and Department entities through one-to-many relationships, with the Employee taking the many-side in all three relationships. One Chain, one department and one hotel can have multiple employees working but one Employee cannot be working in more than one chain, hotel or department, hence the one-to-many relationships.
Department	As we are managing a huge company which has more than one chain, it is generous enough to imagine there will be an army of work force which will be required and this army will be belonging to different departments which is why this entity is important	Department is connected to the employees entity to one to many relationships. As here a employee can belong to only one department but a department can have several employees
RoomReservation	Room reservation is needed to store data of customers who are booking the room, and also to get information on how many rooms are booked.	Room reservation will only have one and only one invoice, a room can only have one reservation at a time.
RoomBooked	Associative entity for RoomReservation and Amenity entities as this has many to many relationship	This entity has one or many relationship with the RoomReservation table and Room

		table, since one reservation can have multiple room bookings
ReservationHasAmenity	Associative entity for room reservation and amenity entities as this has many to many relationship between them	This entity has zero or many relationship with the Amenity and RoomReservation table
RestaurantReservation	Restaurant reservation is needed to collect data of customers dining in that restaurant.	Customers can have zero or many reservations, a restaurant can have zero or many restaurant reservations.
EventReservation	Stores event specific reservation information for booking hotel rooms/conference rooms, event type, catering, etc.	EventReservation entity has a one-to-one relationship with EventInvoice as each reservation will have one Invoice. It has a one-to-many relationship with Amenity as many amenities could be used during one event. ChainHotel and Customer have a one to one relationship as each ChainHotel can have many events and the same with Customer entities.
EventHasAmenity	Associative entity for EventReservation and Amenity entities as this has many to many relationship	This entity has zero or many relationship with the Amenity and EventReservation table
Invoice	Hotel is created to generate revenue and when a customer books a restaurant or a room all the services are paid. So to track the profits and money we have the invoice entity where we store the bill generated by the customer	The entity invoice is a very important one and is connected to the room reservation with one to one relationship! As here the assumption is that a room reservation will only have one invoice which will be generated.
EventInvoice	Tracks billing details for event specific transactions	Event Invoice has a one to one relationship with an event reservation
CustomerFeedback	It is important for a business to know their service was and to improve in future we are storing customer feedback data in this entity	It is important for a hotel to get the customer feedback but a customer has an option to give it or not. So this entity is a non identifying relationship with many to the customer table

Supplier	Stores supplier information who are responsible for fulfilling the inventory needs that are essential to the hotel	Each ChainHotel entity has a one-to-many relationship with suppliers. It is a non-identifying relationship.
Investor	Stores high level information about the parent organization investors and their involvement status (i.e., active or inactive investor)	Each Chain entity has a one to many relationship with Investors. To avoid a many-to-many relationship for this assignment, we have a business rule that an investor can only invest in one chain.
ChainHasInvestor	Associative entity for investor and chain entities and this has many to many relationship	This entity is related to chain with one or many relationship and with the investor with one to many relationship