Data & Applications

Project Phase - 1

GROUP: 4

Team Members:

1. ***Malla Sailesh (2021101106)***
2. ***Manohar Naga(2021101128)***
3. ***Naveen Chekkapalli (2021101025)***

Introduction to Mini World :-

This is a mini world that stores the data about the availability of rooms in hotels and the details of the hotels such as the rating in a particular city (say hyderabad).

Purpose :-

This is to check if there are any rooms available in the nearby hotels . If parents came to see their son/daughter who is studying in college located in a city different from where parents stay and if they wanted to stay in that city that day . So to check availability of rooms in nearby hotels for staying this would be helpful .

Not only this if anyone comes to a particular city and wants to stay in the hotel for a few days etc then this would also be useful .

Users :-

1.Customers/Guests (who need room to stay) - Naïve/parametric End User

2.Employees in the Hotel :-the hotel manager/staff whose contact number is also stored in the database to contact him about asking for a room , price for room etc in their hotel .

3.Database Administrators :- who manage this database and are responsible for security breaches etc.,

4. Database designers :- who knows which data to be stored :- storing the contact number of hotel manager/staff .

What will the users do with the database?

1.Customers:- They can find a hotel with all the desired requirements in the price range they want.

2.Employees(Manager) :- their contact number is stored in the database so that customers can contact them about the facilities provided , room availability , price etc .

3.Database Administrator :- they manage the database and have the authorizing access to the database . (they only decide to update the features of the database etc).

4.Database Designers :- responsible for identifying the data to be stored in the Database, interact with potential group of users and develop views of the Database .

Applications :-

1. Hotel Customer gets desired hotels: Based on the Need a customer can get his desired nearby hotel with best requirements that fit him

- Pricing,Rating of the Hotel,Rooms Available,Location,Services Provided , AC/NON-ACetc.,

2. Hotel Owner/Manager can make profit: We also provide the option of reviews/rating on the Services they provide the customers in the Hotel so that they can improve their services.

3. Ease of Booking Hotels: Anyone can easily search for a hotel in their nearby location which the customer wants .

4. Customers Need not Physically go to each hotel and check whether the rooms are free. Instead he can directly go to the hotel by checking the details regarding it through our database .

5. Customers can get filters and other options to set Price range,amenities,Location etc.

Database Requirements :-

Strong Entity Types:

- ***Customer :-***
  - ***Name (Composite Attribute)***
  - ***Belong to which City/State/Country (Composite Attribute)***
  - ***Phone Number***
  - ***Date of Birth***
  - ***Email ID***
  - ***Age (Derived Attribute)***
  - ***Gender***
- ***Hotel :-***
  - ***Name (Composite Attribute)***
  - ***Location (Composite Attribute)***
  - ***Max number of rooms***
  - ***Facilities Provided (Multi Valued Attribute)***
- ***Room :-***
  - ***Hotel Name (Composite Attribute)***
  - ***Hotel Location (Composite Attribute)***
  - ***Room Number***
  - ***Max number of persons per Room***
  - ***Price per Day***
- ***Booking Status :-***
  - ***Hotel Name (Composite Attribute)***
  - ***Hotel Location(Composite Attribute)***
  - ***Room Number***

- is_Booked

- Employees :-
  - Name(Composite Attribute)
  - Hotel Name (Composite Attribute)
  - Hotel Location (Composite Attribute)
  - Employee Id
  - Contact Number
  - Role in the hotel

Weak Entity Types:-

- Room Type (Related To Room) :-
  - Room Number
  - Hotel Name (Composite Attribute)
  - AC/Non-Ac
  - Max Number of people accommodated
  - With/ Without Tv
- Reviews/Rating (Related to Hotel) :-
  - Room Rating
  - Food Rating
  - Service Rating
  - Hotel Name (Composite Attribute)

Sub Class:

- Employees(Strong Entity Type):-
  - Manager
  - Receptionist
  - Worker

Primary Keys :-

- Customer :- Email Id

- *Hotel :- Name + Location (i.e both Name , Location together is a primary Key)*
- *Room :- Hotel Name + Hotel Location + Room Number*
- *Booking Status :- Hotel Name + Hotel Location + Room Number*
- *Employees :- Hotel Name + Hotel Location + Employee Id*

Partial Key :-

- *Room Type :- Hotel Name + Room Number (referencing to Room Entity Type :- which is a strong entity type via identifying relationship type) .*
- *Reviews/Rating :- Hotel name (referencing to Hotel Entity Type :- which is a strong entity type via identifying relationship type).*

Foreign Key :-

- *Hotel Name in Room Entity Type is referencing to Name in Hotel Entity Type*
- *Hotel Name in Booking Status Entity Type is referencing to Name in Hotel Entity Type*
- *Hotel Name in Employee Entity Type is referencing to Name in Hotel Entity Type*
- *Room Number in Booking Status Entity Type is referencing to Room Number in Room Entity Type*
- *Hotel Name in Reviews/Rating Entity Type is referencing to Name in Hotel Entity Type*
- *Hotel Name in Room Type Entity Type is referencing to Name in Hotel Entity Type*

- Room Number in Room Type Entity Type is referencing to Room Number in Room Entity Type

Relationship Types :-

- goes_to :- Customer goes_to hotel
  - Degree - 2 (binary)
  - Participating Entities - Customer , Hotel
  - Cardinality Ratio :- m:n (because customer can go to any hotel)
  - Min-Max Ratio :-
    - Customer - (0,N)
    - Hotel - (0,N)
- stays_in :- Customer stays_in Room
  - Degree - 2(binary)
  - Participating Entities - Customer , Room
  - Cardinality ratio :- m:n
  - Min-Max Ratio :-
    - customer - (0,N)
    - room - (0,N)
- has :- hotel has rooms
  - Degree - 2(binary)
  - Participating Entities - Hotel , Room
  - Cardinality Ratio - 1:N
  - Min-Max Ratio :-
    - Hotel - (1,N)
    - Room - (1,1)
- has_1 :- Room has_1 Booking Status

- *Degree - 2 (binary)*
- *Participating Entities - Room , Booking Status*
- *Cardinality Ratio - 1:1*
- *Min-Max Ratio :-*
  - *Room - (1,1)*
  - *Booking Status - (1,1)*

➢ *managing :- Employee(Manager:- supervisor)* **managing** *Employee(other than Manager:-supervisee)*

- *Degree - 1*
- *Participating Entities - Employee*
- *Cardinality Ratio - N:1*
- *Min-Max Ratio :-*
  - *Employee(supervisee) :- (1,1)*
  - *Employee(supervisor) :- (1,N)*

IDENTIFYING RELATIONSHIP TYPES :-

➢1 *is_of :- Room* **is_of** *Room Type*

- *Degree - 2 (binary)*
- *Participating Entities - Room , Room Type(Weak Entity Type)*
- *Cardinality Ratio - N:1*
- *Min-Max Ratio :-*
  - *Room - (1,1)*
  - *Room Type - (1,N)*

➢2 *get :- Hotel* **get** *Reviews/Rating*

- Degree - 2 (binary)
- Participating Entities - Hotel , Reviews/Rating
- Cardinality Ratio - 1:N
- Min-Max Ratio :-
  - Hotel - (1,N)
  - Reviews/Rating - (1,1)

Degree >3 Relationship Type :-

➢ Manages :- Employee **Manages** Rooms in a Hotel to satisfy Customer needs

- Degree :- 4(Quaternary)
- Participating Entities - Employee , Room , Hotel , Customers
- Cardinality Ratio - 1:N:1:M
- Participation Constraint :-
  - Employee manages N Rooms in a Hotel
  - Employee manages rooms in only 1 Hotel at max
  - Employee manages N Rooms in a Hotel to satisfy M Customer needs .

Functional Requirements :-

1. Retrieval :-

   a. Selection :-

      i. Select the Hotels which are in the nearby(100mtr) location of what we enter .

ii. Select the Hotels which have a rating>3.5 and price lesser than 2000 per day .

b. Projection :-

i. List of Service Rating for all the Hotels

ii. List of Hotels which have Rooms with AC

c. Aggregate :-

i. Average Rating of the Hotel given for Food

ii. Minimum/Maximum Rating of the Hotel given for Service

iii. Minimum/Maximum Price of the Room in each Hotel .

*iv. SUM rating of a hotel calculated as SUM = (Food Rating + 2*Service Rating + 3*Room Rating).Then give the rating of the Hotel by dividing it with 6 to get AVG rating of the Hotel.*

d. Search :-

i. Search for the Hotel Name starting with the letter 'L'.

ii. Search for Hotels whose location has the word "banjara hills" in it.

2. Analysis:-

a. Hotels with calculated AVG Rating > say x(4.5)

b. Hotels with Food Rating > say x(4.5) (we can know how good is food in the hotel)

c. Locations of Hotels with AVG Rating > say x(4.5) (we can know which locations are on demand)

d. *Analyzing the number of People choosing that Hotel per month last few years*

3. *Modification :-*

Check this in all below: check whether the value of attributes belong to the corresponding domains of the attributes(CHAR,VARCHAR,INT etc.,) - domain Constraints.

a. *Insert :-*
   i. *Inserting a new Hotel's data(rooms,location,pricing etc.,).Check if there is a Hotel with same "Name and Location"*
   ii. *Inserting details of a new Customer.(Like if signed up for the first time).Check if there is anyone else with the same "emailID"*
   iii. *Insert the details of Reviews/Rating given by the customer on the Hotel.Check if the Hotel that it is referring to exists.*

b. *Update :-*
   i. *Update the "is_Booked(Booking Status)" of a room.*
   ii. *Update the "Price Per day" of a room(if the prices change with inflation)*
   iii. *Update the "contact Number of an Employee" (if he takes up a new active number).Check if it matches with anyone else.(like with any other person in the database)*

c. *Delete :-*

i. Delete a Hotel’s Entry (if they leave the tie-up).Check if it’s there in Room Type,Booking Status etc.,(which have its name as Foreign key)

ii. Delete a Customer’s Entry (if they have deleted their account)

iii. Delete a Employee’s details (if they leave the Hotel)

Summary :-

We are able to manage a “Database of Hotel’s” that makes Customer’s life easy by giving them the chance to explore all hotels without physically visiting them and select the desired hotel.

This also allows the Hotel’s to improve themselves based on the rating given by the Customer’s.