**Names:** Omar Kammouni, Ahmad Wehbe, Mahmoud Saleh, Kassem Al-Ahmad.

**Project Description:**

This project aims at building a database system to manage information about hotels.

**Entities:**

We will maintain information about the following set of entities:

1. Hotel: Name, location, capacity.
2. Customers: name, ID (key), gender, address, email, phone number.
3. Staff: ID (key), name, phone number, shifts.
4. Room: room number, double, triple, floor, availability.
5. Invoice: description, total amount, invoice ID.
6. Guests: ID, name, phone number. (weak entity).
7. Services: Type, Price.
8. Facilities: Type, Price.
9. Owners: ID, Name, address, Phone number, email.
10. Restaurants: Name, #tables and chairs, prices, menu.

**Relationships:**

The following relationships will hold between our entities:

1. Customers will reserve rooms (1-to-many).
2. Customers will pay invoices (1-to-1).
3. Hotel has rooms (1-to-many).
4. Hotel has restaurants (1-to-many).
5. Staff works in hotel (1 hotel-to-many staff).
6. Staff has a manager (1-to-many).
7. Customer uses facilities (1-to-many).
8. Guests visit customers (1 customer-to-2 guests).
9. Rooms requires services (1-to-many).
10. Hotels can be owned by owners (1-to-1).

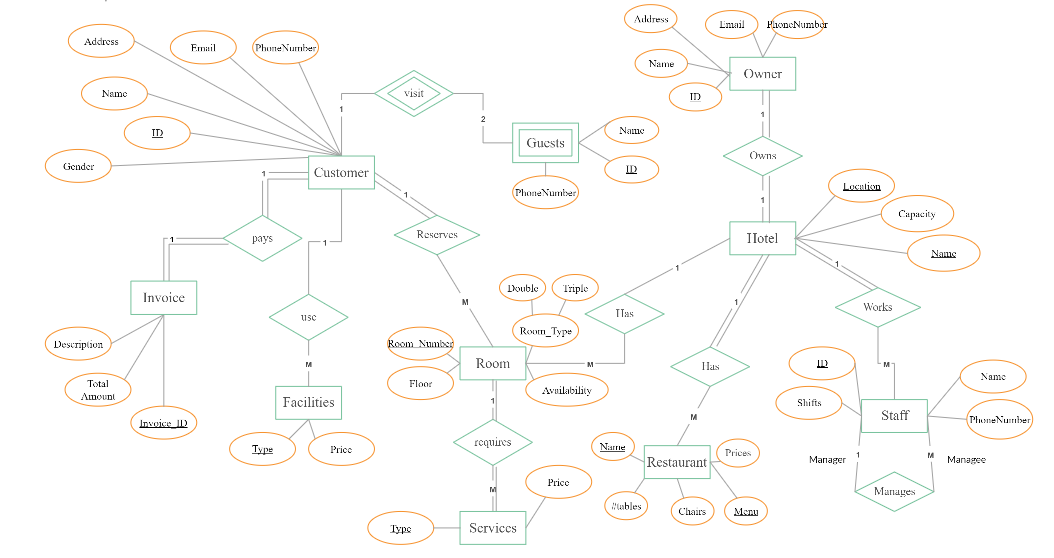
**Constraints:**

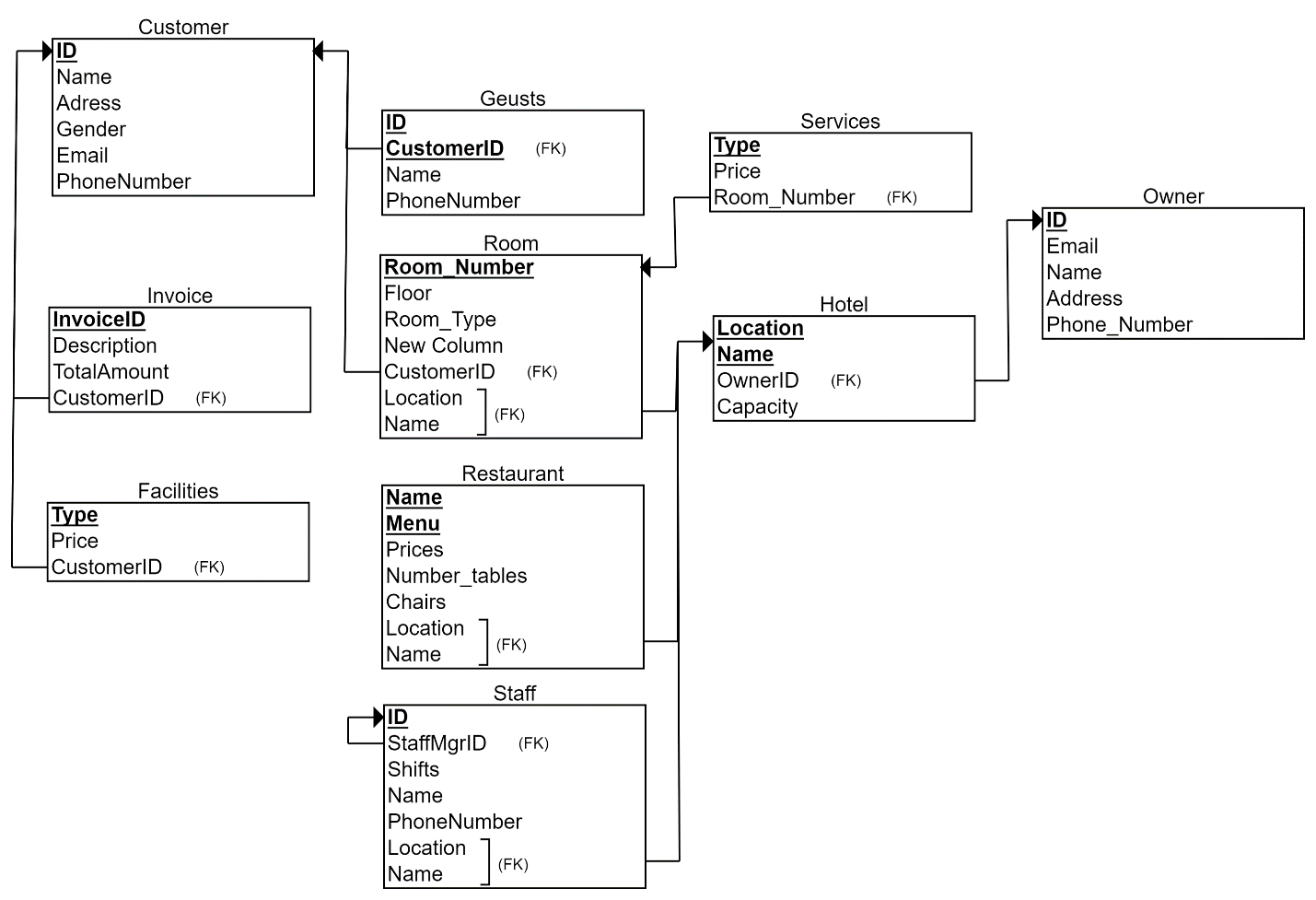
1. No two customers can reserve the same room.
2. A room cannot be in two hotels.
3. An invoice cannot be paid by more than one customer.
4. A staff cannot be managed by more than one manager.
5. A staff cannot work at more than one hotel.
6. Guests cannot visit or stay after 9pm.
7. No more than 2 guests can visit per room.
8. Facilities can only be used by customers.

**Sample Queries:**

1. **List all customers alphabetically.**
2. What is the remaining number of invoices unpaid each day?
3. **List all the branches of the hotel in different countries.**
4. List all male and female customers separately.
5. List all staff by shifts (day or night).
6. List the owners of the hotels.
7. **What are the available services for each room?**
8. **How many tables do we have in each restaurant?**
9. **What is the remaini cng number of rooms available?**
10. **List the customers who used the facilities.**

**EER Diagram:**



**Relational Diagram:**

**Views & Reasons:**

1. List all male and female customers separately:

CREATE VIEW MaleCustomers AS

SELECT COUNT(Name) AS Males

FROM Customer

WHERE Gender = "Male";

CREATE VIEW FemaleCustomers AS

SELECT COUNT(Name) AS Females

FROM Customer

WHERE Gender = "Female";

SELECT \* FROM MaleCustomers;

SELECT \* FROM FemaleCustomers;

The reason behind this view is to do statistics and compare the number of male customers to that of female customers. If there are more male customers, we might upgrade our gym. If there are more female customers, we will open a spa.

1. List the owners of the hotels:

CREATE VIEW HotelOwner AS

SELECT ID, Name, PhoneNumber, OwnerID, HotelName, HotelLocation

FROM Owner INNER JOIN Hotel

Where ID = OwnerID;

SELECT \* FROM HotelOwner;

The reason behind this view is to list all the owners of the hotels with their phone numbers in case there are complaints from customers or Urgent/Emergency matters in the hotel.

1. List all staff by shifts (day or night).

CREATE VIEW StaffDayShift AS

SELECT NAME, ID, PhoneNumber, Hotel\_Location, Hotel\_Name

FROM STAFF

WHERE Shifts = "Day";

CREATE VIEW StaffNightShift AS

SELECT NAME, ID, PhoneNumber, Hotel\_Location, Hotel\_Name

FROM STAFF

WHERE Shifts = "Night";

SELECT \* FROM StaffDayShift;

SELECT \* FROM StaffNightShift;

The reason behind this view is to return the Names and Phone Numbers of the Staff who works at Day and those who works at night.

1. List the remaining number of invoices unpaid each day

CREATE VIEW UnpaidInvoices AS

SELECT Name, CustomerID, InvoiceID, Description, TotalAmount, PhoneNumber

FROM INVOICE INNER JOIN CUSTOMER

WHERE CustomerID = ID AND Description = "not paid";

SELECT COUNT(\*) AS Unpaid FROM UnpaidInvoices;

The reason behind this view is to list all the customers who has not paid their invoice yet.

1. List all customers alphabetically

CREATE VIEW list\_alphabatically AS

SELECT Name, Room\_Number FROM cutomer

AS C INNER JOIN room AS R ON C.ID = R.CustomerID

ORDER BY Name;

SELECT \* AS current\_users FROM list\_alphabatically;

SELECT COUNT(\*) AS number\_user FROM list\_alphabatically;

The reason for this query is to get clean view of all customers in my database and if the employee wants to get the room of a customer using his name only, he can get it easier without doing a query just for that.

1. List all the branches of the hotel in different countries

CREATE VIEW branches AS

SELECT HotelLocation, HotelName FROM hotel;

SELECT \* AS branches\_list FROM branches;

This view is important to know what the names other branches of the hotel are and where the locations of them are. For instance, if we want to redirect a customer to other branch of the hotel, we can tell him the name and the location.

1. How many tables and chairs we have in each restaurant ?

CREATE VIEW RestaurantTablesNumber AS

SELECT Name, Hotel\_Name, Hotel\_Location, Number\_tables, Number\_Chairs

FROM RESTAURANT;

SELECT \* FROM RestaurantTablesNumber;

The reason behind this view is to give access for the administration to know the number of tables and chairs in a restaurant and knowing the hotel that this restaurant is in.

1. What is the remaining number of available rooms ?

CREATE VIEW Available\_Rooms AS

SELECT COUNT(Room\_Number)

FROM ROOM

WHERE Availability = "Available"

UNION

SELECT Room\_Number

FROM ROOM

WHERE Availability = "Available";

SELECT \* FROM Available\_Rooms;

The reason behind this view is to give access for the hotel's management to know how many rooms are still available as well as their numbers.

1. List the customers who used the facilities

CREATE VIEW Facilities\_Customers AS

SELECT Name, Type

FROM FACILITIES INNER JOIN CUSTOMER

WHERE CustomerID = ID;

SELECT \* FROM Facilities\_Customers;

The reason behind this view is to give access for the facilities management to know the names of the customers that used which facilities

**Indexes:**

1. CREATE INDEX idx\_gender ON CUSTOMER(Gender);

Because Gender is most probably going to be used frequently in where statements.

1. CREATE INDEX idx\_ownerid ON HOTEL(OwnerID);

Because OwnerID is most probably going to be used frequently in where statements.

1. CREATE INDEX idx\_shifts ON STAFF(Shifts);

Because shifts is most probably going to be used frequently in where statements.

1. CREATE INDEX idx\_description ON INVOICE(Description);

Because Description is most probably going to be used frequently in where statements.

1. CREATE INDEX idx\_availability ON ROOM(Availability);

Because availability is most probably going to be used frequently in where statements**.**

1. CREATE INDEX idx\_name ON CUSTOMER(Name);

Because name is used in the query order by in list\_alphabatically view.