★★★★★

Hotel
Management System

Group 2
Akshit Saxena – 002673782
Neville Roy – 002726594
Urmi Patel – 002772667
Zhiliang Liu - 002650051

Contents

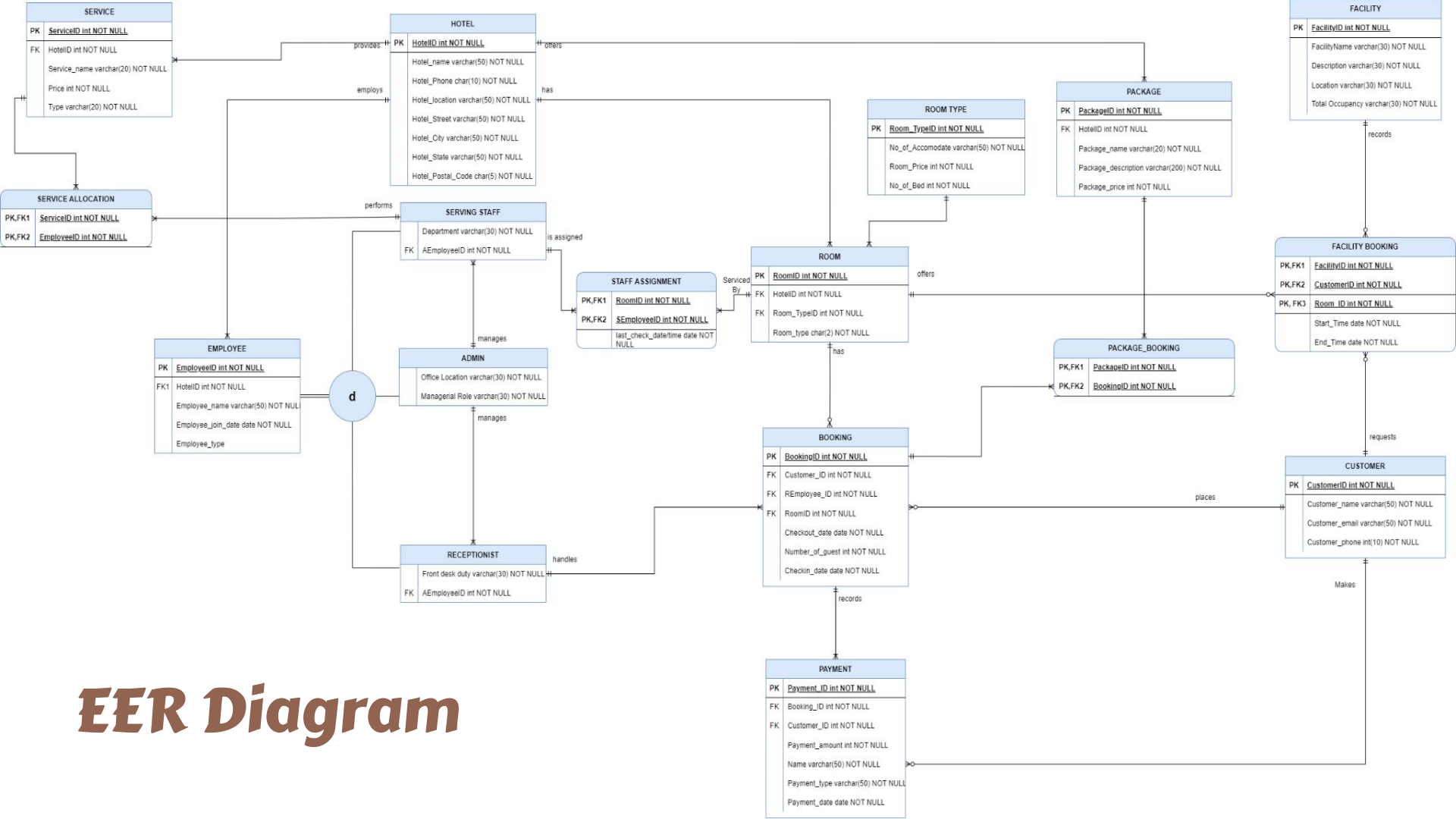
Project Overview

- In this project we aimed to design a effective database model for the purpose of hotel management.
- Hotel management operations generate large quantities of data that should be readily accessible to hotel staff to ensure smooth operations.
- The proposed database model enables storage and retrieval of data in an organised manner.
- Ensuring the security and privacy of guest operations is also an essential feature

High Level Design

- **Entities**
  - ADMIN
  - BOOKING
  - CUSTOMER
  - EMPLOYEE
  - FACILITY
  - HOTEL
  - PACKAGE
  - PAYMENT
  - ROOM
  - SERVICE

- **Relationships Used**
  - One-to-one
  - One-to-many
  - Many-to-many
  - Associate entities


SERVICE
PK ServiceID int NOT NULL
FK HotelID int NOT NULL
Service_name varchar(20) NOT NULL
Price int NOT NULL
Type varchar(20) NOT NULL
HOTEL
PK HotelID int NOT NULL
Hotel_name varchar(50) NOT NULL
Hotel_Phone char(10) NOT NULL
Hotel_location varchar(50) NOT NULL
Hotel_Street varchar(50) NOT NULL
Hotel_City varchar(50) NOT NULL
Hotel_State varchar(50) NOT NULL
Hotel_Postal_Code char(5) NOT NULL
FACILITY
PK FacilityID int NOT NULL
FacilityName varchar(30) NOT NULL
Description varchar(30) NOT NULL
Location varchar(30) NOT NULL
Total Occupancy varchar(30) NOT NULL
ROOM TYPE
PK Room_TypeID int NOT NULL
No_of_Accomodate varchar(50) NOT NULL
Room_Price int NOT NULL
No_of_Bed int NOT NULL
PACKAGE
PK PackageID int NOT NULL
FK HotelID int NOT NULL
Package_name varchar(20) NOT NULL
Package_description varchar(200) NOT NULL
Package_price int NOT NULL
SERVICE ALLOCATION
PK,FK1 ServiceID int NOT NULL
PK,FK2 EmployeeID int NOT NULL
SERVING STAFF
Department varchar(30) NOT NULL
FK AEmployeeID int NOT NULL
STAFF ASSIGNMENT
PK,FK1 RoomID int NOT NULL
PK,FK2 SEmployeeID int NOT NULL
last_check_date/time date NOT NULL
ROOM
PK RoomID int NOT NULL
FK HotelID int NOT NULL
FK Room_TypeID int NOT NULL
Room_type char(2) NOT NULL
FACILITY BOOKING
PK,FK1 FacilityID int NOT NULL
PK,FK2 CustomerID int NOT NULL
PK, FK3 Room_ID int NOT NULL
Start_Time date NOT NULL
End_Time date NOT NULL
EMPLOYEE
PK EmployeeID int NOT NULL
FK1 HotelID int NOT NULL
Employee_name varchar(50) NOT NULL
Employee_join_date date NOT NULL
Employee_type
d
ADMIN
Office Location varchar(30) NOT NULL
Managerial Role varchar(30) NOT NULL
PACKAGE_BOOKING
PK,FK1 PackageID int NOT NULL
PK,FK2 BookingID int NOT NULL
BOOKING
PK BookingID int NOT NULL
FK Customer_ID int NOT NULL
FK REmployee_ID int NOT NULL
FK RoomID int NOT NULL
Checkout_date date NOT NULL
Number_of_guest int NOT NULL
Checkin_date date NOT NULL
CUSTOMER
PK CustomerID int NOT NULL
Customer_name varchar(50) NOT NULL
Customer_email varchar(50) NOT NULL
Customer_phone int(10) NOT NULL
RECEPTIONIST
Front desk duty varchar(30) NOT NULL
FK AEmployeeID int NOT NULL
PAYMENT
PK Payment_ID int NOT NULL
FK Booking_ID int NOT NULL
FK Customer_ID int NOT NULL
Payment_amount int NOT NULL
Name varchar(50) NOT NULL
Payment_type varchar(50) NOT NULL
Payment_date date NOT NULL
provides
offers
employs
has
performs
is assigned
manages
Serviced By
offers
has
records
manages
handles
places
requests
Makes
records

EER Diagram

Database Objects - Stored Procedure

```
CREATE PROCEDURE GetRoomDetails
    @InputRoomID INT,
    @HotelID INT OUTPUT,
    @RoomTypeID INT OUTPUT,
    @RoomType VARCHAR(2) OUTPUT
AS
BEGIN
    BEGIN TRY
        SELECT
            @HotelID = HotelID,
            @RoomTypeID = RoomTypeID,
            @RoomType = RoomType
        FROM ROOM
        WHERE RoomID = @InputRoomID;
    END TRY
    BEGIN CATCH
        PRINT ERROR_MESSAGE();
    END CATCH;
END;
```

Messages

23:10:49 Started executing query at Line 21
Hotel ID: 2
Room Type ID: 2
Room Type: B
Total execution time: 00:00:00.008

Database Object - DML Trigger

```
CREATE TRIGGER tr_booking_insert
ON BOOKING
AFTER INSERT
AS
BEGIN
    INSERT INTO PACKAGE_BOOKING (PackageID, BookingID)
    SELECT p.PackageID, i.BookingID
    FROM inserted i
    INNER JOIN ROOM r ON i.RoomID = r.RoomID
    INNER JOIN PACKAGE p ON r.HotelID = p.HotelID;
END;
```

- This trigger is designed to automatically create corresponding records in the PACKAGE_BOOKING table when a new booking is inserted into the BOOKING table.
- It does so by associating the booking with a package based on the hotel and room associated with the booking.

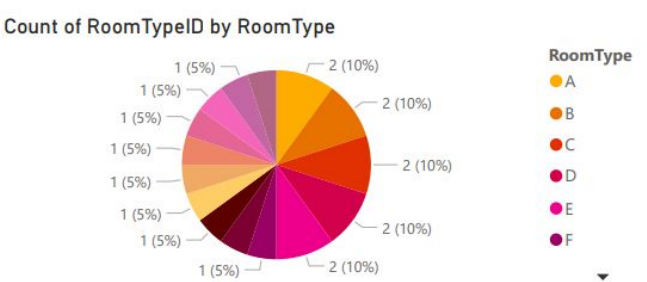
Database Objects - View

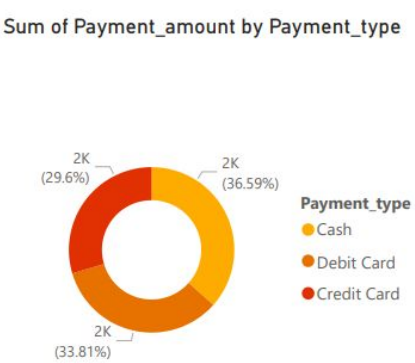
```
CREATE VIEW CustomerView AS
SELECT CustomerID, Customer_name, Customer_email, Customer_phone
FROM CUSTOMER;
```

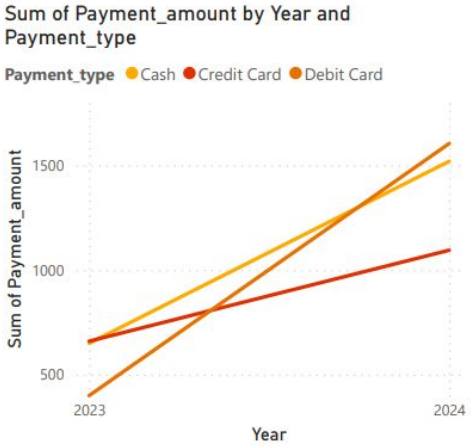
	CustomerID	Customer_name	Customer_email	Customer_phone
1	1	John Doe	john.doe@example.com	1234567890
2	2	Jane Smith	jane.smith@example.com	9876543210
3	3	Mike Johnson	mike.johnson@example.com	5551234567
4	4	Emily Brown	emily.brown@example.com	4567890123
5	5	Chris Wilson	chris.wilson@example.com	7890123456
6	6	Anna Taylor	anna.taylor@example.com	3216549870
7	7	David Miller	david.miller@example.com	6669998888
8	8	Sophie Davis	sophie.davis@example.com	1112223333
9	9	Ryan Turner	ryan.turner@example.com	4445556666
10	10	Olivia White	olivia.white@example.com	2223334444
11	11	Ethan Carter	ethan.carter@example.com	7778889999
12	12	Emma Harris	emma.harris@example.com	3334445555
13	13	Jack Turner	jack.turner@example.com	9990001111
14	14	Ava Robinson	ava.robinson@example.com	1110009999
15	15	Lucas Parker	lucas.parker@example.com	4445556666
16	16	Lily Hughes	lily.hughes@example.com	8887776666

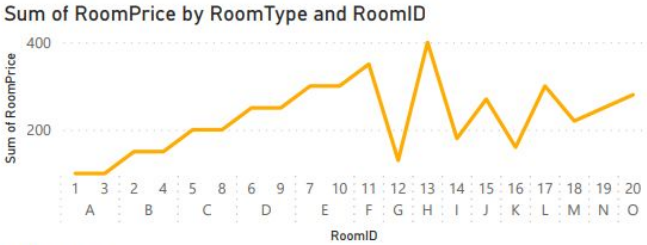
- CustomerView is used to display essential customer details.
- It displays all contact details of the customer for the various hotels.
- This simplifies the access to customer information.
- Enhances the data security by utilizing a view for controlled data retrieval.

Dashboard

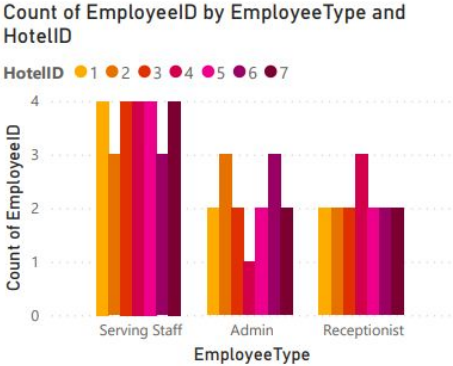

Count of RoomTypeID by RoomType
1 (5%)
2 (10%)
1 (5%)
2 (10%)
1 (5%)
2 (10%)
1 (5%)
2 (10%)
1 (5%)
2 (10%)
1 (5%)
2 (10%)
1 (5%)
2 (10%)
RoomType
A
B
C
D
E
F

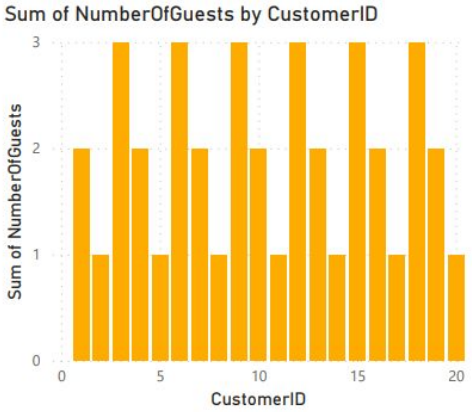

Sum of Payment_amount by Payment_type
2K
(29.6%)
2K
(36.59%)
2K
(33.81%)
Payment_type
Cash
Debit Card
Credit Card


Sum of Payment_amount by Year and Payment_type
Payment_type
Cash
Credit Card
Debit Card
Sum of Payment_amount
1500
1000
500
2023
2024
Year


Sum of RoomPrice by RoomType and RoomID
Sum of RoomPrice
400
200
1 3 2 4 5 8 6 9 7 10 11 12 13 14 15 16 17 18 19 20
A B C D E F G H I J K L M N O
RoomID


FullAddress
UNITED STATES
Gulf of Mexico
MEXICO
CUBA
HAITI
GUATEMALA
Caribbean
Microsoft Bing
© 2023 TomTom, © 2023 Microsoft Corporation


Count of EmployeeID by EmployeeType and HotelID
HotelID
1
2
3
4
5
6
7
Count of EmployeeID
4
3
2
1
0
Serving Staff
Admin
Receptionist
EmployeeType


Sum of NumberOfGuests by CustomerID
Sum of NumberOfGuests
3
2
1
0
0
5
10
15
20
CustomerID

Thank You