

Cơ Sở Dữ Liệu

Báo Cáo Bài Tập Lớn

Tên Nhóm : 0205

Thành Viên : Ngô Đình Linh (23020394)

Tên Dự Án : Cơ Sở Dữ Liệu Quản Lý Khách Sạn

Chi Tiết Nội Dung Các Bước Thực Hiện :

1.Lựa Chọn Hệ Thống

- Dự án được thực hiện nhằm xây dựng một hệ thống cơ sở dữ liệu quản lý đặt phòng khách sạn. Hệ thống cung cấp các chức năng như quản lý khách hàng, khách sạn, phòng, dịch vụ bổ sung, đặt phòng, và hóa đơn.

Hệ thống bao gồm các thành phần chính sau:

- **Thực thể (Entities):**
 - **Customer (Khách hàng):** Quản lý thông tin khách hàng.
 - **Hotel (Khách sạn):** Quản lý thông tin khách sạn.
 - **Room (Phòng):** Quản lý thông tin các phòng trong khách sạn.
 - **Booking (Đặt phòng):** Quản lý thông tin đặt phòng.
 - **Service (Dịch vụ):** Quản lý thông tin các dịch vụ bổ sung.
 - **ServiceBooking (Đặt dịch vụ):** Quản lý thông tin đặt dịch vụ bổ sung.
 - **Staff (Nhân viên):** Quản lý thông tin nhân viên khách sạn.
 - **Invoice (Hóa đơn):** Quản lý thông tin hóa đơn khách hàng.
 - **Feedback (Đánh giá) :** Quản lý các lượt đánh giá dịch vụ của khách hàng
 - **Promotion (Khuyến mãi) :** Quản lý các thông tin khuyến mãi của các dịch vụ

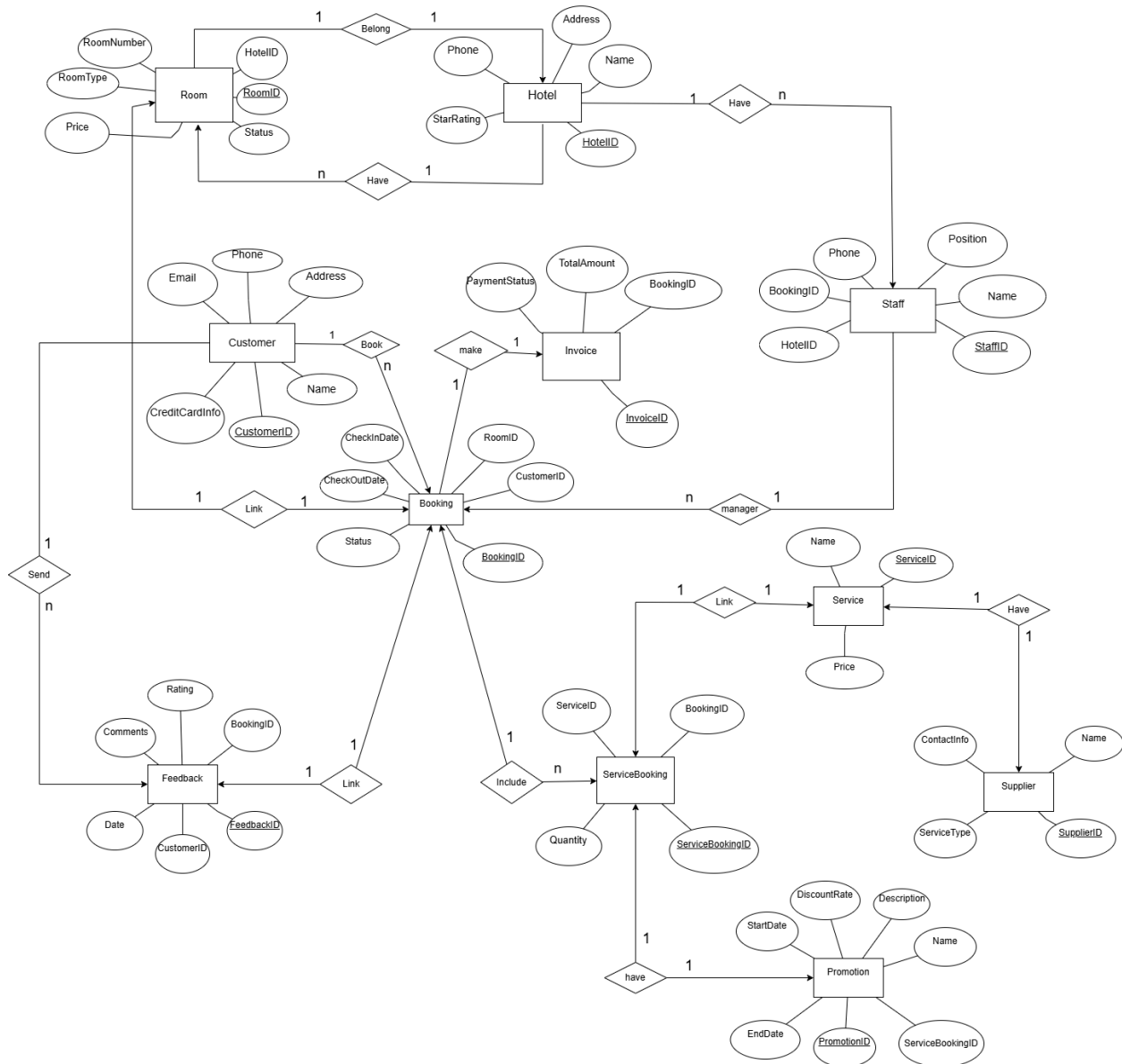
- Supplier (Nhà cung cấp): Quản lý thông các nhà cung cấp dịch vụ
- **Luồng công việc chính:**
 - **Đặt phòng:** Khách hàng chọn phòng từ khách sạn, tạo đơn đặt phòng và nhận hóa đơn.
 - **Sử dụng dịch vụ:** Khách hàng có thể thêm dịch vụ vào đơn đặt phòng.
 - **Phản hồi:** Sau khi hoàn thành, khách hàng có thể gửi phản hồi về trải nghiệm.
 - **Khuyến mãi:** Các dịch vụ có thể có chương trình khuyến mãi để thu hút khách hàng.
 - **Quản lý:** Nhân viên khách sạn quản lý các đơn đặt phòng và dịch vụ cho khách hàng.

2.Thiết Kế Mô Hình ER

Mô hình ER được xây dựng để thể hiện các thực thể, thuộc tính và mối quan hệ giữa chúng. Các mối quan hệ chính bao gồm:

- Một Khách Hàng có thể thực hiện nhiều lần Đặt Phòng.
- Một Khách Sạn có nhiều nhân viên
- Mỗi Đặt Phòng chỉ liên kết với một Phòng cụ thể.
- Một Khách Sạn có thể có nhiều Phòng.
- Mỗi Phòng chỉ thuộc về một Khách Sạn duy nhất.
- Một Đặt Phòng có thể bao gồm nhiều Đặt Dịch Vụ.
- Mỗi Đặt Dịch Vụ chỉ liên kết với một Dịch Vụ cụ thể.
- Một Nhân Viên có thể quản lý nhiều Đặt Phòng.
- Mỗi Nhân Viên chỉ thuộc một khách sạn duy nhất
- Mỗi Hóa Đơn chỉ liên kết với một Đặt Phòng duy nhất.
- Khách Hàng có thể gửi nhiều phản hồi.
- Một Đặt Phòng chỉ có một phản hồi duy nhất
- Một Dịch Vụ có một nhà cung cấp duy nhất

- Một Đặt Dịch Vụ có một khuyến mãi duy nhất

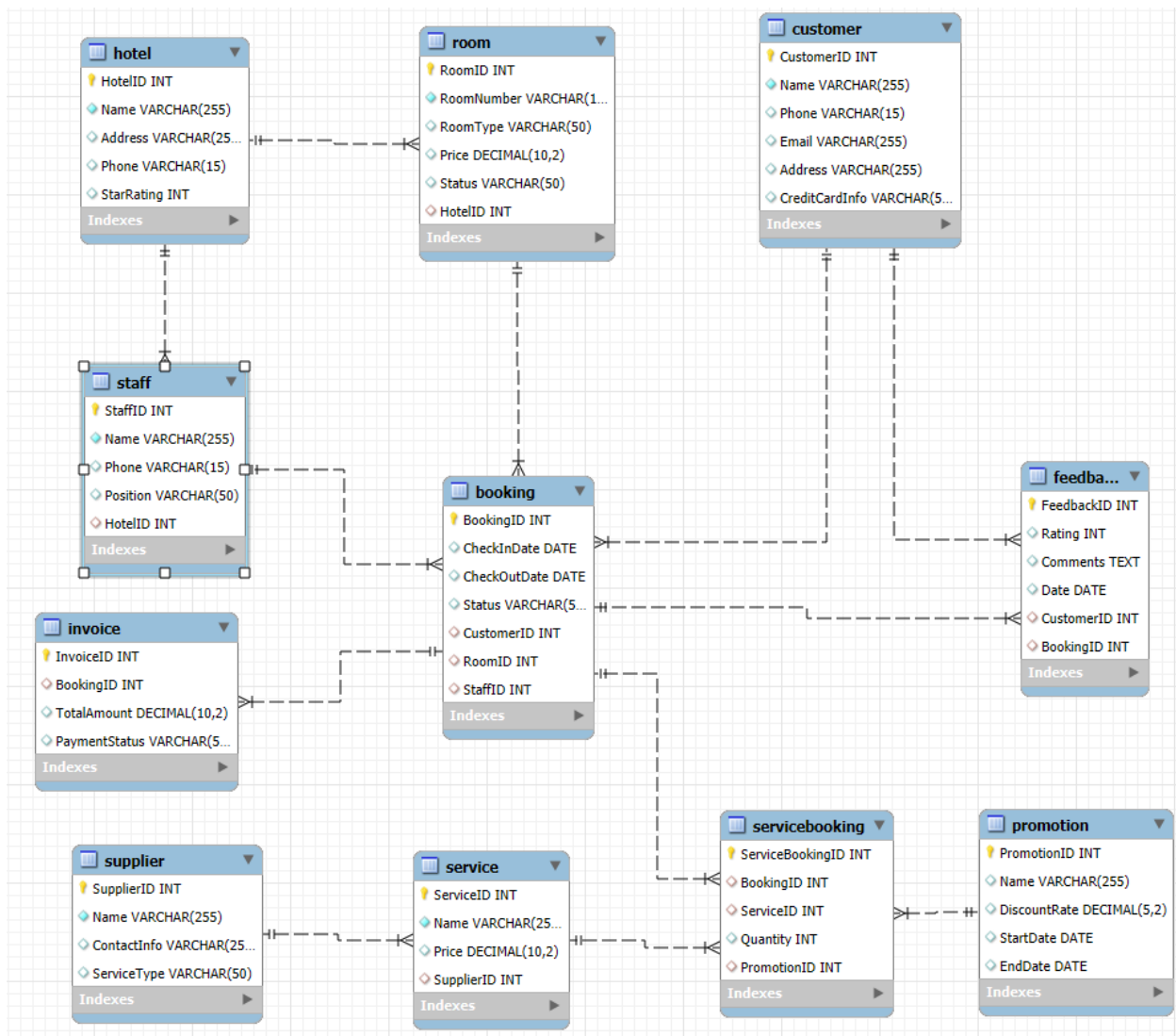


3.Thiết Kế Mô Hình Quan Hệ

Dựa trên mô hình ER, các bảng được thiết kế và chuẩn hóa theo dạng chuẩn thứ ba (3NF). Mỗi bảng đại diện cho một thực thể hoặc mối quan hệ và được thiết kế như sau:

- **Customer:** Chứa thông tin khách hàng.
- **Hotel:** Chứa thông tin về các khách sạn.

- **Room:** Liên kết với Hotel và chứa thông tin về các phòng.
- **Booking:** Liên kết với Customer và Room, chứa thông tin về các đặt phòng.
- **Service:** Chứa thông tin về các dịch vụ bổ sung.
- **ServiceBooking:** Liên kết giữa Booking và Service.
- **Staff:** Chứa thông tin về nhân viên khách sạn.
- **Invoice:** Liên kết với Booking, chứa thông tin hóa đơn.
- **Feedback :** Liên kết với Customer và Booking, chứa thông tin phản hồi của khách hàng
- **Promotion:** Liên kết với ServiceBooking , chứa thông tin khuyến mãi của các dịch vụ
- **Supplier :** Liên kết với Service , chứa thông tin của các nhà cung cấp dịch vụ



4. Tạo Cơ Sở Dữ Liệu

Tạo tệp `createDB.sql`, các bảng được tạo trong cơ sở dữ liệu `Hotel_Management`. Dưới đây là các bảng đã được tạo:

1. **Customer** (Khách hàng)
2. **Hotel** (Khách sạn)
3. **Room** (Phòng)
4. **Booking** (Đặt phòng)
5. **Service** (Dịch vụ)
6. **ServiceBooking** (Đặt dịch vụ)

7. **Staff (Nhân viên)**
8. **Invoice (Hóa đơn)**
9. Feedback (Đánh giá)
10. Promotion (Khuyến mãi)
11. Supplier (Nhà cung cấp)

5. Thiết Lập Ràng Buộc

Trong tệp `constraints.sql`, ba ràng buộc chính đã được thêm vào:

1. **Số điện thoại khách hàng phải duy nhất:** Ràng buộc `Unique_Phone` đảm bảo không có trùng lặp số điện thoại trong bảng `Customer`.
2. **Số phòng trong mỗi khách sạn không được trùng:** Ràng buộc `Unique_Room` đảm bảo không trùng số phòng trong cùng một khách sạn.
3. **Trạng thái thanh toán chỉ nhận các giá trị hợp lệ:** Chỉ cho phép các giá trị `Paid`, `Unpaid`, và `Refunded` trong bảng `Invoice`.

```
-- Đảm bảo số điện thoại khách hàng là duy nhất
ALTER TABLE Customer ADD CONSTRAINT Unique_Phone UNIQUE (Phone);

-- Đảm bảo số phòng trong mỗi khách sạn không trùng lặp
ALTER TABLE Room ADD CONSTRAINT Unique_Room UNIQUE (HotelID, RoomNumber);

-- Đảm bảo trạng thái thanh toán chỉ nhận các giá trị hợp lệ
ALTER TABLE Invoice MODIFY PaymentStatus ENUM('Paid', 'Unpaid', 'Refunded');
```

6. Chèn Dữ Liệu

Dữ liệu mẫu đã được chèn vào cơ sở dữ liệu trong tệp `insert.sql`, với ít nhất 5 bản ghi cho mỗi bảng:

- **Customer:** 7 khách hàng.
- **Hotel:** 7 khách sạn.
- **Room:** 7 phòng thuộc các khách sạn khác nhau.

- **Service:** 7 dịch vụ bổ sung.
- **Staff:** 7 nhân viên.
- **Booking:** 7 đặt phòng.
- **ServiceBooking:** 7 bản ghi đặt dịch vụ.
- **Invoice:** 7 hóa đơn.
- Supplier : 7 nhà cung cấp dịch vụ
- Promotion : 7 thông tin khuyến mãi
- Feedback : 5 lượt phản hồi

7.Tạo Các Câu Lệnh Truy Vấn

Trong tệp `queries.sql`, các truy vấn SQL được viết để giải quyết các yêu cầu khác nhau:

1. Danh sách tất cả các khách sạn với số lượng phòng.
2. Tìm tất cả các phòng đôi .
3. Tổng doanh thu từ tất cả các hóa đơn.
4. Danh sách các khách hàng đã đặt phòng.
5. Dịch vụ nào được sử dụng nhiều nhất.

```

-- 1. Danh sách tất cả các khách sạn với số lượng phòng
▸ SELECT H.Name AS HotelName, COUNT(R.RoomID) AS RoomCount
   FROM Hotel H
  LEFT JOIN Room R ON H.HotelID = R.HotelID
 GROUP BY H.HotelID;

-- 2. Tìm tất cả các phòng đôi
▸ SELECT RoomNumber, RoomType, Price
   FROM Room
  WHERE RoomType = 'Single' ;

-- 3. Tổng doanh thu từ tất cả các hóa đơn
▸ SELECT SUM(TotalAmount) AS TotalRevenue
   FROM Invoice;

-- 4. Danh sách các khách hàng đã đặt phòng
▸ SELECT DISTINCT C.Name AS CustomerName, C.Phone
   FROM Customer C
  JOIN Booking B ON C.CustomerID = B.CustomerID;

-- 5. Dịch vụ nào được sử dụng nhiều nhất
▸ SELECT S.Name AS ServiceName, SUM(SB.Quantity) AS TotalUsage
   FROM Service S
  JOIN ServiceBooking SB ON S.ServiceID = SB.ServiceID
 GROUP BY S.ServiceID
 ORDER BY TotalUsage DESC
 LIMIT 1;

```

8. Tạo Giao Dịch

Ba giao dịch (transaction) được viết trong tệp `transaction.sql` để đảm bảo tính toàn vẹn dữ liệu khi đặt phòng, bao gồm:

1. Thêm đơn đặt phòng và hóa đơn, rollback nếu có lỗi.
2. **Thêm phản hồi từ khách hàng.**
3. Thêm dịch vụ và khuyến mãi.


```

-- 1. Thêm đơn đặt phòng và hóa đơn, rollback nếu có lỗi
START TRANSACTION;
INSERT INTO Booking (CheckInDate, CheckOutDate, Status, CustomerID, RoomID)
VALUES ('2024-12-15', '2024-12-20', 'Confirmed', 1, 2);

INSERT INTO Invoice (BookingID, TotalAmount, PaymentStatus)
VALUES (LAST_INSERT_ID(), 500.00, 'Unpaid');

ROLLBACK; -- Hoàn tác nếu có lỗi

-- 2. Thêm phản hồi từ khách hàng
START TRANSACTION;
INSERT INTO Feedback (Rating, Comments, Date, CustomerID, BookingID)
VALUES (5, 'Excellent service!', '2024-12-12', 3, 2);

COMMIT; -- Lưu thay đổi

-- 3. Thêm dịch vụ và khuyến mãi
START TRANSACTION;
INSERT INTO Service (Name, Price) VALUES ('Spa', 50.00);
INSERT INTO Promotion (Name, DiscountRate, StartDate, EndDate)
VALUES ('Holiday Sale', 10, '2024-12-01', '2024-12-31');

COMMIT;

```

9. Tạo Trigger Và Procedure

Hai trigger được tạo trong tệp `trigger.sql` :

- **Trigger `UpdateRoomStatusAfterBooking`** : Sau khi thêm một bản ghi vào bảng `Booking` , trạng thái của phòng trong bảng `Room` sẽ được cập nhật thành "Booked".
- **Trigger `NotifyFeedback`** : Gửi thông báo khi có phản hồi mới

```

-- Trigger: Tự động cập nhật trạng thái phòng khi có đơn đặt phòng
CREATE TRIGGER UpdateRoomStatusAfterBooking
AFTER INSERT ON Booking
FOR EACH ROW
BEGIN
    UPDATE Room
    SET Status = 'Occupied'
    WHERE RoomID = NEW.RoomID;
END;

-- Trigger: Gửi thông báo khi có phản hồi mới
CREATE TRIGGER NotifyFeedback
AFTER INSERT ON Feedback
FOR EACH ROW
BEGIN
    INSERT INTO Notifications (Message, Date)
    VALUES (CONCAT('New feedback received from CustomerID: ', NEW.CustomerID), NOW());
END;

```

Một procedure được tạo trong tệp `procedure.sql` để tính tổng doanh thu theo khách sạn:

- **Procedure GetHotelRevenue:** tính tổng doanh thu theo id của khách sạn.

```

-- Procedure: Tính tổng doanh thu theo khách sạn
CREATE PROCEDURE GetHotelRevenue(IN hotelID INT)
BEGIN
    SELECT SUM(I.TotalAmount) AS TotalRevenue
    FROM Invoice I
    JOIN Booking B ON I.BookingID = B.BookingID
    JOIN Room R ON B.RoomID = R.RoomID
    WHERE R.HotelID = hotelID;
END;

```

Kết quả đạt được

Hệ thống cơ sở dữ liệu đã được thiết kế hoàn chỉnh và đầy đủ chức năng, bao gồm:

1. Cơ sở dữ liệu với 11 bảng chuẩn hóa.
2. Dữ liệu mẫu với ít nhất 7 bản ghi trên mỗi bảng.

3. **Các ràng buộc đảm bảo tính toàn vẹn dữ liệu.**
4. **Truy vấn SQL để lấy dữ liệu từ hệ thống.**
5. **Tăng cường tính năng với trigger và procedure.**
6. **Giao dịch đảm bảo tính toàn vẹn dữ liệu khi thao tác phức tạp.**

Kết Luận

Dự án đã hoàn thành đầy đủ các yêu cầu, từ phân tích, thiết kế, triển khai đến kiểm tra tính năng. Hệ thống có thể được sử dụng để quản lý đặt phòng khách sạn một cách hiệu quả và có tiềm năng mở rộng trong tương lai.