PROJECT

COFFEE SHOP MANAGEMENT

Student:

Nguyễn Hoàng Hùng – HE171511

Mục lục:

1. Phát biểu bài toán
2. Mô tả nghiệp vụ hệ thống
3. Mô hình ER của hệ thống
4. Mô hình quan hệ
5. Đặc tả yêu cầu dữ liệu
6. Danh sách các ràng buộc dữ liệu
7. Cài đặt vật lí
8. Phát biểu bài toán

Dự án quản lí quán Coffee tạo ra một cơ sở dữ liệu để quản lí và lưu trữ dữ liệu các nhân viên, khách hàng, sản phẩm, Order của khách hàng.

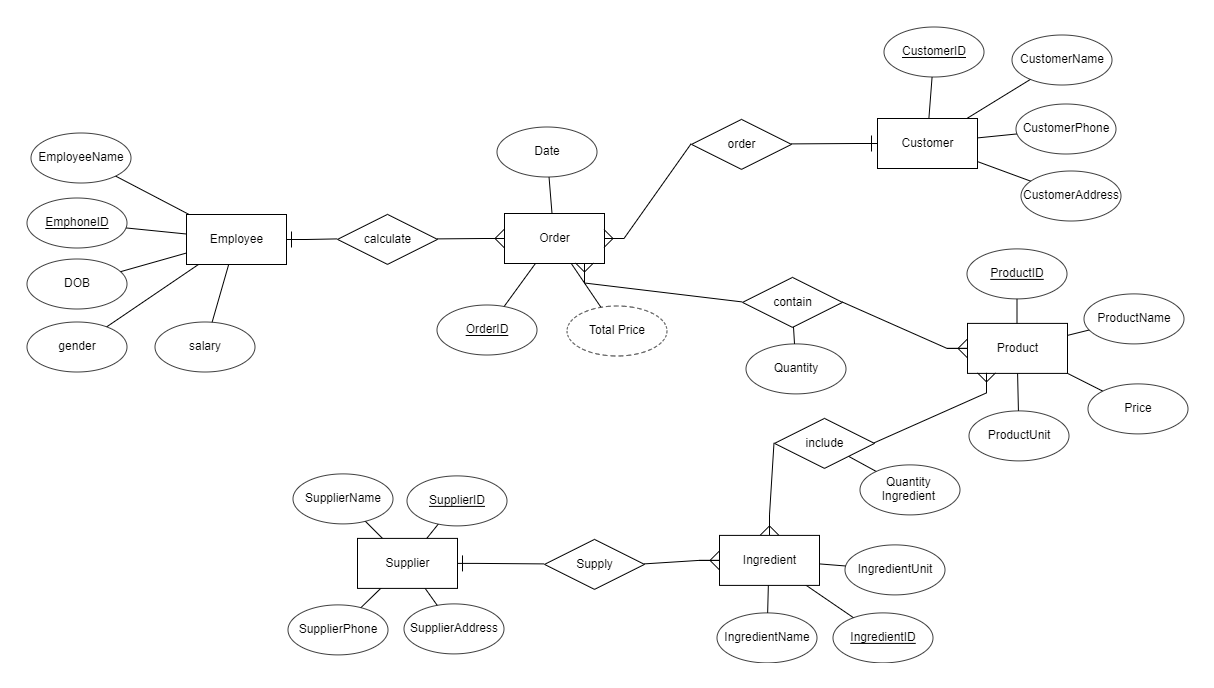
Cơ sở dữ liệu có thể từ các order của khách hàng để tính tổng số tiền mà khách hàng phải trả.

* Employee làm việc trong Store, trong Store có thể có nhiều Employee tuy nhiên một Employee chỉ làm việc cho một quán coffee.
* Các nhân viên trong quán sẽ có một người quản lí, và người quản lí đó sẽ quản lí tất cả các nhân viên còn lại trong store.
* Coffee Shop có lưu các thông tin của Employee gồm có: ID, Name, DOB, gender, phone, salary.
* Store gồm các thông tin: Store id, Store name, Email, phone, address.
* Mỗi Employee có thể phục vụ nhiều Customer, tuy nhiên mỗi Customer chỉ được phục vụ bởi một Employee.
* Các Customer sẽ order các Product ở trong Order. Nhiều Customer có thể order vào chung một Order.
* Customer gồm các thông tin: Customer ID, Customer Name, Customer Phone, Customer Address.
* Trong order sẽ chứa các Product do Customer đặt, Order sẽ gồm các thông tin:Order ID, date, Employee id, Product ID, Product quantity, Customer ID, Total Price.
* Một Product có thể ở trong nhiều Order khác nhau. Product chứa các thông tin sau: Product ID, Product Name, Price, Ingredient ID. Và trong Product có thể chứa nhiều Ingredient.
* Một Ingredient có thể ở trong nhiều Product khác nhau. Ingredient chứa các thông tin sau: Ingredient ID, Ingredient Name, Supplier ID.
* Supplier gồm các thông tin: Supplier ID, Supplier Name, Supplier Phone, Address.

1. Mô tả nghiệp vụ của hệ thống

* Hệ thống có thể tính được tổng tiền hóa đơn cho từng hóa đơn, tính tổng tiền toàn bộ hóa đơn cho từng khách hàng.
* Liệt kê danh sách sản phẩm, danh sách nhân viên, danh sách khách hàng, danh sách công thức pha chế

1. Mô hình ER của hệ thống



1. Mô hình quan hệ

Store (StoreID, StoreName, StoreEmail, StorePhone, StoreAddress)

Employee (EmployeeID, EmployeeName, DOB, gender, salary, StoreID)

Customer (CustomerID, CustomerName, CustomerPhone, CustomerAddress, EmployeeID)

Order (OrderID, OrderDate, CustomerID, EmployeeID)

OrderDetail (OrderID, ProductID, ProductQuantity)

Product (ProductID, ProductName, ProductUnit, Price)

ProductIngredient ( ProductID, IngredientID, IngredientQuantity)

Ingredient (IngredientID, IngredientName, IngredientUnit)

Supplier (SupplierID, SupplierName, SupplierPhone, SupplierAdress)

1. Đặc tả yêu cầu dữ liệu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data  Element | Description | Composition of data type | Length | Values |
| Store | Đây là Store, nơi để bán hàng và phục vụ Customer | StoreID,  StoreName, StoreEmail, StorePhone, StoreAddress |  |  |
| StoreID | có giá trị duy nhất, không trùng lặp với các Store khác | Int | 4 |  |
| StoreName | Tên của Store, sẽ không trùng lặp với các Store khác | Nvarchar | 50 |  |
| StoreEmail | Email của Store, các Khách hàng có feedback có thể sử dụng địa chỉ này. | Varchar | 50 | Email có dạng: |
| StorePhone | Sdt của Store | Char | 11 | Mỗi kí tự trong khoảng 0-9 |
| StoreAddress | Địa chỉ của cửa hàng | Nvarchar | 50 |  |
| Employee | Nhân viên sẽ phục vụ Khách hàng và order theo yêu cầu khách hàng | EmployeeID, EmployeeName, EmployeePhone, DOB, gender, salary, StoreID |  |  |
| Employee ID | có giá trị duy nhất, không trùng lặp với các Empoyee khác để phân biệt | Int | 4 |  |
| Employee Name | Tên của nhân viên | Nvarchar | 50 |  |
| DOB | Ngày sinh của nhân viên để xác định nhân viên đó từ 18 tuổi trở lên | Date | yyyy/MM/dd | dd=01-31,  MM = 1-12  yyyy>2005 |
| gender | Giới tính của nhân viên | varchar | 10 | Male hoặc female |
| salary | Lương của nhân viên | Double |  | Salary > 0 |
| Customer | Khách hàng | CustomerID  CustomerName CustomerPhone  CustomerAddress |  |  |
| CustomerID | Mỗi khách hành khi order sẽ có một ID riêng | int | 4 | cusID > 0 |
| CustomerName | Tên của khách hàng khi order | Nvarchar | 50 |  |
| CustomerAddress | Địa chỉ của khách hàng | Nvarchar | 50 |  |
| CustomerPhone | Số điện thoại của khách hàng khi họ order | Char | 11 | Mỗi kí tự trong khoảng 0-9 |
| Order | Đơn hàng của khách hàng | OrderID, date, CustomerID, EmployeeID |  |  |
| OrderID | Mã đơn hàng | Int | 9 |  |
| Date | Thời gian order món | Date | hh:mm  yyyy/MM/dd | hh = 6-23,  mm =00-59  dd = 1-31  MM = 1- 12 |
| Product |  | ProductID, ProductName, Price, SupplierID |  |  |
| ProductID | Mã của từng sản phẩm | Int | 4 |  |
| ProductName | Tên sản phẩm | Nvarchar | 50 |  |
| ProductUnit | Đơn vị tính với mỗi sản phẩm | Nvarchar | 50 |  |
| Price | Đơn giá của từng sản phẩm | Double |  | Price > 0 |
| Ingredient | Các nguyên liệu có trong từng sản phẩm | IngredientID,  IngredientName,  IngredientUnit |  |  |
| IngredientID |  | Int | 4 |  |
| IngredientName |  | Nvarchar | 50 |  |
| IngredientUnit |  | Nvarchar | 50 |  |
| Supplier | Nhà cung cấp các sản phẩm | SupplierID, SupplierName SupplierPhone SupplierAdress |  |  |
| SupplierID |  | Int | 4 |  |
| SupplierName |  | Nvarchar | 50 |  |
| SupplierPhone |  | Char | 11 | Mỗi kí tự trong khoảng 0-9 |
| SupplierAdress |  | Nvarchar | 50 |  |

1. Danh sách các ràng buộc dữ liệu

-Trong table Employee có thuộc tính StoreID là khóa ngoại từ bảng Store.

- Trong Customer có thuộc tính EmployeeID là khóa ngoại từ bảng Employee.

- Trong table Order có thuộc tính EmployeeID là khóa ngoại từ bảng Employe, và CustomerID là khóa ngoại từ bảng Customer.

- Trong table OrderDetail, có thuộc tính OrderID là khóa ngoại từ bảng Order.s

-Trong bảng Product có chứa thuộc tính IngredientID là khóa ngoại từ bảng Ingredient.

- Trong bảng Ingredient có chứa thuộc tính SupplierID là khóa ngoại từ bảng Suppplier.

- Các thuộc tính StoreID, CustomerID, EmployeeID, OrderID, ProductID, IngredientID, SupplierID phải là UNIQUE và NOT NULL

- Các thuộc tính StorePhone và CustomerPhone chứa 11 kí tự là các chữ số từ 0-9

1. Cài đặt vật lí

CREATE DATABASE CoffeeShop;

Use CoffeeShop;

CREATE TABLE Store (

StoreID int PRIMARY KEY NOT NULL,

StoreName Nvarchar(50) default null,

StoreEmail varchar(50) default null,

StorePhone char(11) CHECK (StorePhone like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]') default null,

StoreAddress varchar(50) default null

);

Create table Customer (

CustomerID int PRIMARY KEY NOT NULL,

CustomerName Nvarchar(50) default null,

CustomerPhone char(10) CHECK (CustomerPhone like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]') default null,

CustomerAddress Nvarchar(50) default null

);

Create table Employee (

EmployeeID int PRIMARY KEY NOT NULL,

EmpoyeeName Nvarchar (50) default null,

DOB Date UNIQUE CHECK (year(getDate()) - year(DOB) > 18) default null,

Gender char(1) CHECK (Gender IN ('M','F')) default null,

Salary Decimal default null,

StoreID int FOREIGN KEY REFERENCES dbo.Store(StoreID)

);

CREATE TABLE Supplier (

SupplierID int PRIMARY KEY NOT NULL ,

SupplierName Nvarchar(50) default null,

SupplierPhone char(10) UNIQUE CHECK (SupplierPhone like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]') default null,

SupplierAddress Nvarchar(50) default null

);

CREATE TABLE Ingredient(

IngredientID int PRIMARY KEY NOT NULL,

IngredientName Nvarchar(50) default null,

IngredientUnit Nvarchar(50) default null,

SupplierID int FOREIGN KEY REFERENCES dbo.Supplier(SupplierID)

);

CREATE TABLE Product(

ProductID int PRIMARY KEY NOT NULL,

ProductName Nvarchar(50) default null,

ProductUnit Nvarchar(50) default null,

Price Decimal default null,

);

CREATE TABLE IngredientProduct(

IngredientID int FOREIGN KEY REFERENCES dbo.Ingredient(IngredientID),

IngredientQuantity decimal(18,0) default null,

ProductID int FOREIGN KEY REFERENCES dbo.Product (ProductID)

);

CREATE TABLE Orders (

OrderID int PRIMARY KEY NOT NULL,

OrderDate Date Check (OrderDate LIKE 'hh:mm - yyyy/MM/dd') default null,

EmployeeID int FOREIGN KEY REFERENCES dbo.Employee (EmployeeID),

CustomerID int FOREIGN KEY REFERENCES dbo.Customer (CustomerID),

);

CREATE TABLE OrderDetail(

OrderID int FOREIGN KEY REFERENCES dbo.Orders(OrderID),

ProductID int FOREIGN KEY REFERENCES dbo.Product (ProductID),

ProductQuantity int NOT NULL

);

CREATE TRIGGER CheckIngredientQuantity

ON OrderDetail

AFTER INSERT

AS

BEGIN

DECLARE @totalQuantity DECIMAL(18, 0);

DECLARE @requiredQuantity DECIMAL(18, 0);

-- Calculate required quantity of ingredients

SELECT @requiredQuantity = SUM(ip.IngredientQuantity \* od.ProductQuantity)

FROM inserted AS od

INNER JOIN IngredientProduct AS ip ON od.ProductID = ip.ProductID;

-- Calculate total quantity of available ingredients

SELECT @totalQuantity = SUM(ip.IngredientQuantity)

FROM IngredientProduct AS ip

INNER JOIN inserted AS od ON od.ProductID = ip.ProductID;

-- Check if total quantity is less than required quantity

IF @totalQuantity < @requiredQuantity

BEGIN

RAISERROR ('Số lượng nguyên liệu không đủ cho việc sản xuất sản phẩm', 16, 1);

ROLLBACK TRANSACTION; -- Rollback transaction if condition is not met

END

END;