VIETNAM ACADEMY OF SCIENCE AND TECHNOLOGY
UNIVERSITY OF SCIENCE AND TECHNOLOGY OF HANOI

FINAL REPORT

Project: **Restaurant Management System Database Project using MySQL**

By

Nguyễn Thị Thảo - BI12-423 (Group leader)

Nguyễn Hoàng Dương- BI12-120

Phùng Đức Thái- BI12-396

Hứa Hải Minh- BI12-272

Phạm Hải Nam- BI12-307

Subject: Fundamentals of databases

Lecturer: Dr. Nguyen Hoang Ha

Hanoi, October, 2022

A. Introduction

1. Objective

Restaurant is a kind of business that serves people all over world with ready-made food. This database project aims to manage the average restaurant's activities and resources, such as customers, orders, dishes, waiters, bills and inventory. Information on flexibility in terms of adding and removing a single item, as well as categories, would be provided by the database design. A more appropriate data system for reporting and retrieving information resulting from data discharges into the stored data would also result from this database design. The services that are provided include food ordering by the customers through systems, customer information management, and waiter information management, ingredients stored in the inventory and food bills that customers need to pay after meals. In fact, the user can search for the needed dishes through this system according to price range and category of food and later they are willing to be capable of ordering the meal.

2. Purpose

This document provides high-level information on the specifications for the future database implementation to support the functional requirements for "restaurant" target database.

3. Scope, Approach, and Methods

The Database Design for the "Restaurant" is composed of definitions for database objects derived by mapping entities to tables, attributes to columns, unique identifiers to unique keys and relationships to foreign keys.
Our project scope is limited by the list of functional requirements. During design, these definitions may be enhanced in order to support the requirements of the "Restaurant" application listed in the requirements specifications.

4. Database Software Utilities and Developer Tools

No.	Software	Version	Comments
1	My SQL Workbench	8.0.31 64 bit	Database visualization and scripting tool

B. Database Specifications or User story

1. User story

Customer

As a customer: I should be able to look at the various food options available in the restaurant along with the price for each item.

As a customer: I should be able to select my desirable dishes and add them to my cart/order.

As a customer: I should have a cart containing all the chosen items so far, accessible all the time.

Restaurant Manager

Note: restaurant manager should be the further user who use that database in order to accomplish future appropriate work.

As a restaurant manager: I should be able to update new needed information about my restaurant, including chef information while hiring them successfully from the interview.

As a restaurant manager: I should be able to view weekly sales for my outlet.

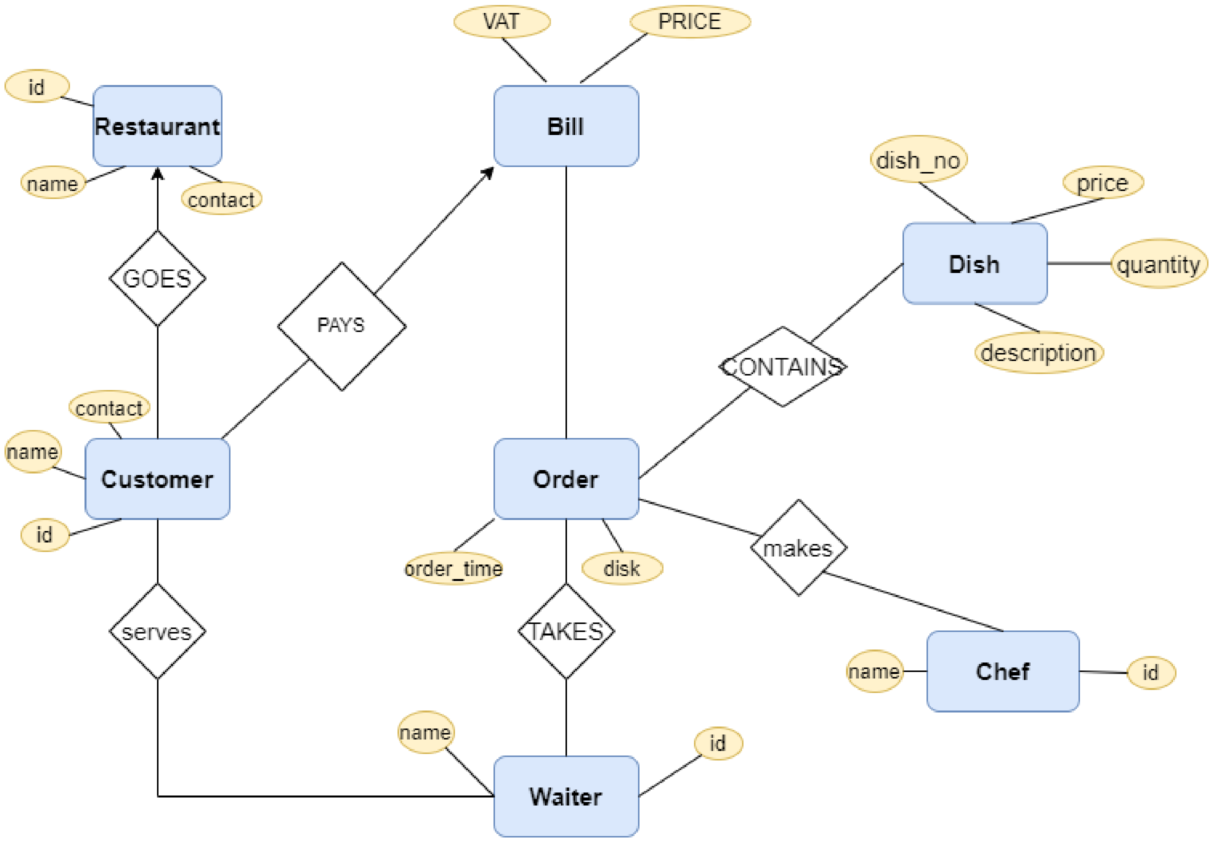

Figure 1: Entity Relationship Diagram (ERD) represents seven entities and their correlations

An Entity Relationship diagram (ERD) is used to represent the E-R model. It contains all known entities for the given scenario, their attributes, identifiers, and the relationships that exist among the entities.

As seen on figure above the ER diagram has seven entities which are made up of various attributes including the entity's primary key and other attributes like the name, ID, address and contact.

2. Normalization

- A restaurant has a unique restaurant ID(R_id), name(R_name), contact number (contact_no) and address.
- Many customers can go to one restaurant.
- A customer has a unique customer ID(cus_id), name(Cus_name), contact number(Contact_no).
- One waiter can serve more than one customer.
- A waiter has a unique ID(W_id), name (W_name)
- A customer orders through a waiter and the order is prepared by the chef.
- An order has a unique order number(order_no), number of items(no_items) and order time(order_time).
- A chef has unique id(Chef_id), name(Chef_name)
- An order contains a dish which has a unique dish number(dish_no), quantity,price and description.
- Customer pay bills that contains a unique bill number(bill_no), price, order detail(order_detail) and VAT.

3. Applied normalization

Relationship: Go

UNF: R_id, R_name, Contact_no, Address, Cus_id, Contact_no, Cus_name

1NF:

Customer- Cus_id, Cus_name, Contact_no, R_id,

Restaurant- R_id, R_name, Contact_no, address

2NF:

Customer- Cus_id, Cus_name, Contact_no, R_id,

Restaurant: R_id, R_name, Contact_no, address

3NF:

Restaurant- R_id, R_name, Contact_no, address

Restaurant_detail :RES_DETAIL_ID,R_id, R_name, address

Serves:

UNF: Cus_ID, Cus_name, Contact_no, W_id, W_name

1NF:

Customer: Cus_id, Cus_name, contact_no

Waiter: W_id, W_name,Cus_id

2NF:
Customer: Cus_id, Cus_name, contact_no
Waiter: W_id, W_name, Cus_id
3NF:
Customer: Cus_id, Cus_name, contact_no
Waiter: W_id, W_name, Cus_id

Makes:

UNF: chef_id, Chef_name, order_no, no_items, ord_time
1NF:
Chef: Chef_id, Chef_name, order_no
Order: order_no
2NF:
Chef: Chef_id, Chef_name, order_no
Order: order_no
3NF:
Chef: Chef_id, Chef_name, order_no
Order: Order_no

Relationship: Takes

UNF: W_id, W_name, order_no, no_items, order_time
1NF:
Waiter: W_id, W_name, Order_no
Order: Order_no, no_items, order_time
2NF:
Waiter: W_id, W_name, Order_no
Order: order_no, no_items, order_time
3NF:
Waiter: W_id, W_name, order_no
Order: order_no, no_items,
Order_info: no_items, order_time

Contains:

UNF: Order_no, ord_time, dish_no, quantity, price, description
1NF:
Order: Order_no
Dish: dish_no, quantity, price, description
2NF:
Order: order_no
Dish: dish_no, quantity, price, description
3NF:
Order: order_no
Dish: dish_no,

Dish: quantity, price,quantity,description

Pays:

UNF: Cus_name, Cus_id, contact_no, b_no, price, ord_detail, VAT

1NF:

Customer: Cus_id, Cus_name

Bill: b_no, price, ord_no,

2NF:

Customer: Cus_id, Cus_name

Bill: b_no, price, ord_no

3NF:

Customer: Cus_id, Cus_name

Bill: b_no, price, ord_no

Bill_detail: price, VAT

C. Relation model

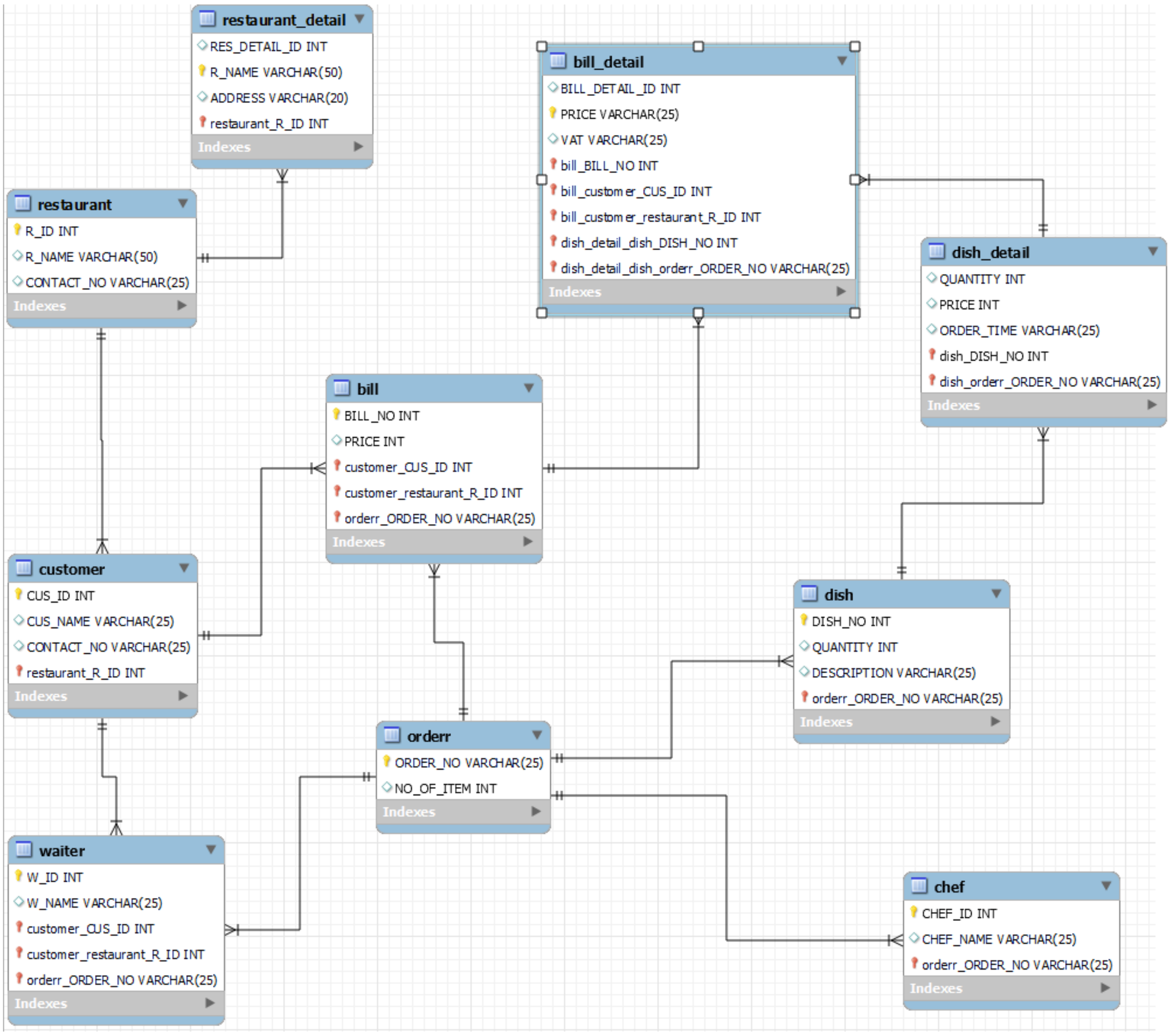

Figure 2: The schema is a graphical representation of relational databases' entities, attributes that are present in those entities and the relationship among these entities.

Restaurant

Restaurant data types		
Field Name	Data type	Text size
R_ID	INT	8
R_NAME	VARCHAR	50
CONTACT_NO	VARCHAR	25

Table 1: Data types of Restaurant

Restaurant_Detail

Restaurant_Detail data types		
Field Name	Data type	Text size
RES_DETAIL_ID	INT	8
R_ID	INT	8
R_NAME	VARCHAR	50
ADDRESS	VARCHAR	20

Table 2: Data types of Restaurant_Detail

Customer

Customer data types		
Field Name	Data type	Text size
CUS_ID	INT	10
CUS_NAME	VARCHAR	25
CONTACT_NO	VARCHAR	25
R_ID	INT	8

Table 3: Data types of Customer

Orderr

Order data types		
Field Name	Data type	Text size
ORDER_NO	VARCHAR	25

Table 4: Data types of Order

Waiter

Waiter data types		
Field Name	Data type	Text size
W_ID	INT	10
W_NAME	VARCHAR	25
CUS_ID	INT	10
ORDER_NO	VARCHAR	25

Table 5: Data types of Waiter

Dish

Dish data types		
Field Name	Data type	Text size
DISH_NO	INT	10
QUANTITY	INT	10
DESCRIPTION	VARCHAR	25

Table 6: Data types of Dish

Dish_Detail

Dish_Detail data types		
Field Name	Data type	Text size
DISH_DETAIL_ID	INT	10
DISH_NO	INT	10
QUANTITY	INT	10
PRICE	INT	10
ORDER_TIME	VARCHAR	25

Table 8: Data types of Dish_Detail

Chef

Chef data types		
Field Name	Data type	Text size
CHEF_ID	INT	10
CHEF_NAME	VARCHAR	25
ORDER_NO	VARCHAR	25

Table 9: Data types of Chef

Bill

Bill data types		
Field Name	Data type	Text size
BILL_NO	INT	10
PRICE	INT	10
ORDER_NO	VARCHAR	25
CUS_ID	INT	10

Table 10: Data types of Bill

Bill_Detail

Bill_Detail data types		
Field Name	Data type	Text size
BILL_DETAIL_ID	INT	8
DISH_DETAIL_ID	INT	10
BILL_NO	INT	10
PRICE	VARCHAR	25
VAT	VARCHAR	25

Table 11: Data types of Bill_Detail

D. Database Design and Functionalities

1. Design & Functional Support

The database will be designed in order to meet the below listed functional requirements.

Requirement		
ID	Name	Description
1	Easy information retrieval	Easy retrieval of information from Restaurant Database about clients (customers), dishes, bill, and orders.
2	Alteration	Possible to modify content of database.
3	Bill	Able to aggregate data in order to calculate the total price for the order.
4	Addition	Project members can insert data directly into the database.

5	Contact	Easy retrieval of information from database.
6	Food management	Increase your control over the variety of import and export items.

2. Assumptions of business situation

Situation 1: Improve interactions with customers (enhance the better position of customers to be very important persons - VIP)

- Based on the data, it is simple to identify customers who frequent the restaurants the most, or whose total bill is high or whose bills may require special attention.
- Our sales advertisements can focus customers' attention significantly through emails.
- Restaurant managements are able to identify new clients and offer them better initial service.
-

Situation 2: Specific decisions for future orientation

- Food trends may be forecasted and new meals added to the list by gathering information from dishes that were ordered.
- Managers can maximize profits by using data on food imports and popular dishes.

Managers can evaluate staff more objectively with the help of serving procedures and customer feedback.

E. Further developments

Being a start-up model, the restaurant database does not include online features, such as online reservations or email confirmations. The database created is a mere prototype to ensure the running of the restaurant daily routine. As the restaurant becomes more successful, an online system can be created to replace the current one.

F. Conclusion

In conclusion, the restaurant management system assists administrators to facilitate the workload of a restaurant employee to run the restaurant business and help customers with ordering tables. Ideally, this project aims to enhance customers' experience, along with restaurant management of administrators who directly manipulate employees' work.