[Food Delivery and Review Platform Database]

[Milestone 4]

[12/05/2024]

[CSC 675/775-01 AND Food Delivery and Review Platform Database]

[Thiha Aung]

[taung5@sfsu.edu]

Table of Contents

I. Project Description

A. Overview

I will create a project database to support a Food Delivery and Review Platform. The database will handle various functionalities such as managing customer profiles, restaurant menus, placing orders, assigning delivery drivers, and handling reviews. The project aims to connect users with local restaurants, offering an efficient and user-friendly way to order food online and provide feedback on the service.

B. What this project does

The project will implement a system that allows customers to browse restaurants, place orders, and write reviews after receiving their food. Restaurants can update their menus, view customer orders, and assign delivery drivers to fulfill orders. Delivery drivers will be notified of delivery assignments, and customers will be able to track their food delivery in real-time. The database ensures that all interactions between customers, restaurants, and drivers are managed efficiently.

C. Why is it needed

As the demand for food delivery services continues to grow, there is a need for a scalable and efficient system that can handle large amounts of data in real time. Restaurants often struggle to keep track of orders and manage customer feedback, while customers want a seamless and convenient experience when ordering food. This project will streamline these processes, improving service efficiency and customer satisfaction.

D. What Problem does it solve

The system solves the problem of managing complex data related to food orders, customer feedback, restaurant operations, and delivery logistics. By integrating all aspects into one database, the platform ensures that customer orders are processed quickly, delivery assignments are automated, and reviews are appropriately linked to the relevant restaurants.

E. Who does this benefit

This system benefits to:

1. Customers can get an easy-to-use interface to browse restaurants, place orders, and leave reviews.
2. Restaurant Owners can manage their menu, track orders, and monitor customer feedback.
3. Delivery Drivers can easily receive and manage their delivery assignments more efficiently.

4. System Administrators can oversee and manage the platform, and ensure smooth operations and resolve any issues conveniently.

II. Functional Database Requirements

A. List of Main Entities

1. Customer

   Customers can browse restaurants, place orders, and write reviews.

2. Restaurant

   Restaurants can have menus, receive orders.

3. Menu

   Menus can be found in restaurants.

4. Order

   Orders are placed by customers.

5. Order Item

   Order Items are specific items including in an order.

6. Delivery Driver

   Delivery drivers accept orders from restaurants and deliver to assigned customers.

7. Review

   Reviews are made by customers.

8. Payment

   Customers put their payment to place orders.

9. Address

   Address is where the orders are delivered.

10. Cuisine Type

    Menus can be subcategorized into different cuisine types.

11. Discount Code

    Discount Code can give a discount on final payment.

12. Restaurant Owner

Restaurant owners can manage their restaurants, menus and orders.

13. Order Status

Order status can be processed, ready or delivered.

14. Restaurant Location

Restaurant location is where orders to be picked up.

15. Delivery Assignment

Delivery Assignments are made to delivery drivers.

16. User Profile

Customers can put their information on their user profile.

B. List of Relationships

All Entities listed in Section A need to be listed at least once in this section.

1. Customer - Order

Customer has a One to many relationship with Order.

- One Customer can have many Orders.
- One Order can only belong to one Customer.

2. Order - Order Item

Order has a One to many relationship with Order Item.

- An Order includes many Order Items.
- An Order Item belongs to one Order.

3. Restaurant - Menu

Restaurant has One to many relationship with the menu.

- A Restaurant provides many Menus.
- A Menu belongs to one Restaurant.

4. Menu - Order Item

Menu has One to many relationship with the order item.

- A Menu contains many Order Items.
- An Order Item belongs to one Menu.

5. Customer - Review

Customer has One to many relationship with the review.

- A Customer writes many Reviews.
- A review belongs to one Customer.

6. Delivery Driver - Delivery Assignment

Delivery Driver has One to many relationship with Delivery Assignment.

- A delivery driver can have zero or many delivery assignments. .
- A delivery assignment belongs to a delivery driver.

7. Address - Delivery Assignment

Address has One to many relationship with the Delivery Assignment.

- An address can accept many delivery assignments.
- A Delivery Assignment belongs to an address.

8. Customer - Payment

Customer has One to many relationship with the Payment.

- A Customer can have many payment.
- A payment belongs to one Customer.

9. Restaurant - Cuisine Type

Restaurant has many to many relationship with the cuisine type.

- A Restaurant serves many cuisine types.
- A Cuisine Type is served by many restaurants.

10. Restaurant - DIscount Code

Restaurant has One to many relationship with the discount code.

- A Restaurant can have many discount codes.
- A discount code belongs to one restaurant.

11. Restaurant Owner - Restaurant

The Restaurant Owner has One to many relationship with the restaurant.

- A Restaurant Owner can have one or many restaurants.
- A restaurant belongs to one restaurant owner.

12. Order - Order Status

The Order has One to one relationship with the order status.

- An Order can have one order status at a time.
- An order status belongs to one order.

13. Restaurant - Restaurant Location

Restaurant has One to many relationship with the restaurant location.

- A Restaurant can have one or many restaurant locations.
- A restaurant location belongs to one restaurant.

14. Customer - User Profile

Customer has One to one relationship with the user profile.

- A Customer can have one user profile.
- An user profile belongs to one Customer.

15. Customer - Address

Customer has many to many relationship with the address.

- A Customer can have at least one or many addresses.
- An address can belong to one or many customers.

C. List of Actors

1. Customer

Customers can browse restaurants, place orders, and write reviews.

a) Browse restaurants

Customers can browse the list of available restaurants on the platform by filtering by cuisine, location, or popularity. They can view detailed restaurant profiles, including menus and customer reviews.

SELECT r.restaurant_id, r.restaurant_name, r.manager_id, r.email, rl.street, rl.city, rl.state, rl.zipcode \ FROM restaurant r JOIN restaurant_location rl ON r.restaurant_id = rl.restaurant_id;

b) Place orders

After selecting a restaurant, customers can add items to their cart and place an order. The system allows them to choose their payment.

INSERT INTO `order` (order_id, order_name, customer_id) VALUES (?, ?, ?);

c) write reviews

Customers can write reviews for the restaurant, providing feedback on the food quality, delivery service, and overall experience.

INSERT INTO review (review_id, time, ranking, customer_id) VALUES (?, ?, ?, ?);

2. Restaurant Owner

Restaurant owners manage their restaurant, update menus, and handle orders.

a) Update Menu

Restaurant owners can update menus by adding new items, removing old items, and modifying prices.

UPDATE menu SET menu_name = ? WHERE menu_id = ? AND restaurant_id = ?;

b) Handle Orders

Once an order is placed by a customer, restaurant owners receive the order details and can confirm, prepare, and assign to a delivery driver to fulfill the order.

UPDATE order_status SET status = 'Ready for Delivery'

WHERE order_status_id = ?;

UPDATE delivery_assign SET delivery_driver_id = ?, status = 'Assigned' WHERE delivery_assign_id = ?;

3. Delivery Driver

Delivery drivers pick up food orders from restaurants and deliver them to customers.

a) Accept Delivery Assignment

When a new delivery is assigned to the delivery driver. They can accept or reject delivery assignments based on their availability.

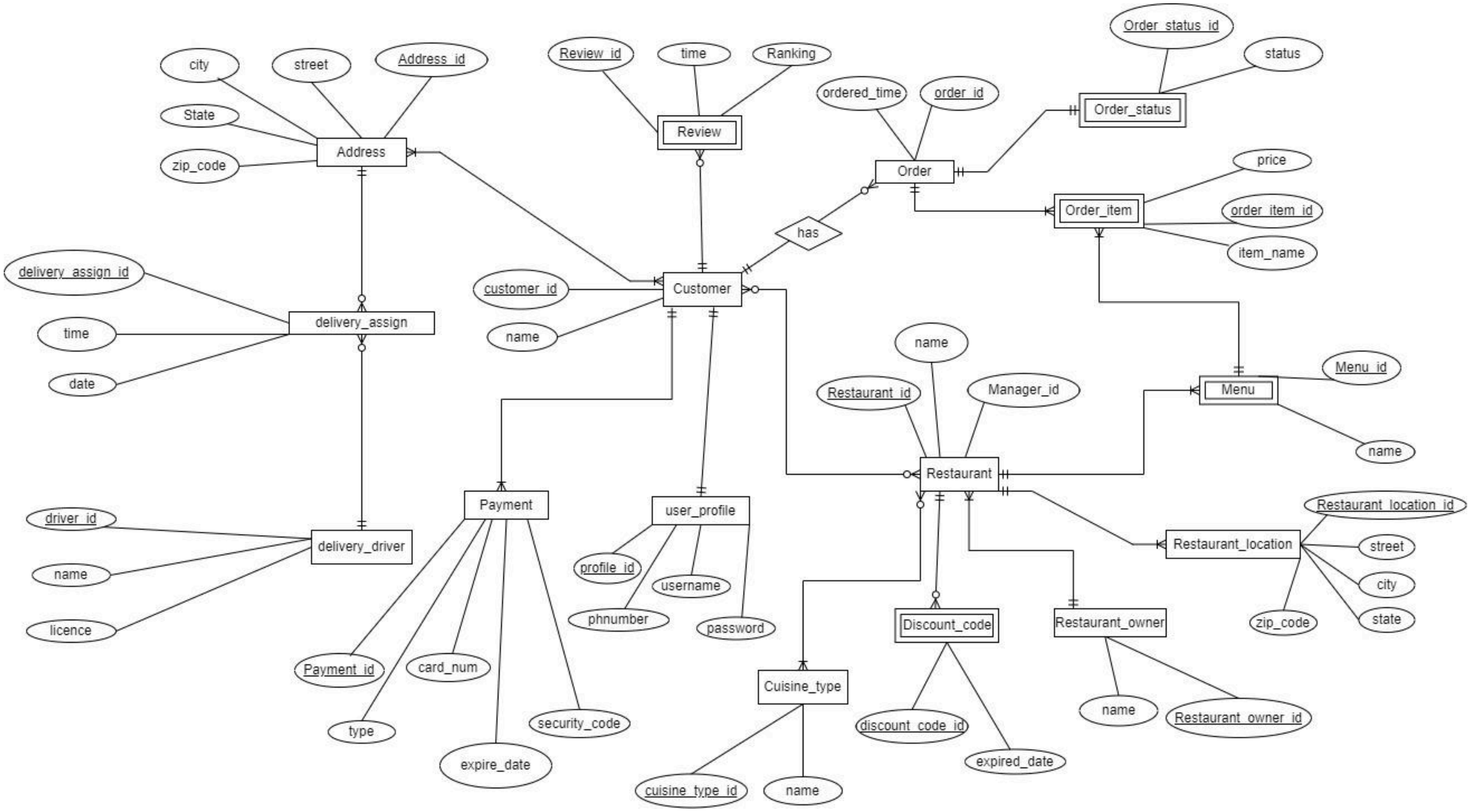
```
UPDATE delivery_assign SET delivery_driver_id = ?, status =
'Accepted' WHERE delivery_assign_id = ? AND status = 'Pending';
```

b) Deliver Order

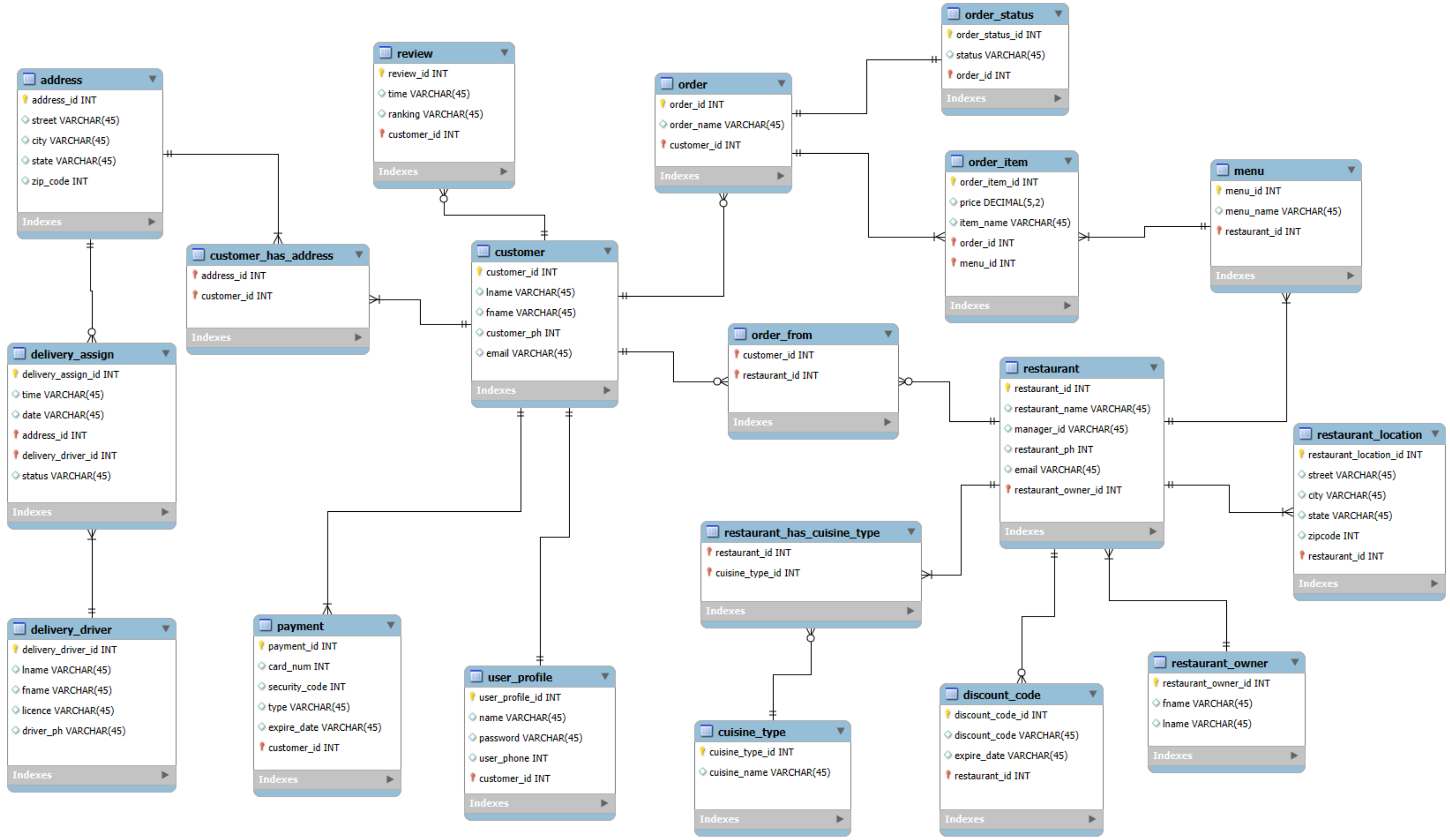
After accepting a delivery assignment, drivers collect the order from the restaurant and deliver it to the customer's specified address.

```
UPDATE delivery_assign SET delivery_driver_id = ?, status =
'Delivered' WHERE delivery_assign_id = ? AND status = 'Accepted';
```

III. Entity Relationship Diagram



A. Extended Entity Relationship Diagram (EERD)



IV. Non-Functional Database Requirements

A. Performance

The system must support 100,000 employees. It must handle simultaneous access by customers, restaurants, and delivery drivers without performance degradation.

All queries should respond within 0.20 seconds to ensure a smooth user experience, especially during peak hours when many users might be placing orders simultaneously.

B. Backup

Daily automatic backups of the entire database must be performed to protect against data loss. The orders, reviews, customer data, and delivery assignments will be stored in the database.

C. Security

Implement role-based access control (RBAC), ensuring that only authorized actors can access certain parts of the database. For example, customers should not access restaurant or driver-specific data, and vice versa.