

# DATABASE DESIGN FOR FOOD ORDERING SYSTEM

## MEMBERS:

Name	NUID
Anusha Das	002985875
Aishwarya Ankush Katkar	002961534
Rushikesh Dalvi	002922393
Aditya Agarwal	002139964
Maurya Ramteke	002980963

## PROBLEM STATEMENT

1. The restaurant finds it difficult to quickly manage online orders, further causing a discrepancy in data.
2. There is a need for an automated food ordering system to ease the process of handling customer orders, tracking restaurant sales, and checking on food deliveries, which will, in turn, help the restaurant manage online orders efficiently.

## OBJECTIVES

1. To increase efficiency and improve services provided by the restaurant by streamlining the process of online ordering and delivery.
2. Will allow restaurant owners to edit and display a restaurant's menu using the Food ordering system.
3. Keeping track of customer orders and deliveries.
4. To keep track of restaurant sales for analysis purposes.
5. To analyze the most popular dishes in the restaurant.
6. To keep track of regular customers.
7. To enable customers to have visual confirmation that the order was placed correctly.
8. Eliminate paperwork and increase accuracy, speed of service, sales volume, and customer satisfaction.

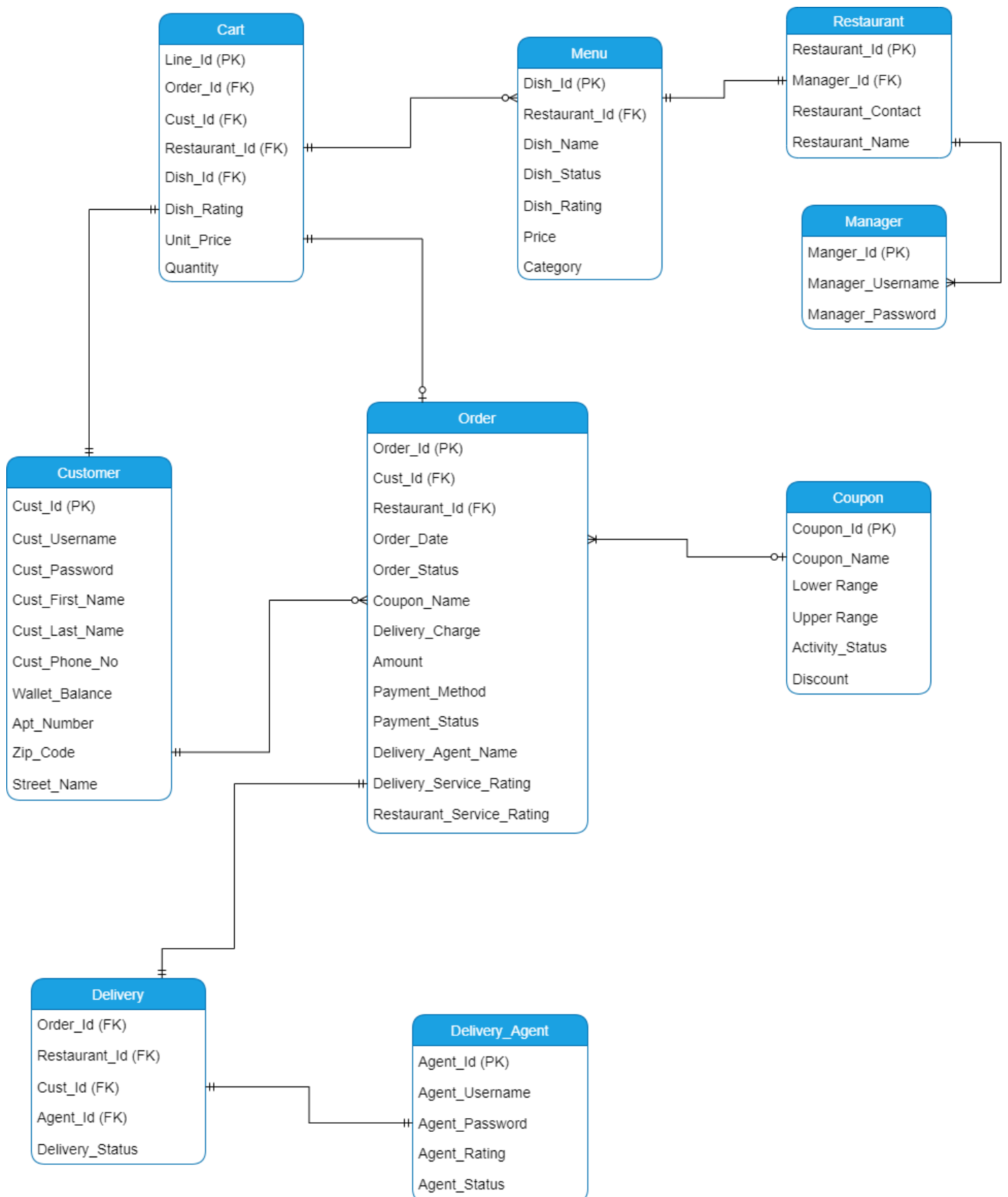
## PROPOSED SOLUTION

The above problems can be solved as follows:

1. Whenever a new customer is registered a new CUST\_ID will be generated to uniquely identify that customer. Likewise, we have separate tables MANAGER and DELIVERY\_AGENT which will be specific to manager and delivery agent respectively.

2. Using our database design, the Manager can obtain all the Customer order details, Menu details and Feedback details which we have defined as separate entities. Thus, the Manager can retrieve all the Customer and Menu details and update them with ease.
3. We have defined the entities in a way that whenever a transaction in PAYMENT table is complete, order data is inserted into ORDERS table with a PRIMARY key ORDER\_ID that uniquely identifies each order.
4. The 3 entities – ORDER, DELIVERY and DELIVERY\_AGENT are linked in such a way that whenever an order is placed, ORDER\_STATUS field is updated in ORDER table, and a delivery agent is assigned to that order in the DELIVERY table.
5. Once the food is delivered, Delivery Agent has access to change DELIVERY\_STATUS in Delivery table which in turn will update ORDER\_STATUS to “Completed” in ORDER table. This is possible by using keys ORDER\_ID and DELIVERY\_ID.
6. We have used the ORDER\_ID as a PRIMARY key in Payments table and as FOREIGN key in the ORDERS table to find sales of restaurant effectively.
7. The entity CUST\_WALLET is introduced to enhance the Payment functionality. Along with regular card payment, customers can use points in the CUST\_WALLET as and when required.

## REVISED ENTITY RELATIONSHIP DIAGRAM:



## ENTITIES & ATTRIBUTES

### CUSTOMER ENTITY

ATTRIBUTES	DATA TYPE AND SIZE	COMMENTS
CUST_ID	NUMBER	PRIMARY KEY, AUTO-GENERATED
CUST_USERNAME	VARCHAR (50)	NOT NULL
CUST_PASSWORD	VARCHAR (20)	NOT NULL
CUST_FIRST_NAME	VARCHAR (50)	NOT NULL
CUST_LAST_NAME	VARCHAR (50)	NOT NULL
CUST_PHONE_NO	NUMBER	NOT NULL
WALLET_BALANCE	NUMBER	NOT NULL
APT_NUMBER	NUMBER	NOT NULL
ZIP_CODE	NUMBER	NOT NULL
STREET_NAME	VARCHAR (50)	NOT NULL

### MENU ENTITY

ATTRIBUTES	DATA TYPE AND SIZE	COMMENTS
DISH_ID	NUMBER	PRIMARY KEY, AUTO-GENERATED
RESTAURANT_ID	NUMBER	FOREIGN KEY referencing RESTAURANT_ID from RESTAURANT table
DISH_NAME	VARCHAR (30)	NOT NULL
DISH_STATUS	VARCHAR (30)	NOT NULL
DISH_RATING	DECIMAL (10, 2)	NOT NULL
PRICE	DECIMAL (10, 2)	NOT NULL
CATEGORY	VARCHAR (30)	NOT NULL

### CART ENTITY

ATTRIBUTES	DATA TYPE AND SIZE	COMMENTS
LINE_ID	NUMBER	PRIMARY KEY, AUTO-GENERATED
ORDER_ID	NUMBER	FOREIGN KEY referencing ORDER_ID from ORDER table
CUST_ID	NUMBER	FOREIGN KEY referencing CUST_ID from CUSTOMER table
RESTAURANT_ID	NUMBER	FOREIGN KEY referencing RESTAURANT_ID from RESTAURANT table
DISH_ID	NUMBER	FOREIGN KEY referencing DISH_ID from MENU table
DISH_RATING	NUMBER	NULL
UNIT_PRICE	DECIMAL (10, 2)	NOT NULL
QUANTITY	NUMBER	NOT NULL

### ORDER ENTITY

ATTRIBUTES	DATA TYPE AND SIZE	COMMENTS
ORDER_ID	NUMBER	PRIMARY KEY, AUTO-GENERATED
CUST_ID	NUMBER	FOREIGN KEY referencing CUST_ID from CUSTOMER table
RESTAURANT_ID	NUMBER	FOREIGN KEY referencing RESTAURANT_ID from RESTAURANT table
ORDER_DATE	TIMESTAMP	CURRENT_TIMESTAMP
ORDER_STATUS	VARCHAR (50)	NOT NULL
COUPON_NAME	VARCHAR (20)	NOT NULL
DELIVERY_CHARGE	NUMBER	NOT NULL
AMOUNT	DECIMAL (10, 2)	NOT NULL
PAYMENT_METHOD	VARCHAR (20)	NOT NULL
PAYMENT_STATUS	VARCHAR (20)	NOT NULL
DELIVERY_AGENT_NAME	VARCHAR (50)	NOT NULL
DELIVERY_SERVICE_RATING	NUMBER	NOT NULL
RESTAURANT_SERVICE_RATING	NUMBER	NOT NULL

### COUPON ENTITY

ATTRIBUTES	DATA TYPE AND SIZE	COMMENTS
COUPON_ID	NUMBER	PRIMARY KEY, AUTO-GENERATED
COUPON_NAME	VARCHAR (30)	NOT NULL
LOWER_RANGE	NUMBER	NOT NULL
UPPER_RANGE	NUMBER	NOT NULL
ACTIVITY_STATUS	VARCHAR (30)	NOT NULL
DISCOUNT	DECIMAL (10, 2)	NOT NULL

### DELIVERY ENTITY

ATTRIBUTES	DATA TYPE AND SIZE	COMMENTS
ORDER_ID	NUMBER	FOREIGN KEY referencing ORDER_ID from ORDER table
RESTAURANT_ID	NUMBER	FOREIGN KEY referencing RESTAURANT_ID from RESTAURANT table
CUST_ID	NUMBER	FOREIGN KEY referencing CUST_ID from CUSTOMER table
AGENT_ID	NUMBER	FOREIGN KEY referencing AGENT_ID from DELIVERY_AGENT table
DELIVERY_STATUS	VARCHAR (30)	NOT NULL

### DELIVERY\_AGENT\_ENTITY

ATTRIBUTES	DATA TYPE AND SIZE	COMMENTS
AGENT_ID	NUMBER	PRIMARY KEY, AUTO-GENERATED
AGENT_USERNAME	VARCHAR (50)	NOT NULL
AGENT_PASSWORD	VARCHAR (50)	NOT NULL
AGENT_RATING	DECIMAL (10, 2)	NOT NULL
AGENT_STATUS	VARCHAR (30)	NOT NULL

### MANAGER\_ENTITY

ATTRIBUTES	DATA TYPE AND SIZE	COMMENTS
MANAGER_ID	NUMBER	PRIMARY KEY, AUTO-GENERATED
MANAGER_USERNAME	VARCHAR (50)	NOT NULL
MANAGER_PASSWORD	VARCHAR (50)	NOT NULL

### RESTAURANT\_ENTITY

ATTRIBUTES	DATA TYPE AND SIZE	COMMENTS
RESTAURANT_ID	NUMBER	PRIMARY KEY, AUTO-GENERATED
MANAGER_ID	NUMBER	FOREIGN KEY referencing MANAGER_ID from MANAGER table
RESTAURANT_CONTACT	NUMBER (10)	NOT NULL
RESTAURANT_NAME	VARCHAR (50)	NOT NULL

## **BUSINESS RULES:**

1. Our Food Delivery System currently manages only one RESTAURANT
2. Every CUSTOMER should have a unique username
3. CUSTOMER can place one order at a time
4. One ORDER can have only one CUSTOMER
5. 1% of every 'completed' order amount will be added to WALLET\_BALANCE
6. The refunded payment of orders will be reflected on WALLET\_BALANCE
7. Only the MANAGER role is eligible to provide a refund on orders to respective customers
8. There are 3 modes of PAYMENT\_METHOD:
  - 8.1. Debit Card
  - 8.2. Credit Card
  - 8.3. Wallet Balance
9. WALLET\_BALANCE can be used to pay order amount if  $WALLET\_BALANCE \geq AMOUNT$
10. DELIVERY\_CHARGE of \$2 is applied on orders below \$10
11. 1% TAX is applied on every order
12.  $ORDER\_AMOUNT (BILL) = PRICE * QUANTITY - DISCOUNT + DELIVERY\_CHARGE (if any) + TAX$
13. PAYMENT\_STATUS has 2 categories:
  - 13.1. Paid
  - 13.2. Pending
14. DELIVERY\_AGENT is assigned when ORDER\_STATUS is 'CONFIRMED'
15. CUSTOMER cannot add items to an order after ORDER\_STATUS is 'PLACED SUCCESSFULLY' or 'CONFIRMED' or 'PREPARING' or 'OUT FOR DELIVERY' or 'DELAY IN DELIVERY'
16. CUSTOMER can cancel the ORDER if the ORDER\_STATUS = 'CONFIRMED' and cannot cancel the placed ORDER if ORDER\_STATUS is 'PREPARING' or 'OUT FOR DELIVERY' or 'DELAY IN DELIVERY' or 'DELIVERED'
17. Only canceled orders are eligible for Refunds
18. Currently, the COUPONS table has 3 coupons: SILVER, GOLD, and PLATINUM with their respective discount rates. Manager can disable these coupons or add more coupons to the COUPONS table with different discount rates (in percentage value)
19. COUPON\_NAME can be applied on ORDER AMOUNT if the below conditions are met:
  - 19.1. SILVER: For  $AMOUNT > \$30$  and  $\leq \$50$ , 10% off on AMOUNT
  - 19.2. GOLD: For  $AMOUNT > \$50$  and  $\leq \$100$ , 20% off on AMOUNT
  - 19.3. PLATINUM: For  $AMOUNT > \$100$ , 30% off on AMOUNT
20. ACTIVITY\_STATUS in COUPON can be set to 'ACTIVE' or 'NOT ACTIVE' by the MANAGER
21. DISH\_RATING in MENU will show the average rating of that dish based on CUSTOMER FEEDBACK
22. DISH\_RATING, RESTAURANT\_SERVICE\_RATING, DELIVERY\_SERVICE\_RATING will be taken from the CUSTOMER for each ORDER
23. DISH\_STATUS will show the availability of dishes and can be updated by the MANAGER
24. MENU has the following CATEGORY of dishes:
  - 24.1. Appetizer
  - 24.2. Main Course
  - 24.3. Dessert

24.4. Beverage

25. ORDER\_STATUS has the following categories:

25.1. Placed Successfully

25.2. Confirmed

25.3. Preparing

25.4. Out for delivery

25.5. Canceled

25.6. Refunded

25.7. Delivered

25.8. Delay In Delivery

26. DELIVERY\_AGENT is assigned when order status is 'PREPARING'

27. The manager will update the ORDER\_STATUS to 'OUT FOR DELIVERY' once the order is prepared and a Delivery Agent is assigned

28. DELIVERY\_STATUS has the following categories:

28.1. Delivered

28.2. Delay In Delivery

29. AGENT\_STATUS categories can be 'AVAILABLE' or 'NOT AVAILABLE'

30. DELIVERY\_AGENT will update the DELIVERY\_STATUS

31. DELIVERY\_STATUS updates ORDER\_STATUS through triggers

32. One DELIVERY\_AGENT can deliver only one ORDER at a time

33. Only the MANAGER can add/remove delivery agents

34. One ORDER is delivered by only one DELIVERY\_AGENT

35. Only customers will give ratings

36. All given ratings will be on a scale of 1-5



## **VIEWS:**

### **vw\_dish\_sales**

- This view will show the sales of all dishes made by the restaurant.
- It is a view that will be created using CART, MENU, ORDER tables.
- It will be accessible only by the Manager and Admin roles.

### **vw\_top\_10\_dishes**

- This view will show the top 10 most selling dishes.
- This view is created using MENU and ORDER tables.
- The view is accessible to the Manager and Admin roles.

### **vw\_top\_10\_customers**

- This view will show the restaurant's 10 most frequent customers.
- This is created using the CUSTOMER and ORDER tables.
- The view is accessible to the Manager and Admin roles.

### **vw\_monthly\_sales**

- This view will show the monthly sales of the restaurant.
- This is created using the ORDER table.
- The view is accessible to the Manager and Admin roles.

### **vw\_top\_delivery\_person**

- This view will show the delivery person with the highest delivery rating.
- This view is created using the DELIVERY table.
- The view is accessible to the Manager and Admin roles.

### **vw\_delivery\_agent\_ratings**

- This view will show the rating for a specific agent id and name.
- It is a view that is created using DELIVERY\_AGENT, ORDER, DELIVERY table.
- This view will be available for Manager, Customer, and Admin roles.

### **vw\_restaurant\_service\_ratings**

- This view will show the restaurant's average customer service rating which will be calculated as an average of all customer service ratings of all the orders provided by that restaurant.
- It is a view created using the ORDER and RESTAURANT table.
- It is a view available for Manager, Customer, and Admin roles.

### **vw\_delivery\_agent\_status**

- This view will show the delivery agent's status.
- It is a view that will be created on the DELIVERY\_AGENT table.

- This view will be available for manager and admin roles.

#### **vw\_active\_orders**

- This will show all the active orders
- This view is created using Order Table
- It is accessible to Manager and Admin Roles

#### **vw\_completed\_orders**

- This will show all the completed orders
- It is created using Order Table
- This view is only accessible to the Manager and Admin Roles

#### **vw\_cancelled\_orders**

- This will show all the canceled orders
- This view is created using Order Table
- It can be accessed by Manager and Admin Roles

#### **vw\_all\_orders**

- This will show all the Orders which are active, completed, or canceled
- It is created using Order Table
- This view can be accessed by Manager and Admin Roles

#### **vw\_customer\_order\_address**

- This view will show the customer order address details.
- It is a view created on the DELIVERY table.
- This view will be available for Delivery Agent and Admin roles

#### **vw\_menu**

- This view will show the menu details.
- It is a view created on the MENU table.
- This view will be available for Manager and Admin roles

#### **vw\_coupons**

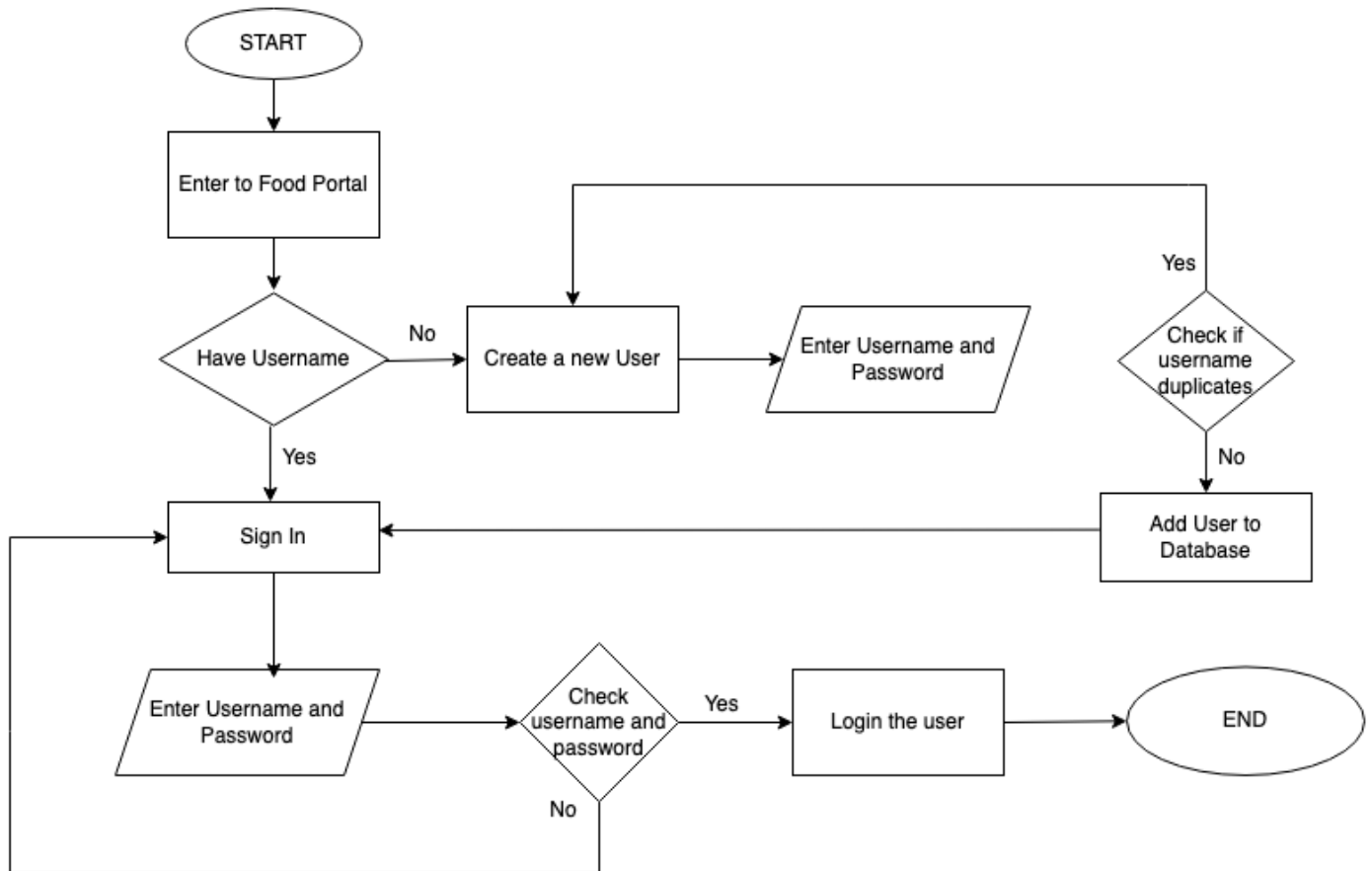
- This view will show all the available coupons that can be applied.
- It is a view that is created using the COUPONS table.
- This view will be accessible only to the Customer and Admin roles.

#### **vw\_restaurants**

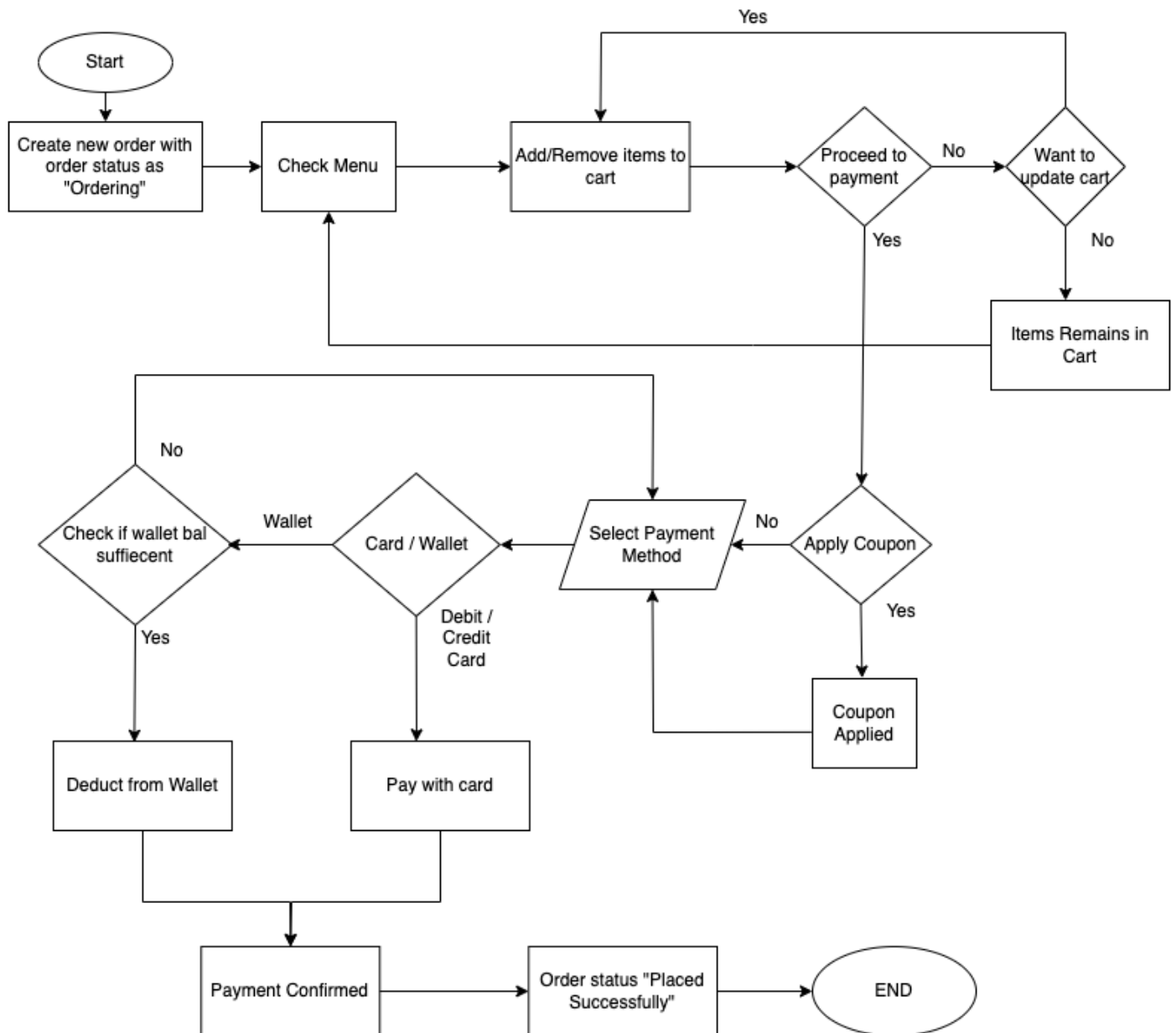
- This view will show all the restaurants.
- It is a view that will be created using the RESTAURANT table.
- It will be accessible only by Customer, Manager, and Admin roles.

## DATA FLOW DIAGRAMS

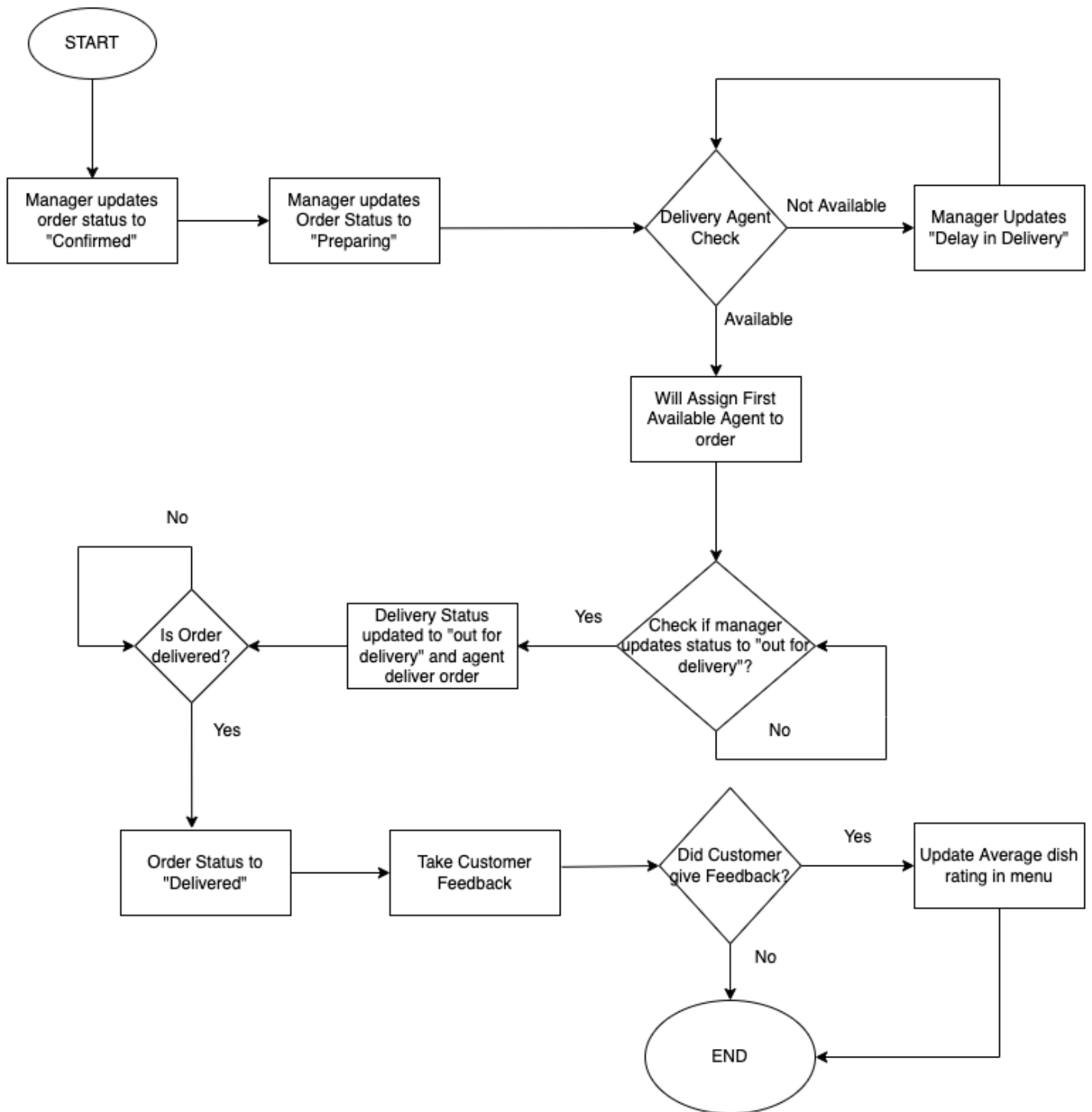
### LOGIN MODULE



## ORDER PROCESSING & PAYMENT MODULE



## DELIVERY MODULE



## **SECURITY CONSTRAINTS: (User level Access/Permissions)**

### **Customer:**

1. Does not have access to MANAGER, RESTAURANT, DELIVERY\_AGENT tables
2. Has READ/WRITE/UPDATE access to CUSTOMER table
3. Has only READ access to MENU table
4. Has READ/UPDATE access to CART table
5. Has READ/WRITE/UPDATE access to ORDER table
6. Has READ access to COUPON table

### **Manager:**

1. Has READ/WRITE/UPDATE access to MENU table
2. Has READ/WRITE access to COUPON table
3. Has READ/WRITE access to RESTAURANT table
4. Has READ/WRITE access to ORDER table
5. Has READ access to the DELIVERY table

### **Admin:**

1. Has full access to all the entities in the database

### **Delivery Agent:**

1. Has READ/WRITE access to DELIVERY\_AGENT table
2. Has UPDATE access to the DELIVERY table