

# Food Ordering Management Project

With an food ordering system, you can accept and manage online orders for delivery or takeaway orders. Customers view a digital menu, such as an app or website, and place and pay online. Venues then receive the order information via their preferred online food ordering platform and prepare the order for delivery or customer pick-up.

The Food Ordering System Database Management Project is a comprehensive software solution designed to streamline and optimize the operations of a food delivery and takeout service. This project encompasses the entire spectrum of managing customer information, cuisine categories, employees, chefs, ingredients, food items, beverages, deliveries, orders, and payments within the food ordering ecosystem.

## **Key Components and Functionalities:**

### **Customer Management:**

The system captures and maintains essential customer information, including personal details, contact information, allergies, and a unique customer ID for each user.

### **Cuisine Management:**

It allows for categorizing various cuisines, enabling efficient organization of food items based on their culinary styles and origins. Each cuisine is assigned a unique cuisine ID.

### **Employee Management:**

The system tracks information about employees involved in food preparation, delivery, and customer service. Employee details include names, contact information, salary, and a unique employee ID.

### **Chef Management:**

For food preparation and quality control, the project tracks chef details, linking them to specific cuisines. Chef information includes name, contact information, and a reference to their cuisine specialization and employee association.

### **Ingredient Management:**

Ingredients used in food preparation are cataloged, with each ingredient assigned a unique identifier for easy reference.

### **Food Item Management:**

The system provides an extensive database of food items available for order, encompassing details such as name, price, quantity, availability, and references to their respective cuisine, ingredients, and chefs.

### **Beverage Management:**

The project includes a section for managing beverage items, capturing data such as name, price, quantity, and availability.

**Delivery Management:**

The system oversees the logistics of food deliveries, maintaining information about delivery personnel, vehicles, charges, dates, and times. Deliveries are associated with specific customers and employees, ensuring efficient tracking and execution.

**Order Management:**

It records customer orders, calculating total costs and referencing individual food items, beverages, and delivery information. Each order is identified by a unique order ID.

**Payment Management:**

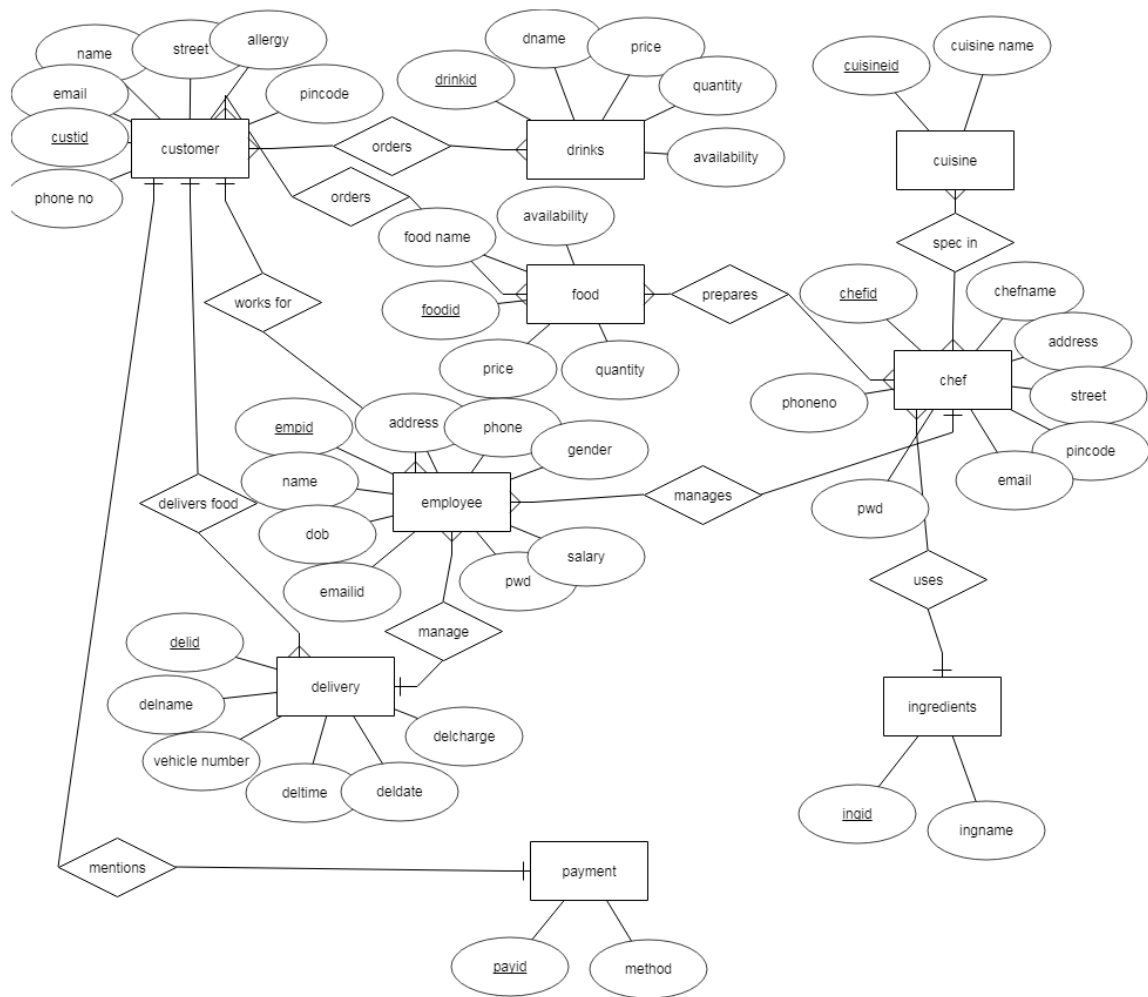
This component handles the financial aspect of the system, managing payment methods and connecting payments to specific customers and their respective orders.

The Food Ordering System Database Management Project not only facilitates efficient order processing but also assists in managing the entire food delivery ecosystem. The use of primary and foreign keys in the database design ensures data integrity, relationships between different entities, and efficient retrieval of information. This project serves as a fundamental tool for modernizing and enhancing the operations of food ordering services, delivering convenience to both customers and providers.

---

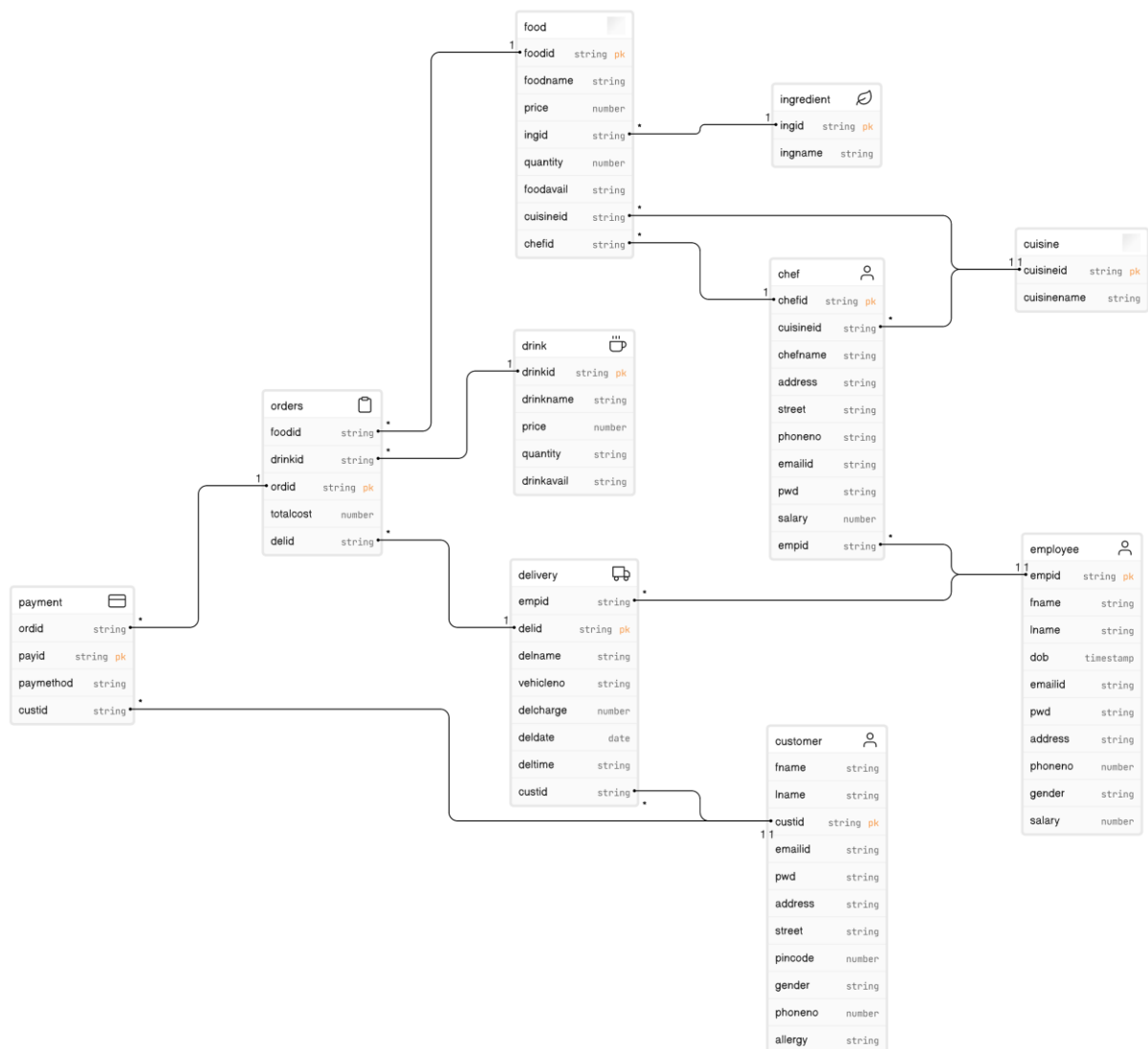
## Entity Relationship Diagram

An ER diagram is a visual representation that illustrates the entities (tables) within a database and the relationships between them. In the Online Food Ordering System Database Management System, an ER diagram would show how entities like customers, orders, chefs, and ingredients are connected, providing a holistic view of the database structure.



## UML Class diagram:

A class diagram is a visual representation of the classes, objects, and their relationships in an object-oriented system. In the context of the project, a class diagram would illustrate the various classes (e.g., Customer, Order, Chef) and their attributes and associations, helping to model the system's object-oriented design and structure.



## Entities and Tables:

### 1. Customer table:

The "customer" table stores information about the customers using the food ordering system. It includes details like first name, last name, customer ID (primary key), email address, address, phone number, allergies, and more. The customer ID serves as the primary key, ensuring each customer has a unique identifier.

- Entity: Customer
- Attributes:
  - First Name (fname): The first name of the customer.
  - Last Name (lname): The last name of the customer.
  - Customer ID (custid - Primary Key): A unique identifier for each customer.
  - Email Address (emailid): The customer's email address.
  - Password (pwd): The password associated with the customer's account.
  - Address (address): The street address of the customer.
  - Street (street): The street where the customer resides.
  - Pincode (pincode): The postal code of the customer's location.
  - Gender (gender): The gender of the customer.
  - Phone Number (phoneno): The customer's contact phone number.
  - Allergy (allergy): Any allergies or dietary restrictions the customer may have.

### 2. Cuisine table:

The "cuisine" table is responsible for storing different types of cuisines available in the system. It uses a "cuisineid" as a primary key, ensuring each cuisine has a unique identifier. Other tables can use this key as a reference (foreign key) to associate food items with specific cuisines.

- Entity: Cuisine
- Attributes:
  - Cuisine ID (cuisineid - Primary Key): A unique identifier for each cuisine category.
  - Cuisine Name (cuisinename): The name of the cuisine category.

### 3. Employee table:

The "employee" table manages information about the employees of the food ordering system, including first name, last name, employee ID (primary key), date of birth, email, address, phone number, and salary. The employee ID serves as the primary key, guaranteeing each employee's uniqueness.

- Entity: Employee
- Attributes:
  - Employee ID (empid - Primary Key): A unique identifier for each employee.
  - First Name (fname): The first name of the employee.
  - Last Name (lname): The last name of the employee.
  - Date of Birth (dob): The employee's date of birth.
  - Email Address (emailid): The email address of the employee.

Password (pwd): The password for the employee's account.  
Address (address): The street address where the employee resides.  
Phone Number (phoneno): The contact phone number of the employee.  
Gender (gender): The gender of the employee.  
Salary (salary): The employee's salary.

#### **4. Chef table:**

The "chef" table stores data about chefs responsible for preparing food items. It includes fields such as chef name, address, phone number, cuisine ID (foreign key), employee ID (foreign key), and more. The cuisine ID and employee ID are foreign keys that reference the "cuisine" and "employee" tables, respectively.

- Entity: Chef
- Attributes:
  - Chef ID (chefid - Primary Key): A unique identifier for each chef.
  - Chef Name (chefname): The name of the chef.
  - Address (address): The street address of the chef.
  - Street (street): The street where the chef resides.
  - Phone Number (phoneno): The contact phone number of the chef.
  - Cuisine ID (cuisineid - Foreign Key): A reference to the cuisine category the chef specializes in.
  - Employee ID (empid - Foreign Key): A reference to the employee associated with the chef.
  - Email Address (emailid): The email address of the chef.
  - Password (pwd): The password for the chef's account.
  - Salary (salary): The salary of the chef.

#### **5. Ingredient table:**

The "ingredient" table manages various ingredients used in food preparation. It employs an "ingid" as a primary key to ensure the uniqueness of each ingredient.

- Entity: Ingredient
- Attributes:
  - Ingredient ID (ingid - Primary Key): A unique identifier for each ingredient.
  - Ingredient Name (ingname): The name of the ingredient.

#### **6. Food table:**

The "food" table contains information about food items available for ordering. It includes attributes like food name, price, quantity, cuisine ID (foreign key), ingredient ID (foreign key), chef ID (foreign key), and more. The cuisine, ingredient, and chef IDs serve as foreign keys, linking the food items to their respective cuisines, ingredients, and chefs.

- Entity: Food
- Attributes:
  - Food ID (foodid - Primary Key): A unique identifier for each food item.

Food Name (foodname): The name of the food item.

Price (price): The price of the food item.

Quantity (quantity): The available quantity of the food item.

Food Availability (foodavail): Indicates if the food item is available or not.

Cuisine ID (cuisineid - Foreign Key): A reference to the cuisine category to which the food item belongs.

Ingredient ID (ingid - Foreign Key): A reference to the ingredient used in the food item.

Chef ID (chefid - Foreign Key): A reference to the chef responsible for preparing the food.

## **7. Drink table:**

The "drink" table stores details about available beverages. It includes fields like drink name, price, quantity, and availability. The "drinkid" acts as the primary key to ensure each beverage has a unique identifier.

- Entity: Drink
- Attributes:
  - Drink ID (drinkid - Primary Key): A unique identifier for each beverage.
  - Drink Name (drinkname): The name of the beverage.
  - Price (price): The price of the beverage.
  - Quantity (quantity): The available quantity of the beverage.
  - Drink Availability (drinkavail): Indicates if the beverage is available or not.

## **8. Delivery table:**

The "delivery" table manages information related to food delivery. It includes data about delivery personnel, such as name, vehicle number, delivery charge, and delivery date. It also includes customer ID (foreign key) and employee ID (foreign key), linking deliveries to customers and employees.

- Entity: Delivery
- Attributes:
  - Delivery ID (delid - Primary Key): A unique identifier for each delivery.
  - Delivery Name (delname): The name of the delivery personnel.
  - Vehicle Number (vehiclno): The number of the delivery vehicle.
  - Delivery Charge (delcharge): The delivery charge for the service.
  - Delivery Date (deldate): The date scheduled for the delivery.
  - Delivery Time (deltime): The estimated delivery time.
  - Customer ID (custid - Foreign Key): A reference to the customer who placed the delivery order.
  - Employee ID (empid - Foreign Key): A reference to the employee responsible for the delivery.

### 9. Orders table:

The "orders" table records information about customer orders. It includes details like the order ID (primary key), total cost, food ID (foreign key), drink ID (foreign key), and delivery ID (foreign key). These foreign keys connect orders to the specific food, drinks, and delivery associated with each order.

- Entity: Orders
- Attributes:
  - Order ID (ordid - Primary Key): A unique identifier for each customer order.
  - Total Cost (totalcost): The total cost of the order.
  - Food ID (foodid - Foreign Key): A reference to the food item included in the order.
  - Drink ID (drinkid - Foreign Key): A reference to the beverage included in the order.
  - Delivery ID (delid - Foreign Key): A reference to the delivery service associated with the order.

### 10. Payment table:

The "payment" table handles payment information for orders. It includes fields like payment method, customer ID (foreign key), and order ID (foreign key). The customer and order IDs are foreign keys, establishing connections between payments, customers, and their corresponding orders.

- Entity: Payment
- Attributes:
  - Payment ID (payid - Primary Key): A unique identifier for each payment transaction.
  - Payment Method (paymethod): The method used for payment.
  - Customer ID (custid - Foreign Key): A reference to the customer making the payment.
  - Order ID (ordid - Foreign Key): A reference to the order for which the payment is made.

These tables collectively form the foundation of the Online Food Ordering System Database Management System, facilitating efficient data storage and retrieval for the entire food ordering process.



**Here are some interesting SQL queries that you can use for the tables you've provided:**

- Retrieve the names of all customers who have allergies.  

```
SELECT fname, lname  
FROM customer  
WHERE allergy IS NOT NULL;
```
- Find the cuisines offered by a specific chef.  

```
SELECT c.cuisinename  
FROM cuisine c  
JOIN chef ch ON c.cuisineid = ch.cuisineid  
WHERE ch.chefname = 'Chef Mario';
```
- List the total number of ingredients for each cuisine  

```
SELECT c.cuisinename, COUNT(i.ingid) AS ingredient_count  
FROM cuisine c  
LEFT JOIN food f ON c.cuisineid = f.cuisineid  
LEFT JOIN ingredient i ON f.ingid = i.ingid  
GROUP BY c.cuisineid, c.cuisinename;
```
- Calculate the total cost of each order, including food and drinks.  

```
SELECT o.ordid, SUM(f.price + d.price) AS total_cost  
FROM orders o  
LEFT JOIN food f ON o.foodid = f.foodid  
LEFT JOIN drink d ON o.drinkid = d.drinkid  
GROUP BY o.ordid;
```
- Retrieve the names of customers who have placed orders through the "Express Delivery" service.  

```
SELECT c.fname, c.lname  
FROM customer c  
JOIN delivery d ON c.custid = d.custid  
WHERE d.delname = 'Express Delivery';
```
- List the chefs who have not prepared any food.  

```
SELECT ch.chefname  
FROM chef ch  
LEFT JOIN food f ON ch.chefid = f.chefid  
WHERE f.foodid IS NULL;
```
- Retrieve the names of all customers who are allergic to eggs  

```
select fname, lname  
From customer  
Where allergy="Peanuts";
```

- Find the ingredients added in Margherita Pizza  
select ingname  
From ingredient  
Where ingid in (select ingid from food Where foodname="Margherita Pizza");