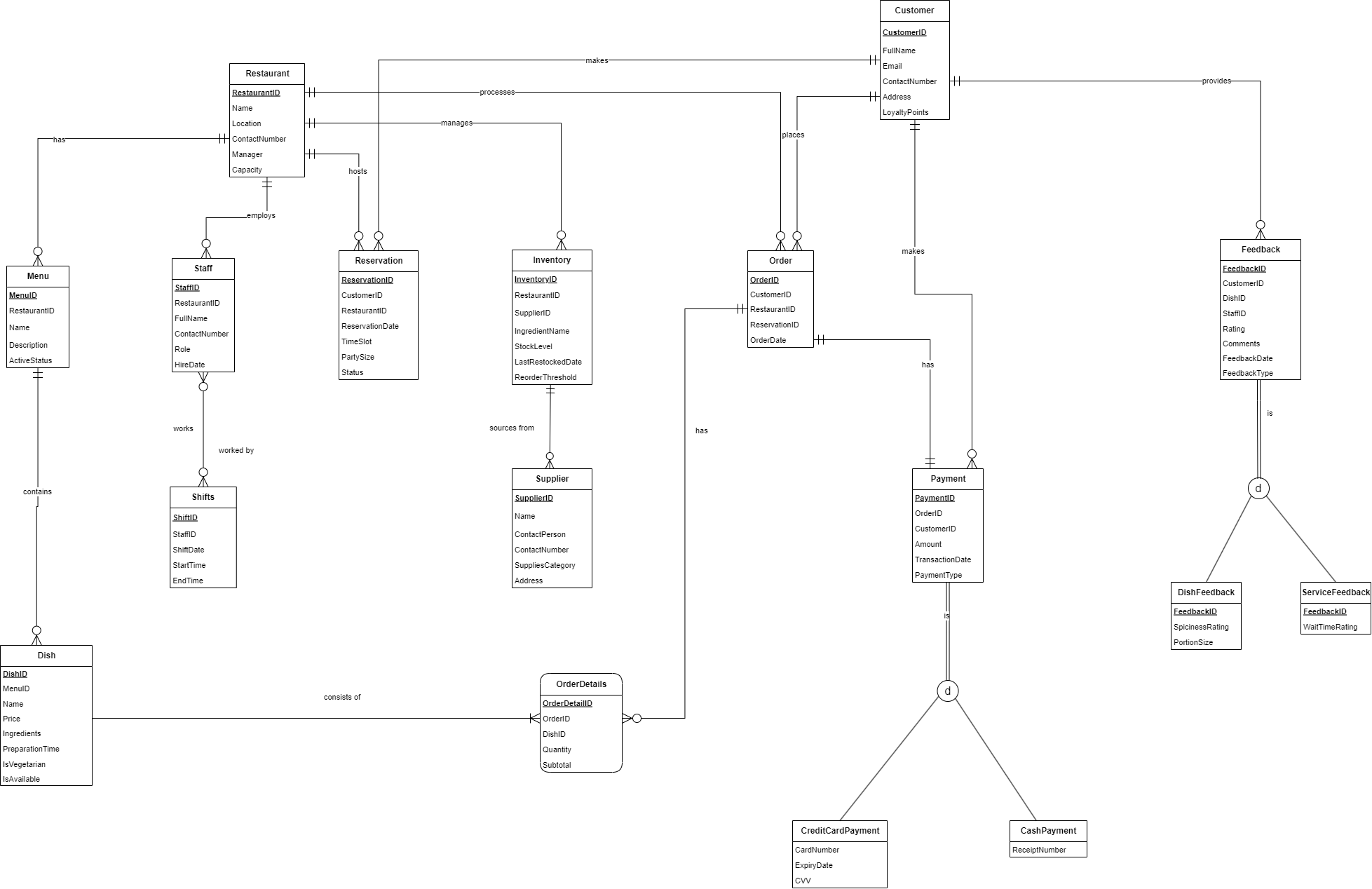
# P2 Database Design, Conceptual ERD

# Group 2:

1. Madhura Adadande
2. Kiran Sathya Sunkoji Rao
3. Harshitha Chennareddy
4. Yue Li
5. Sneh Patil



# Restaurant Database Design Document

## 1. Introduction and Business Objectives

This database design supports a small-to-mid-scale restaurant operation. The goal is to maintain a centralized system that manages:

• Reservations (tracking time slots, party size, and status)  
• Orders (including line-item details for dishes)  
• Menus (with dish information)  
• Inventory (stock level monitoring, supplier management)  
• Payments (supporting multiple payment methods)  
• Staff scheduling (shifts, roles)  
• Customer feedback (dish-specific and service-oriented)

By consolidating these functionalities, the restaurant can streamline daily workflows—ranging from keeping track of supply levels to scheduling staff—and leverage data-driven insights to enhance efficiency and customer satisfaction.

## 2. Business Problems Addressed

### Reservation Management

• Track when, where, and by whom reservations are made.

• Manage party size, time slots, and reservation status to optimize table usage.

### Menu and Ordering

• Centrally manage all dishes, including ingredients, preparation times, and availability.

• Allow customers (on-site or online) to place orders with detailed line items (dish quantity, subtotal).

### Inventory Control

• Monitor stock levels of ingredients, track the last restocked date, and set reorder thresholds for automatic alerts.

• Record supplier information to streamline procurement and maintain reliable supply chains.

### Staff and Shift Scheduling

• Keep detailed records of staff members, their roles, and assigned shifts.

• Ensure proper staffing levels and accountability for specific shifts (kitchen, front-of-house, etc.).

### Payments and Billing

• Support multiple payment methods (credit card, cash), each with unique attributes (e.g., card details, receipt).

• Provide an auditable record of transaction dates and amounts for accounting and analysis.

### Customer Feedback and Loyalty

• Capture both dish-specific and service‐related feedback for targeted improvements.

• Track loyalty points to reward repeat customers and encourage ongoing patronage.

## 3. Entities and Relationships

### Restaurant

Attributes:

* • RestaurantID (PK)
* • Name
* • Location
* • ContactNumber
* • Manager
* • Capacity

Relationships:

* • Manages → Inventory (1:N)
* • Employs → Staff (1:N)
* • Hosts → Reservation (1:N)
* • Has → Menu (1:N)
* • Has → Order (1:N)

### Staff

Attributes:

* • StaffID (PK)
* • RestaurantID (FK)
* • FullName
* • ContactNumber
* • Role
* • HireDate

Relationships:

* • Works → Shifts (1:N)

### Shifts

Attributes:

* • ShiftID (PK)
* • StaffID (FK)
* • ShiftDate
* • StartTime
* • EndTime

Relationships:

* • Worked By → Staff (N:1)

### Menu

Attributes:

* • MenuID (PK)
* • RestaurantID (FK)
* • Name
* • Description
* • ActiveStatus

Relationships:

* • Contains → Dish (1:N)

### Dish

Attributes:

* • DishID (PK)
* • MenuID (FK)
* • Name
* • Price
* • Ingredients
* • PreparationTime
* • IsVegetarian (boolean)
* • IsAvailable (boolean)

Relationships:

* • Belongs To → Menu (N:1)
* • Appears In → OrderDetails (1:N)

## 4. Key Database Design Decisions

* • Separating Menu and Dish
* • OrderDetails as an Associative Entity
* • Specialized Payment Subtypes
* • Feedback Specialization
* • Inventory and Supplier Separation
* • Shifts Tracking
* • Loyalty Points
* • Reservation Association
* • One or Many Payments per Order

## 5. Summary

This Restaurant Management Database consolidates all critical data—Reservations, Orders, Menus, Inventory, Payments, Staff, and Customer Feedback—into a single, consistent schema. By separating concerns (through specialized entities and subtypes) and following normalization principles, the design supports operational efficiency, scalability, data integrity, and better customer experience.