Chinese Restaurant Management System

Database Specification: Purpose, Business Problems Addressed and Business Rules

Team 9: Fang-Ying Tien, Jingzhi Zhang, Yaqing Peng, Yiyun Lyu, Yi-Chieh Huang

1. **Data Purpose:**
   The primary objective of this database is to systematically organize and preserve the critical data utilized by a restaurant for the management of its dish offerings, customer orders, shopping carts, and customer profiles. This database is designed for access by authorized restaurant staff and customers placing orders.

2. **Business Problems Addressed:**
   - Streamline the process of managing customer orders, dish offerings, and shopping carts to enhance operational efficiency.
   - Provide a systematic structure for organizing critical restaurant data, ensuring easy access and retrieval by authorized staff.
   - Facilitate seamless interaction between customers and the restaurant through order placement and management, enhancing overall customer experience.
   - Predict orders based on historical data to anticipate customer demands and optimize inventory management, thereby improving resource allocation and reducing waste.
   - Implement robust access controls to safeguard sensitive customer information and prevent unauthorized access to the database.

3. **Business Rules:**
   - Each dish has zero or one discount promotion
   - Each dish belongs to one category
   - Each dish belongs to zero or more set meals
   - Each shopping cart item has zero or one dish
   - Each shopping cart item has zero or one set meal
   - Each shopping cart item has zero or one flavor option
   - Each shopping cart item belongs to zero or one order detail
   - Each order has one or more order detail
   - Each order belongs to only one customer
   - Each order has only one billing
   - Each customer has zero or more payment accounts
   - Each payment account has only one billing address

4. **Design Requirements:**
   - Use Crow's Foot Notation
   - Specify the primary key field in each table by labeling "PK" besides the field
   - Specify the foreign key fields in each table by labeling "FK" besides each field

- Draw a line between each two tables to show the relationship between them
- Specify the cardinality and participation of each table by placing crow's feet symbols by the end of each line

5. Design Decisions:

Entity Name	Why is Entity Included	How is Entity Related
(1) Category	The entity facilitates efficient menu organization by categorizing dishes. It includes attributes such as categoryId for unique identification, categoryName for clear labeling, createTime and updateTime for tracking changes over time.	The entity is linked to the Dish entity forming a one-to-many relationship. This means that one dish category can contain multiple dishes.
(2) Dish	The Dish entity serves as a crucial component for organizing and managing the restaurant's menu offerings. Each dish has a unique Id, corresponding name, price, and availability status. Categorization into specific categories aids in efficient menu organization, while associated discounts are integrated for accurate pricing. Timestamps for creation and updates ensure operational efficiency.	The entity is linked to four entities. Firstly, it has a many-to-one relationship with Category, facilitating organized menu structuring where multiple dishes can belong to a single category. Secondly, regarding discounts, it operates under a many to zero-or-one relationship, allowing one discount to apply to multiple dishes. Each dish is directly associated with shoppingCartItem, representing a one to zero-or-one relationship, ensuring accurate order processing and tracking. Lastly, the entity is linked to the DishSetMeal entity, forming a one to zero-or-many relationship, permitting each dish to be bundled into multiple set meal options.
(3) DishSetMeal	The entity establishes connections between individual dishes and set meals. It links specific dishes with corresponding set meals, enabling	The entity acts as the intermediary, forming a many-to-many relationship between the Dish and SetMeal entities. Each dish can be bundled into

	efficient categorization and grouping. The status attribute indicates the availability of dishes within set meals. Timestamps track creation and updates, ensuring operational efficiency.	multiple set meal options, while various combinations of dishes can be associated with one particular set meal option.
(4) SetMeal	The entity defines bundled meal options offered by the restaurant. It includes attributes such as setMealName, price, status, and setMealDiscount, which collectively specify the details of the set meal. Additionally, timestamps track creation and updates.	The entity is linked to two entities. It's forming a one to zero-or-one relationship with shoppingCartItem, as each set meal corresponds to a single item in the shopping cart. Additionally, it indicates a one-to-many relationship with DishSetMeal, allowing various combinations of dishes in a set meal.
(5) Payment	The Payment entity is primarily included to document each customer's payment details, directly linking to the Customer and Address entities. This setup ensures that each payment is associated with a specific customer and their address, facilitating the management of billing and transaction histories. This structure is crucial for maintaining accurate and secure records of financial transactions within the system.	This entity is related to the Customer entity with a many-to-one relationship. Multiple payments can belong to one customer. It also has a one-to-one relationship with the Address entity, as each payment record has its own billing address.
(6) Flavor	The Flavor entity represents customer preferences for dish flavors, enhancing menu customization. It addresses varied taste desires, ensuring offerings meet individual preferences.The inclusion of createTime and updateTime attributes facilitates the tracking and updating of flavor options in response to customer	This Flavor entity is directly related to the ShoppingCartItem entity, forming a one to zero-or-many relationship. This setup allows for the specification of flavor preferences at the shopping cart level, enabling personalized selections and enhancing the customer experience.

	feedback and changing tastes, ensuring the menu remains dynamic and customer-focused.	
(7) Discount	Manage promotional offers on dishes to enhance sales and customer satisfaction. This involves specifying the level of discount and tracking the creation or modification of these offers, allowing for timely promotions and adjustments.	The Discount entity is associated with the Dish entity in one to zero-or-many relationship, allowing for discounts to be applied directly to the dishes. This relationship helps in dynamically adjusting prices based on promotional activities, providing flexibility in pricing strategies and offering value to customers. The createTime and updateTime attributes facilitate the management and tracking of discount validity and adjustment.
(8) OrderDetail	The OrderDetail entity is key for detailing each order, linking orders to the chosen items in the shopping cart. It ensures every selection by the customer is accurately documented and processed. The addition of "createTime" and "updateTime" attributes allows for monitoring when each order detail is created and modified, maintaining an up-to-date and precise record of order specifics.	The OrderDetail entity is associated with the ShoppingCartItem entity in a one-to-one relationship, meaning each OrderDetail corresponds to one, and only one, ShoppingCartItem. This signifies that every item selected in the shopping cart and included in an order has a unique order detail associated with it. Additionally, the OrderDetail entity has a many-to-one relationship with the "Order" entity, meaning each order can have multiple order details.
(9) Order	The Order entity is designed to store information about customer orders in the restaurant to track total price, tax, tips. The status determines whether order has completed. The createTime and updatedTime are also recorded.	The Order is related to the Customer entity with a many-to-one relationship. One or multiple orders belong to one customer. It also has a one-to-one relationship with the Billing entity, as each order has its own billing information. Additionally, it linked with

		the OrderDetail entity with a one-to-many relationship. One order may include multiple order detail data.
(10) Customer	The Customer entity is designed to store the profile of the restaurant's customers, including id which is the primary key, username, password, gender, email and phone number. The status determines whether the customer account information has been deleted. The createTime and updatedTime are also recorded.	The Customer entity is related to the Order entity in a one to zero-or-many relationship, as each customer may have zero or more orders. It also has a one to zero-or- many relationship with the Payment entity, representing the customer's payment information. It has a one to zero-or-many relationship with ShoppingCartItem as well, representing that each customer can own zero or many ShoppingCartItem.
(11) Address	The Address entity is to store the customer's billing address. This includes details, such as street, city, state and zip code. The createTime and updatedTime are also recorded.	The Address entity has a one to zero-or-one relationship with the Payment entity, as each address can be associated with zero or one payment.
(12) Billing	The Billing entity is included to capture information related to bills paid by customers for their orders. This includes details such as payment method. The billing status represents whether the customer has already paid for their order. The payment createTime and updateTime will also be recorded.	The Billing entity is related to the Order entity, representing the bill details for a specific order. It has a one-to-one relationship with the Order entity, as each order corresponds to a single bill.
(13)ShoppingCartItem	The ShoppingCartItem entity enables customers to add their desired dishes and set meals into a shopping cart while specifying their counts. The status marks whether this	The ShoppingCartItem entity is associated with the Dish entity and SetMeal entity in a one to zero-to-one relationship which means customers can add at most one dish or

	item has been added into order detail. The createTime and updateTime of the shopping cart items are also recorded.	set meal into the shopping cart at a time. It has a many to zero-or-one relationship with the Flavor entity. This entity is also linked to the OrderDetail entity in a one to zero-or-one relationship, allowing customers to decide whether to order the current item. It is related to the Customer with many to one relationship as well, which means one or many ShoppingCartItem belong to one customer. The count field records the number of the current item. The status specifies whether the item is added in the order detail.