

Database Design Document: Cloud Kitchen Management System

1. Description of Business Problems Addressed

Cloud kitchens operate multiple virtual restaurant brands from a single physical kitchen, creating unique operational challenges:

- **Operational Fragmentation:** Managing distinct menus, pricing, and branding for multiple virtual restaurants (e.g., "Burger Haven", "Pasta Palace") from one kitchen creates coordination complexity.
- **Resource Inefficiency:** Shared ingredient inventory across brands leads to food waste from poor expiration tracking and over-ordering. Without centralized visibility, ingredients expire unused or run out unexpectedly.
- **Logistical Bottlenecks:** Orders arrive simultaneously from multiple delivery platforms (Uber Eats, DoorDash, Grubhub) without centralized prioritization, causing delayed fulfillment and customer complaints.
- **Limited Analytics:** Absence of unified performance tracking prevents identifying which brands or menu items are profitable, optimizing ingredient costs, or improving kitchen capacity utilization.

2. List of Entities and Relationships

BRAND

Virtual restaurant operating from the cloud kitchen.

Relationships:

- One BRAND offers many MENU-ITEMs
- One BRAND receives many ORDERS

Justification: Enables separate performance tracking and branding for each virtual restaurant.

MENU-ITEM

Individual dishes offered by brands.

Relationships:

- Many MENU-ITEMs belong to one BRAND
- Many MENU-ITEMs use many INGREDIENTs (via RECIPE)
- Many MENU-ITEMs appear in many ORDER-ITEMs

Justification: Allows distinct menus per brand with recipe-based inventory tracking.

INGREDIENT

Raw materials used across all brands.

Relationships:

- One INGREDIENT is supplied by many SUPPLIERS
- One INGREDIENT is used in many RECIPES

Justification: Centralized tracking enables cross-brand inventory optimization and automatic low-stock alerts.

RECIPE

Specifications linking menu items to required ingredients with quantities.

Relationships:

- One RECIPE connects one MENU-ITEM to many INGREDIENTS (associative entity)

Justification: Enables automatic menu availability updates when ingredients are out of stock and supports inventory deduction when orders are placed.

CUSTOMER

Individuals ordering from any brand.

Relationships:

- One CUSTOMER places many ORDERS

Justification: Unified profiles enable cross-brand loyalty programs and comprehensive order history.

ORDER

Customer purchase requests from delivery platforms or direct channels.

Relationships:

- One ORDER is placed by one CUSTOMER
- One ORDER is for one BRAND
- One ORDER contains many ORDER-ITEMS
- One ORDER is assigned to one DELIVERY-PARTNER
- One ORDER generates one PAYMENT
- One ORDER is prepared at multiple KITCHEN-STATIONS

Attributes include: Customer feedback (rating, review text, review date)

Justification: Central transaction entity tracking order lifecycle and customer satisfaction.

ORDER-ITEM

Individual menu items within an order.

Relationships:

- Many ORDER-ITEMs belong to one ORDER
- One ORDER-ITEM references one MENU-ITEM

Justification: Supports multi-item orders and preserves historical pricing.

PAYMENT

Financial transactions for orders.

Relationships:

- One PAYMENT settles one ORDER

Justification: Separate entity enables refund management and tracks revenue from multiple payment sources.

DELIVERY-PARTNER

Third-party delivery services handling order fulfillment.

Relationships:

- One DELIVERY-PARTNER handles many ORDERS

Justification: Enables performance comparison, commission tracking, and cost optimization across delivery platforms.

SUPPLIER

Vendors providing ingredients.

Relationships:

- One SUPPLIER supplies many INGREDIENTs

Justification: Manages vendor relationships and supports multi-source procurement.

KITCHEN-STATION

Physical preparation areas (grill, salad prep, wok station).

Relationships:

- One KITCHEN-STATION is managed by many KITCHEN-STAFF

- Many ORDERS prepared across multiple KITCHEN-STATIONS

Justification: Enables workload distribution and capacity planning.

KITCHEN-STAFF

Employees working in the kitchen.

Relationships:

- Many KITCHEN-STAFF manage KITCHEN-STATIONS
- One KITCHEN-STAFF member prepares many ORDERS

Justification: Supports shift scheduling and role-based access control.

3. Key Design Decisions

The following decisions were made to ensure the conceptual model meets the business objectives:

- **Cross-Brand Inventory Integration**

Ingredients are centralized rather than separated by brand. This enables automatic low-stock alerts and usage tracking across all menus, reducing waste and preventing stockouts.

- **Recipe as Associative Entity**

RECIPE links MENU-ITEMs to INGREDIENTs with quantity specifications. If an ingredient runs out, menu items automatically become unavailable, preventing orders for items that cannot be prepared.

- **Delivery Partner Separation**

DELIVERY-PARTNER is standalone rather than just an ORDER attribute. This allows tracking of partner performance, commission rates, and delivery times for optimization.

- **Unified Customer Profiles**

Single CUSTOMER entity across all brands enables cross-brand loyalty programs and personalized marketing instead of treating customers separately per brand.

- **Payment Independence**

PAYMENT is separate from ORDER to handle refunds without deleting order records and to track revenue from multiple sources (cash, card, online, platforms).

- **Feedback in ORDER**

Customer ratings and reviews are stored as ORDER attributes rather than a separate entity, simplifying the model while maintaining feedback tracking capabilities.