

Схема базы данных системы заказа еды в ресторанах

(логическая и физическая модель)

1. Логическая модель

Логическая модель включает следующие сущности:

User — пользователь системы (клиент, сотрудник кухни, администратор меню).

Restaurant — ресторан (в системе может быть несколько ресторанов).

Category — категория блюд (основные блюда, закуски, напитки, десерты).

Dish — блюдо в меню.

Order — заказ клиента.

OrderItem — позиция в заказе.

OrderStatusHistory — история изменения статусов заказа.

Payment — информация об оплате заказа.

Notification — уведомление для пользователя.

Cart — корзина (временная или постоянная).

VerificationCode — код подтверждения телефона.

Бизнес-правила оформления заказа

Процесс оформления заказа (гость или зарегистрированный пользователь)

- Гость может зайти на сайт через QR код на столе.
- Для подтверждения заказа требуется ввод телефона с подтверждением через SMS код.
- После подтверждения телефона можно заполнить профиль (имя, дата рождения).
- Заказ создается со статусом 'pending' и переходит в 'получен' после подтверждения оплаты.
- Каждому заказу присваивается уникальный номер (например: ORD1-2024121-001).

Оплата заказа

- Оплата производится онлайн.
- При онлайн оплате система ожидает подтверждения от платежного шлюза.

- При неудачной оплате пользователь может повторить попытку.
- Статус заказа обновляется только после успешной оплаты.

Обработка заказа кухней

- После успешной оплаты заказ автоматически появляется на кухонном терминале.
- Сотрудник кухни меняет статус: 'Получен' → 'Готовится' → 'Готов'.
- Каждое изменение статуса фиксируется в истории и отправляет уведомление клиенту.

Связи между сущностями

- Один пользователь может иметь несколько заказов.
- Один заказ содержит несколько позиций (блюд).
- Одно блюдо принадлежит одной категории.
- Один заказ может иметь несколько статусов в истории.
- Один заказ имеет одну (или несколько попыток) оплаты.
- Одно блюдо может быть в нескольких заказах.
- Один ресторан имеет множество блюд в меню.
- Один пользователь может получать множество уведомлений.

2. Физическая модель

2.1 Таблица users (пользователи)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
phone	VARCHAR(20)	NOT NULL, UNIQUE
email	VARCHAR(255)	NULL
name	VARCHAR(100)	NULL
birth_date	DATE	NULL
password_hash	VARCHAR(255)	NULL (для сотрудников)

role	VARCHAR(30)	NOT NULL DEFAULT 'customer', CHECK (role IN ('customer', 'kitchen_staff', 'menu_admin', 'manager'))
is_phone_verified	BOOLEAN	DEFAULT FALSE
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

Индексы: idx_users_phone (phone), idx_users_role (role).

2.2 Таблица verification_codes (коды подтверждения)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
phone	VARCHAR(20)	NOT NULL
code	VARCHAR(6)	NOT NULL
purpose	VARCHAR(30)	NOT NULL DEFAULT 'phone_verify', CHECK (purpose IN ('phone_verify', 'password_reset'))
is_used	BOOLEAN	DEFAULT FALSE
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP
expires_at	TIMESTAMP	NOT NULL DEFAULT (CURRENT_TIMESTAMP + INTERVAL '5 minutes')

Индексы: idx_verification_phone_code (phone, code),
idx_verification_expires (expires_at).

2.3 Таблица restaurants (рестораны)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
name	VARCHAR(200)	NOT NULL
address	VARCHAR(500)	NOT NULL
phone	VARCHAR(20)	NOT NULL
logo_url	VARCHAR(500)	NULL
description	TEXT	NULL
is_active	BOOLEAN	DEFAULT TRUE
opening_hours	JSONB	NULL
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

2.4 Таблица categories (категории блюд)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
name	VARCHAR(100)	NOT NULL
description	TEXT	NULL
display_order	INTEGER	DEFAULT 0
icon_name	VARCHAR(50)	NULL
restaurant_id	INTEGER	FK → restaurants(id)
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

Индексы: idx_categories_restaurant (restaurant_id, display_order).

2.5 Таблица dishes (блюда)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
category_id	INTEGER	NOT NULL, FK → categories(id)
name	VARCHAR(200)	NOT NULL
description	TEXT	NULL
price	DECIMAL(10,2)	NOT NULL CHECK (price >= 0)
old_price	DECIMAL(10,2)	NULL CHECK (old_price >= 0)
ingredients	TEXT	NULL
nutrition_info	JSONB	NULL
weight_grams	INTEGER	NULL
is_available	BOOLEAN	DEFAULT TRUE
is_recommended	BOOLEAN	DEFAULT FALSE
is_spicy	BOOLEAN	DEFAULT FALSE
popularity_score	INTEGER	DEFAULT 0
image_url	VARCHAR(500)	NULL
preparation_time_min	INTEGER	NULL
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

Индексы: idx_dishes_category (category_id), idx_dishes_available (is_available), idx_dishes_popularity (popularity_score).

2.6 Таблица orders (заказы)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
user_id	INTEGER	NOT NULL, FK → users(id)
order_number	VARCHAR(50)	NOT NULL UNIQUE
restaurant_id	INTEGER	NOT NULL, FK → restaurants(id)
status	VARCHAR(30)	NOT NULL DEFAULT 'pending', CHECK (status IN ('pending', 'received', 'preparing', 'ready', 'completed', 'cancelled'))
delivery_type	VARCHAR(20)	NOT NULL DEFAULT 'at_table', CHECK (delivery_type IN ('at_table', 'pickup'))
table_number	VARCHAR(20)	NULL
total_amount	DECIMAL(10,2)	NOT NULL DEFAULT 0 CHECK (total_amount >= 0)
customer_phone	VARCHAR(20)	NOT NULL
customer_name	VARCHAR(100)	NULL
special_instructions	TEXT	NULL
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP
estimated_ready_at	TIMESTAMP	NULL

Индексы: idx_orders_status (status), idx_orders_restaurant (restaurant_id), idx_orders_created (created_at), idx_orders_number (order_number).

2.7 Таблица order_items (позиции заказа)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
order_id	INTEGER	NOT NULL, FK → orders(id) ON DELETE CASCADE
dish_id	INTEGER	NOT NULL, FK → dishes(id)
quantity	INTEGER	NOT NULL CHECK (quantity > 0)
unit_price	DECIMAL(10,2)	NOT NULL CHECK (unit_price >= 0)
special_instructions	TEXT	NULL
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

Индексы: idx_order_items_order (order_id), idx_order_items_dish (dish_id).

2.8 Таблица order_status_history (история статусов заказа)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
order_id	INTEGER	NOT NULL, FK → orders(id) ON DELETE CASCADE
status	VARCHAR(30)	NOT NULL
changed_by	INTEGER	NULL, FK → users(id)
notes	TEXT	NULL
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

Индексы: idx_status_history_order (order_id, created_at).

2.9 Таблица payments (оплата)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
order_id	INTEGER	NOT NULL, FK → orders(id) ON DELETE CASCADE
payment_method	VARCHAR(30)	NOT NULL DEFAULT 'card', CHECK (payment_method IN ('card', 'apple_pay', 'google_pay', 'cash'))
amount	DECIMAL(10,2)	NOT NULL CHECK (amount >= 0)
status	VARCHAR(30)	NOT NULL DEFAULT 'pending', CHECK (status IN ('pending', 'processing', 'success', 'failed', 'refunded'))
transaction_id	VARCHAR(255)	NULL UNIQUE
payment_gateway	VARCHAR(50)	NULL
gateway_response	JSONB	NULL
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

Индексы: idx_payments_order (order_id), idx_payments_status (status), idx_payments_transaction (transaction_id).

2.10 Таблица notifications (уведомления)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
user_id	INTEGER	NULL, FK → users(id)
phone	VARCHAR(20)	NULL
verification_code_id	SERIAL	FK→ verification_codes(id)
order_id	INTEGER	NULL, FK → orders(id)
type	VARCHAR(30)	NOT NULL, CHECK (type IN ('order_status', 'payment', 'verification', 'promotion', 'system'))
channel	VARCHAR(20)	NOT NULL, CHECK (channel IN ('sms', 'email', 'push', 'in_app'))
title	VARCHAR(200)	NULL
message	TEXT	NOT NULL
status	VARCHAR(20)	DEFAULT 'pending', CHECK (status IN ('pending', 'sent', 'delivered', 'failed', 'read'))
sent_at	TIMESTAMP	NULL
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

Индексы: idx_notifications_user (user_id), idx_notifications_phone (phone), idx_notifications_status (status).

2.11 Таблица cart (корзина - постоянная для авторизованных пользователей)

Поле	Тип	Ограничения
id	SERIAL	PRIMARY KEY
user_id	INTEGER	NOT NULL UNIQUE, FK → users(id)
items	JSONB	NOT NULL DEFAULT '[]'::jsonb
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

Индексы: idx_cart_user (user_id).

3. Порядок заполнения таблиц

1. restaurants — при разворачивании системы (минимум один ресторан).
2. categories — после создания ресторана.
3. dishes — добавление блюд в меню администратором.

4. `users` — регистрация сотрудников (кухня, администраторы) и клиентов (после подтверждения телефона).
5. `verification_codes` — при запросе кода подтверждения телефона.
6. `orders` — после оформления заказа:
 - Создается заказ со статусом 'pending'
 - Записываются `order_items`
7. `payments` — при инициировании оплаты:
 - 'pending' → 'processing' → 'success'/'failed'
8. `order_status_history` — автоматически при изменении статуса заказа.
9. `notifications` — при событиях:
 - Отправка кода подтверждения
 - Изменение статуса заказа
 - Успешная оплата
10. `cart` — для авторизованных пользователей при добавлении в корзину.