

# Архитектор базы данных

## 1 Основные сущности

### Таблица: users (Пользователи)

Назначение: хранение учетных данных, контактной информации и прав доступа.

- Поля:

- id (INTEGER) — уникальный идентификатор пользователя. PRIMARY KEY, генерируется последовательностью.
- name (VARCHAR) — отображаемое имя.
- email (VARCHAR(255)) — электронная почта (логин).
- phone (VARCHAR(32)) — телефон.
- password\_hash (VARCHAR) — хэш пароля.
- is\_admin (BOOLEAN) — флаг административных прав (доступ на добавление/удаление товаров).
- created\_at, updated\_at (TIMESTAMPTZ) — метки создания и обновления, created\_at по умолчанию now().

- Ограничения:

- users\_pk (PRIMARY KEY): id.
- NOT NULL: email, password\_hash, is\_admin, id, created\_at.

- Индексы:

- users\_email\_key (UNIQUE INDEX, BTREE): уникальность email.

### Таблица: product\_categories (Категории товаров)

Назначение: классификация товарного каталога.

- Поля:

- id (INTEGER) — PRIMARY KEY.
- name (VARCHAR) — название категории.
- description (TEXT) — описание.

- created\_at, updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- product\_categories\_pk (PRIMARY KEY): id.
- NOT NULL: id, name, created\_at.

- **Индексы:**

- product\_categories\_name\_key (UNIQUE INDEX, BTREE): уникальность названий.

## Таблица: products (Товары)

**Назначение:** карточки товаров охранных систем.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- sku (VARCHAR) — артикул.
- name (VARCHAR) — наименование.
- product\_type (VARCHAR) — тип изделия (камера, датчик, комплект и т.п.).
- price (DECIMAL(12,2)) — цена.
- availability\_status (VARCHAR) — статус наличия (in\_stock, preorder и т.п.).
- category\_id (INTEGER) — ссылка на product\_categories.
- created\_at, updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- products\_pk (PRIMARY KEY): id.
- products\_category\_id\_fkey (FOREIGN KEY): category\_id → product\_categories(id).
- NOT NULL: id, name, price, category\_id, created\_at.

- **Индексы:**

- products\_sku\_key (UNIQUE INDEX, BTREE): уникальность артикула.
- idx\_products\_category\_id (INDEX, BTREE): ускорение фильтрации по категории.

## Таблица: carts (Корзины)

**Назначение:** активная корзина пользователя.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- user\_id (INTEGER) — владелец корзины (users).
- created\_at (TIMESTAMPTZ).

- **Ограничения:**

- carts\_pk (PRIMARY KEY): id.
- carts\_user\_id\_fkey (FOREIGN KEY): user\_id → users(id).
- NOT NULL: id, user\_id, created\_at.

- **Индексы:**

- carts\_user\_id\_key (UNIQUE INDEX, BTREE): гарантирует одну корзину на пользователя.

## Таблица: cart\_items (Позиции корзины)

**Назначение:** товары, добавленные в корзину.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- cart\_id (INTEGER) — ссылка на carts.
- product\_id (INTEGER) — ссылка на products.
- quantity (INTEGER) — количество.
- price (DECIMAL(12,2)) — цена на момент добавления.
- created\_at, updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- cart\_items\_pk (PRIMARY KEY): id.
- cart\_items\_cart\_id\_fkey (FOREIGN KEY): cart\_id → carts(id).
- cart\_items\_product\_id\_fkey (FOREIGN KEY): product\_id → products(id).
- NOT NULL: id, cart\_id, product\_id, quantity, created\_at.

- **Индексы:**

- cart\_items\_pair\_key (UNIQUE INDEX, BTREE): (cart\_id, product\_id) — запрещает дубли.
- idx\_cart\_items\_cart\_id (INDEX, BTREE).

## Таблица: addresses (Адреса)

**Назначение:** адреса доставки и адреса объектов сервиса.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- city (VARCHAR), street (VARCHAR), house (VARCHAR), apartment (VARCHAR).
- comment (TEXT) — примечание.

- created\_at, updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- addresses\_pk (PRIMARY KEY): id.
- NOT NULL: id, city, street, house, created\_at.

- **Индексы:**

- idx\_addresses\_city (INDEX, BTREE).

## **Таблица: service\_objects (Объекты сервиса)**

**Назначение:** места установки/обслуживания систем.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- name (VARCHAR) — название объекта.
- object\_type (VARCHAR) — тип объекта (квартира, офис и т.п.).
- area (DECIMAL(12,2)) — площадь, м<sup>2</sup>.
- address\_id (INTEGER) — ссылка на addresses (1:1).
- created\_at, updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- service\_objects\_pk (PRIMARY KEY): id.
- service\_objects\_address\_id\_fkey (FOREIGN KEY): address\_id → addresses(id).
- NOT NULL: id, name, address\_id, created\_at.

- **Индексы:**

- service\_objects\_address\_id\_key (UNIQUE INDEX, BTREE): обеспечивает связь 1:1 с адресом.

## **Таблица: service\_requests (Сервисные заявки)**

**Назначение:** учет заявок на установку и ремонт.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- number (VARCHAR) — номер заявки.
- type (VARCHAR) — тип: установка/ремонт.
- status (VARCHAR) — статус исполнения.
- description (TEXT) — описание проблемы/работ.
- service\_object\_id (INTEGER) — ссылка на service\_objects.

- created\_at, updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- service\_requests\_pk (PRIMARY KEY): id.
- service\_requests\_service\_object\_id\_fkey (FOREIGN KEY): service\_object\_id → service\_object.
- NOT NULL: id, number, type, status, service\_object\_id, created\_at.

- **Индексы:**

- service\_requests\_number\_key (UNIQUE INDEX, BTREE).
- idx\_service\_requests\_service\_object\_id (INDEX, BTREE).
- idx\_service\_requests\_status (INDEX, BTREE).

## Таблица: install\_requests (Заявки на установку)

**Назначение:** специализация service\_requests для установок.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- service\_request\_id (INTEGER) — заявка-родитель.
- date (DATE) — дата установки.
- created\_at, updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- install\_requests\_pk (PRIMARY KEY): id.
- install\_requests\_service\_request\_id\_fkey (FOREIGN KEY): service\_request\_id → service\_requests(id).
- NOT NULL: id, service\_request\_id, date, created\_at.

- **Индексы:**

- install\_requests\_service\_request\_id\_key (UNIQUE INDEX, BTREE): связь 1:1 с заявкой.

## Таблица: repair\_requests (Заявки на ремонт)

**Назначение:** специализация service\_requests для ремонтов. Структура аналогична install\_requests.

- **Ключевые ограничения и индексы:** PRIMARY KEY id; FOREIGN KEY service\_request\_id → service\_requests(id); UNIQUE INDEX по service\_request\_id; NOT NULL для id, service\_request\_id, date, created\_at.

## Таблица: estimates (Сметы)

**Назначение:** согласование стоимости работ и материалов по заявке.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- service\_request\_id (INTEGER) — ссылка на заявку.
- labor\_cost (DECIMAL(12,2)) — стоимость работ.
- material\_cost (DECIMAL(12,2)) — стоимость материалов.
- status (VARCHAR) — статус сметы.
- created\_at, updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- estimates\_pk (PRIMARY KEY): id.
- estimates\_service\_request\_id\_fkey (FOREIGN KEY): service\_request\_id → service\_requests(id).
- NOT NULL: id, service\_request\_id, labor\_cost, material\_cost, created\_at.

- **Индексы:**

- estimates\_service\_request\_id\_key (UNIQUE INDEX, BTREE): 1:1 со служебной заявкой.

## Таблица: notifications (Уведомления)

**Назначение:** сообщения по ходу обработки заявок.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- service\_request\_id (INTEGER) — ссылка на заявку.
- text (TEXT) — содержимое уведомления.
- sent\_at (TIMESTAMPTZ) — время отправки.

- **Ограничения:**

- notifications\_pk (PRIMARY KEY): id.
- notifications\_service\_request\_id\_fkey (FOREIGN KEY): service\_request\_id → service\_requests(id).
- NOT NULL: id, service\_request\_id, text, sent\_at.

- **Индексы:**

- idx\_notifications\_request\_sent (INDEX, BTREE): (service\_request\_id, sent\_at DESC).

## Таблица: orders (Заказы)

**Назначение:** коммерческие заказы, сформированные по результатам заявок.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- number (VARCHAR) — номер заказа.
- created\_at (TIMESTAMPTZ) — дата оформления.
- status (VARCHAR) — статус заказа.
- total\_amount (DECIMAL(12,2)) — сумма заказа.
- service\_request\_id (INTEGER) — связь с service\_requests (обычно 1:1).
- address\_id (INTEGER) — адрес выполнения/доставки.
- updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- orders\_pk (PRIMARY KEY): id.
- orders\_service\_request\_id\_fkey (FOREIGN KEY): service\_request\_id → service\_requests(id).
- orders\_address\_id\_fkey (FOREIGN KEY): address\_id → addresses(id).
- NOT NULL: id, number, created\_at, total\_amount.

- **Индексы:**

- orders\_number\_key (UNIQUE INDEX, BTREE).
- idx\_orders\_created\_at (INDEX, BTREE).
- orders\_service\_request\_id\_key (UNIQUE INDEX, BTREE): обеспечивает связь 1:1 с заявкой.

## Таблица: deliveries (Доставки)

**Назначение:** параметры доставки по заказу.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- order\_id (INTEGER) — ссылка на orders.
- address\_id (INTEGER) — адрес доставки.
- type (VARCHAR) — тип доставки (курьер, самовывоз).
- cost (DECIMAL(12,2)) — стоимость.
- eta (TIMESTAMPTZ) — планируемая дата/время.

- **Ограничения:**

- deliveries\_pk (PRIMARY KEY): id.

- deliveries\_order\_id\_fkey (FOREIGN KEY): order\_id → orders(id).
- deliveries\_address\_id\_fkey (FOREIGN KEY): address\_id → addresses(id).
- NOT NULL: id, order\_id, type, cost.

- **Индексы:**

- deliveries\_order\_id\_key (UNIQUE INDEX, BTREE): связь 1:1 с заказом.

## Таблица: payments (Платежи)

**Назначение:** движения оплат по заказам.

- **Поля:**

- id (INTEGER) — PRIMARY KEY.
- order\_id (INTEGER) — ссылка на orders.
- method (VARCHAR) — метод оплаты.
- amount (DECIMAL(12,2)) — сумма.
- status (VARCHAR) — состояние («Зачислено», «Ожидает» и т.д.).
- paid\_at (TIMESTAMPTZ) — время проведения, может быть NULL.
- created\_at, updated\_at (TIMESTAMPTZ).

- **Ограничения:**

- payments\_pk (PRIMARY KEY): id.
- payments\_order\_id\_fkey (FOREIGN KEY): order\_id → orders(id).
- NOT NULL: id, order\_id, method, amount, status, created\_at.

- **Индексы:**

- idx\_payments\_order\_id (INDEX, BTREE).
- idx\_payments\_status (INDEX, BTREE).

## 2 Связующие таблицы и специализации

### Связующая: cart\_items

Реализует связь «многие-ко-многим» между carts и products. Составной уникальный индекс (cart\_id, product\_id) предотвращает дубли одной и той же позиции.

### Специализации service\_requests

Таблицы install\_requests и repair\_requests находятся в отношении 1:1 с service\_requests и расширяют заявку типоспецифичными полями (дата работ).

## 1:1 связи

- users — carts: уникальный user\_id в корзине.
- service\_objects — addresses: уникальный address\_id в service\_objects.
- service\_requests — estimates, orders, install\_requests, repair\_requests.
- orders — deliveries.

## 3 Порядок заполнения и бизнес-правила

### Порядок загрузки справочников и данных

1. product\_categories
2. products
3. users (включая признак is\_admin)
4. addresses, затем service\_objects
5. service\_requests и специализации (install\_requests/repair\_requests), далее estimates, notifications
6. carts и cart\_items
7. orders → deliveries, payments

### Ключевые бизнес-ограничения

- Один пользователь — одна активная корзина (carts.user\_id уникален).
- В корзине не может быть двух строк с одним и тем же товаром (cart\_items(cart\_id, product\_id) уникальны).
- Артикул товара (products.sku) и название категории (product\_categories.name) уникальны.
- Администратор (users.is\_admin = TRUE) имеет право создавать, изменять и удалять записи в products; для прочих пользователей операции ограничены политиками доступа.

### Рекомендованные индексы производительности

- orders(created\_at), payments(order\_id, status), notifications(service\_request\_id, sent\_at).
- Индексы по всем внешним ключам для ускорения JOIN.