**Описание структуры проекта IT-отдел**

**Backend (NestJS, Prisma)**

**/backend/src/**

**app.controller.ts / app.service.ts / app.module.ts / main.ts**

* Точка входа приложения. Базовый контроллер ("/" возвращает test-строку), сервис, главный модуль.

**/admin/**

* admin.controller.ts — Контроллер административных функций (например, управление пользователями).
* admin.service.ts — Логика админских операций.
* admin.module.ts — Модуль для админки.
* news.service.ts — Сервис для работы с новостями (может быть отдельно).

**/auth/**

* auth.controller.ts — Регистрация, логин, получение JWT.
* auth.service.ts — Аутентификация, генерация токенов, шифрование паролей.
* auth.module.ts — Модуль auth.
* jwt-auth.guard.ts — Guard для проверки JWT.
* jwt.strategy.ts — Стратегия извлечения пользователя из JWT.
* roles.decorator.ts / roles.guard.ts — Проверка ролей пользователя.

**/equipment/**

* equipment.controller.ts — CRUD для оборудования, загрузка файлов.
* equipment.service.ts — Бизнес-логика оборудования.
* equipment.module.ts — Модуль оборудования.

**/news/**

* news.controller.ts — CRUD для новостей.
* news.service.ts — Работа с новостями.
* news.module.ts — Модуль новостей.

**/notifications/**

* notification.controller.ts — Управление уведомлениями пользователя.
* notification.service.ts — Логика уведомлений.
* notification.gateway.ts — (Если есть) WebSocket-уведомления.
* notification.model.ts — Структура уведомления.
* notification.module.ts — Модуль уведомлений.

**/prisma/**

* prisma.module.ts / prisma.service.ts — Обёртка над Prisma ORM, доступ к базе.
* schema.prisma — Описание структуры всех таблиц/моделей.

**/requests/**

* requests.controller.ts — CRUD заявок, работа с комментариями, статусы.
* requests.service.ts — Логика заявок.
* requests.module.ts — Модуль заявок.
* comments.controller.ts / comments.service.ts — Работа с комментариями (если отдельно).
* dto/ — DTO для заявок.
* enums/ — Перечисления статусов.

**/software/**

* software.controller.ts — CRUD ПО.
* software.service.ts — Логика по ПО.
* software.module.ts — Модуль ПО.
* dto/ — DTO для ПО.

**/telegram/**

* telegram.bot.ts — Бот, команды Telegram.
* telegram.service.ts — Отправка сообщений, логика Telegram API.
* telegram.module.ts — Модуль Telegram.
* telegram.update.ts — (если есть) обработка команд.

**/users/**

* users.controller.ts — CRUD пользователей, профиль.
* users.service.ts — Бизнес-логика пользователей.
* users.module.ts — Модуль пользователей.
* users.controller.spec.ts / users.service.spec.ts — Тесты.

**/uploads/**

* /equipment/ — Файлы, прикреплённые к оборудованию.

**/backup/**

* it\_otdel\_backup.sql — Бэкап базы данных.

**Frontend (React, TypeScript)**

**/frontend/components/**

* RegisterForm.tsx — Форма регистрации.
* RequestCard.tsx — Карточка заявки.
* RequestDetailsModal.tsx — Модальное окно подробностей заявки.
* RequestFilters.tsx — Фильтрация заявок.
* SoftwareTable.tsx — Таблица ПО.
* SoftwareDetailsCard.tsx — Подробная карточка ПО.
* SoftwareFormModal.tsx — Модалка для создания/редактирования ПО.
* UserCard.tsx — Краткая карточка пользователя.
* UserDetailsCard.tsx — Подробная карточка пользователя.
* UserEditForm.tsx — Форма редактирования пользователя.
* UserFormModal.tsx — Модалка создания/редактирования пользователя.
* UserSelect.tsx / UserAutocomplete.tsx / UserSelectCard.tsx — Выбор пользователя (автокомплит, карточка выбора).
* NewRequestModal.tsx — Модалка создания заявки.
* TileCard.tsx / SkeletonCard.tsx — Универсальная карточка, Skeleton для загрузки.
* notifications/NotificationBell.tsx — Колокольчик уведомлений.
* notifications/NotificationList.tsx — Список уведомлений.
* ui/button.tsx — Кастомная кнопка для UI.

**/frontend/hooks/**

* useNotifications.ts — Хук для уведомлений (fetch, subscribe, loading/error state).
* useUserRole.ts — Хук для роли пользователя.

**/frontend/types/**

* user.ts — Типы пользователя, роли, связи.
* equipment.ts — Типы оборудования.
* software.ts — Типы ПО.
* types.ts — Общие типы (enum’ы, справочники, статусы).
* react-csv.d.ts — Типизация для экспорта в CSV.

**/frontend/utils/**

* constants.ts — Константы, enum’ы, справочники.

**/frontend/public/**

* file.svg, globe.svg, next.svg, vercel.svg, window.svg — SVG-иконки.

**Формат документации для каждого файла:**

1. Назначение файла/компонента.
2. Где и как используется.
3. Входные параметры (props, аргументы функций).
4. Связанные сущности/типы.
5. Краткое описание логики (1–2 предложения).
6. Пример использования (если компонент).

**Рекомендация:**

* В начале каждого файла добавлять JSDoc-комментарий (описание, props, пример использования).
* Поддерживать эту структуру в docs/ или в README.md модуля.