



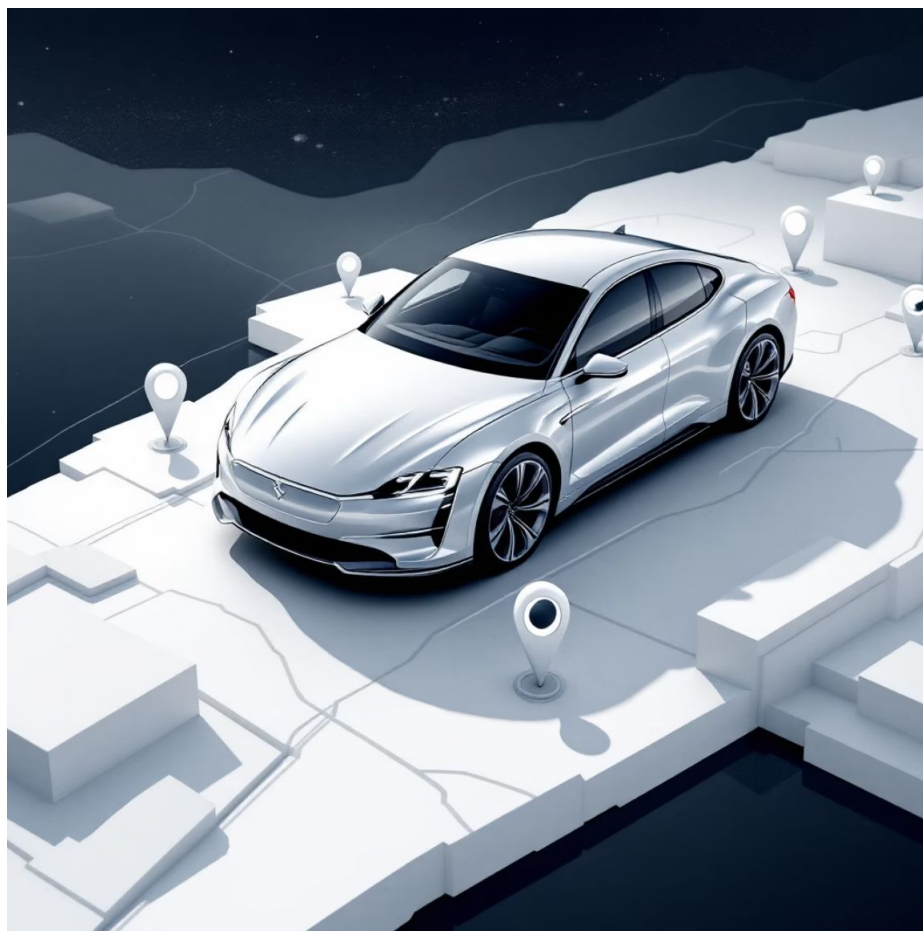
Проект CarShare: Техническая Документация

CarShare — это сервис аренды автомобилей с автоматическим поиском ближайших доступных машин. Система позволяет пользователям регистрироваться, находить автомобили, оформлять аренду и производить оплату.

Назначение и Предметная Область

Назначение проекта

Сервис каршеринга с функцией автоматического поиска ближайших доступных автомобилей.



Автоматизация бизнес-процессов

- Регистрация и аутентификация пользователей
- Управление парком автомобилей
- Поиск и оформление аренды
- Управление платежами
- Генерация отчетов





Основные Требования

Функциональные

- Регистрация и аутентификация (JWT)
- CRUD операции для автомобилей
- Поиск и оформление аренды
- Интеграция с платежной системой
- Административный интерфейс
- Экспорт данных в Excel

Нефункциональные

- Производительность (до 2 секунд)
- Безопасность (хеширование, JWT, защита от SQL)
- Масштабируемость (модульная архитектура)
- Надежность (обработка ошибок, логирование)
- Поддержка веб-интерфейса



Архитектурные Решения

1

Клиент-Серверная Архитектура

Система построена на базе FastAPI: Client → FastAPI Application → Routers → Controllers → Models → Database.

2

Компоненты

API Layer (FastAPI), Business Logic (router), Data Access (SQLModel), Authentication (JWT, OAuth2), Services (почта, отчеты).

Моделируемые Сущности (Model)

1

User
(Пользователь)
id, hash_password, role, email, first_name, license, date_reg.

2

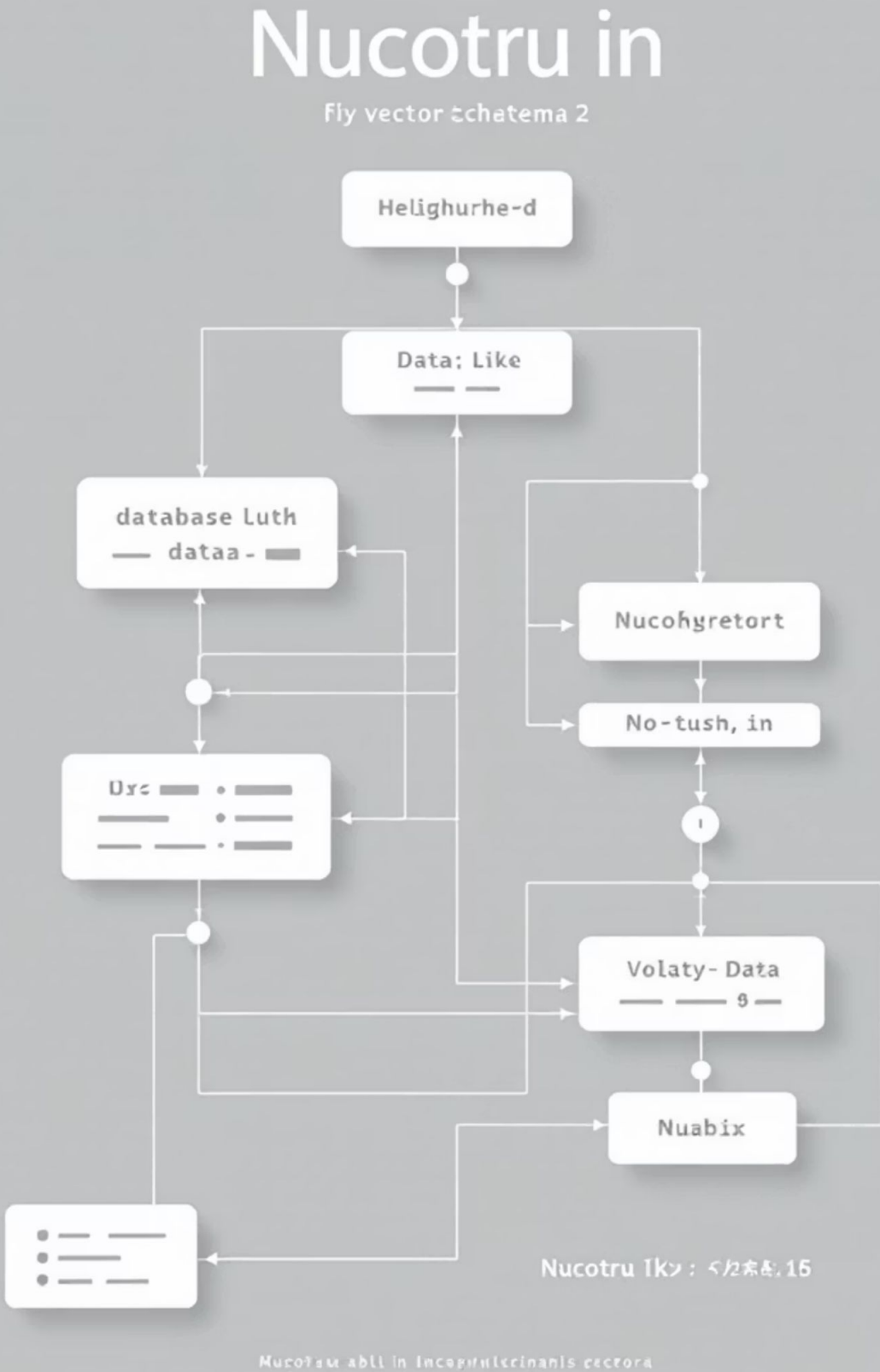
Car (Автомобиль)
id, brand, model, latitude, longitude, car_number, price_order, status.

3

Rent (Аренда)
id, user_id, car_id, data_rent_start, data_rent_end, status.

4

Payment (Платеж)
id, rent_id, user_id, prise, card_number, data, status.



Представление (View) и Контроллеры (Controller)

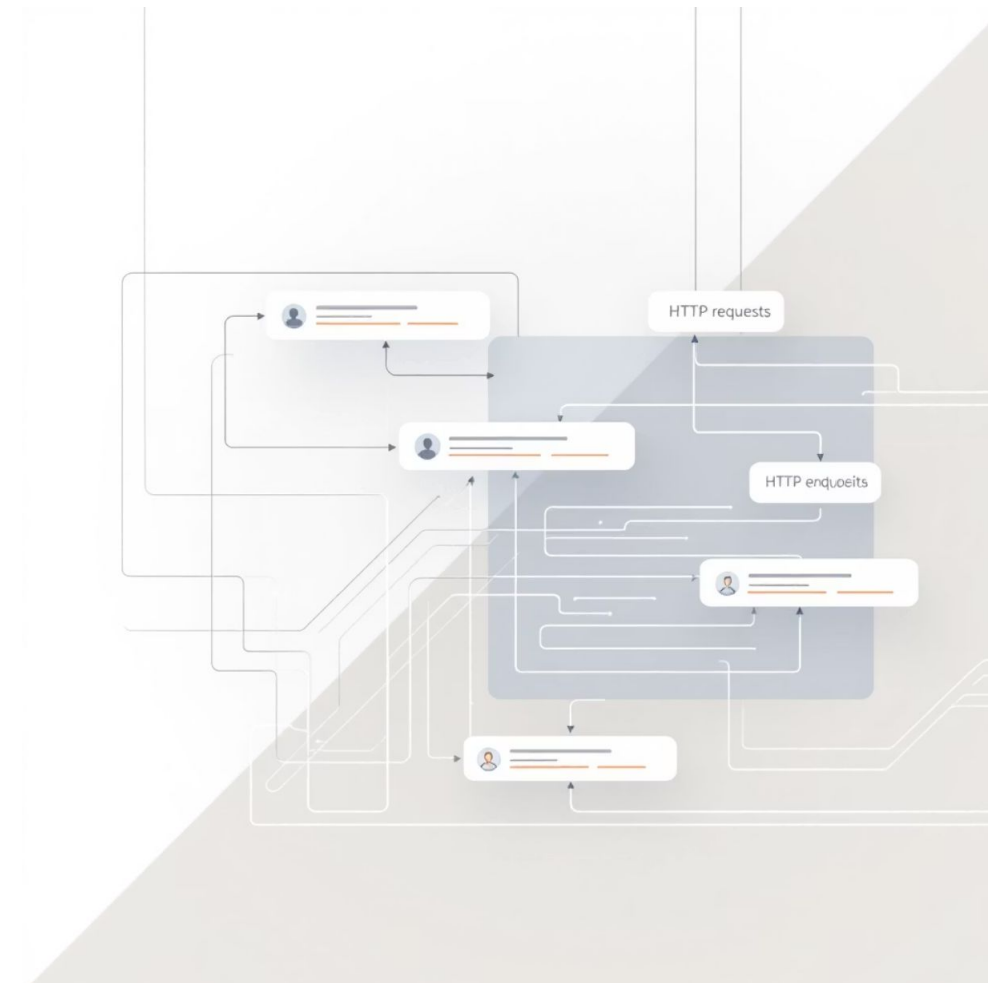
Пользовательские Интерфейсы

- Веб-интерфейс (HTML шаблоны)
- API интерфейс (JSON ответы)
- Административный интерфейс



Управление Потокком Действий

Маршрутизация через FastAPI роутеры (user.py, car_and_rent.py, payment.py, admin.py). Жизненный цикл запроса: HTTP Request → FastAPI App → Router → Controller Function → Database Operation → Response.



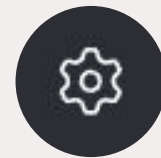


Безопасность и Технологии



Безопасность

Аутентификация JWT, авторизация по ролям, валидация Pydantic, хеширование паролей.



Выбор Технологий

FastAPI (производительность), SQLAlchemy (ORM), JWT (аутентификация), Openpyxl (отчеты), Pydantic (валидация), SQLite (БД).