

Министерство науки и высшего образования Российской Федерации Федеральное
государственное автономное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ОТЧЕТ

Сервис авторизации (AuthService)

Проект: «Платформа парсинга интернет-магазинов»

Дисциплина: «Проектирование информационных систем»

Преподаватель: Прутик А.Ф.

Выполнила Волкова Юлия,

ИУ6-75Б

1. Постановка задачи

Реализовать AuthService — сервис авторизации пользователей для приложения парсинга интернет-магазинов. Сервис обеспечивает регистрацию/вход/выход, проверку активной сессии и выдачу данных профиля.

Сессии — stateful: идентификатор сессии хранится в cookie, а данные сессии сохраняются в Redis с TTL. Данные пользователей хранятся в PostgreSQL.

Взаимодействие между компонентами сервиса — по gRPC.

2. Краткое описание решения

AuthService реализован двумя контейнерами: main_go (HTTP API /api/v1/*) и user_go (gRPC Users). main_go принимает запросы от Web UI, управляет cookie-сессиями в Redis и обращается к user_go по gRPC для операций с пользователями. user_go инкапсулирует работу с PostgreSQL и логику проверки пароля.

3. Архитектура (C4)

3.1.C1 — System Context (общая картина приложения)

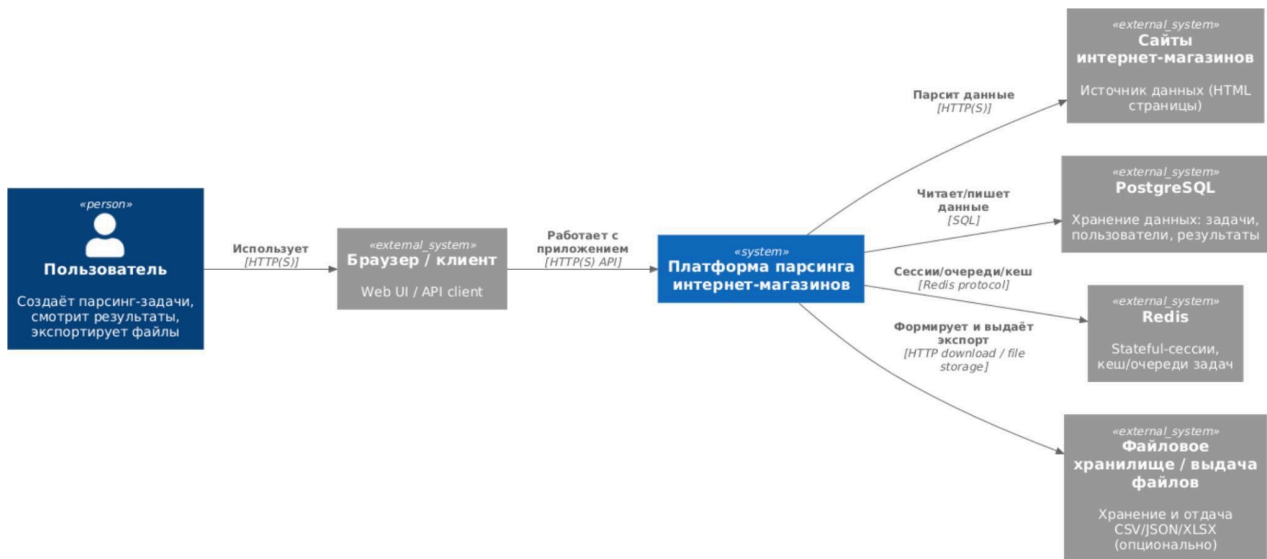


Рисунок 1 — C1: контекст платформы парсинга интернет-магазинов

3.2. C2 — Container diagram (общая картина приложения)

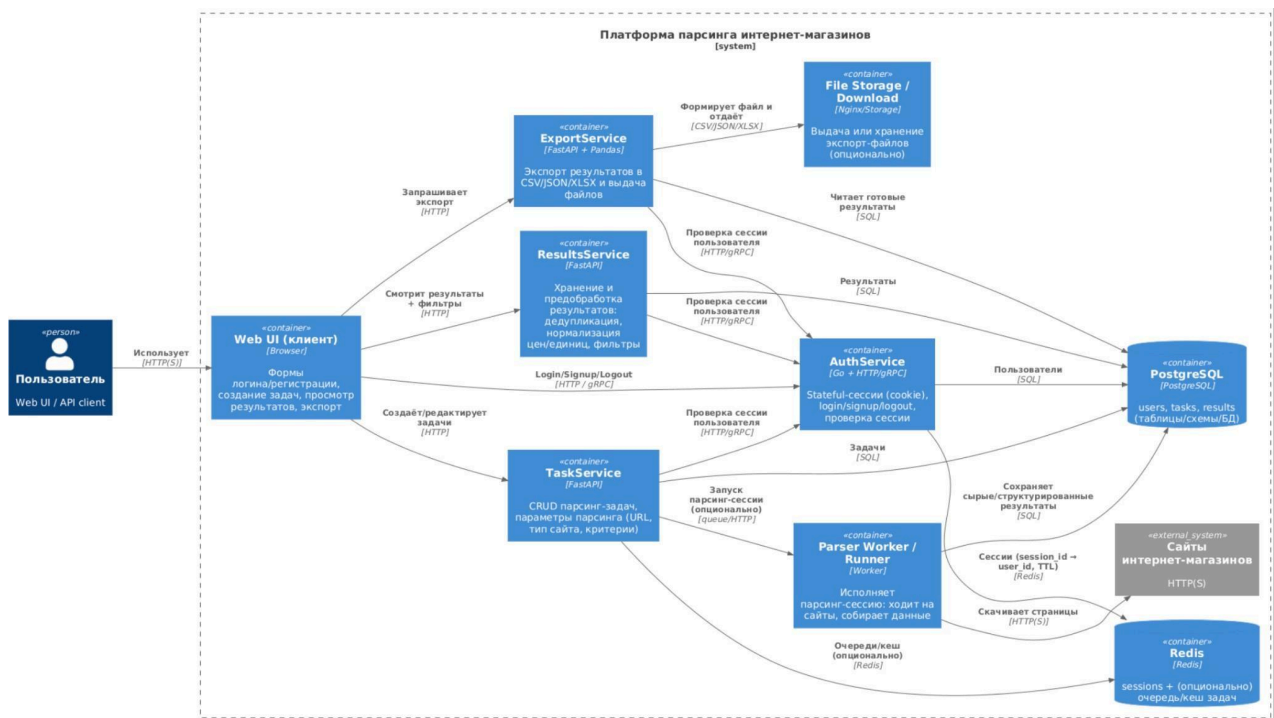


Рисунок 2 — C2: сервисы приложения; AuthService отвечает за сессии и данные пользователя

3.3. C3 — Component diagram (AuthService) Контейнер main_go (HTTP API)

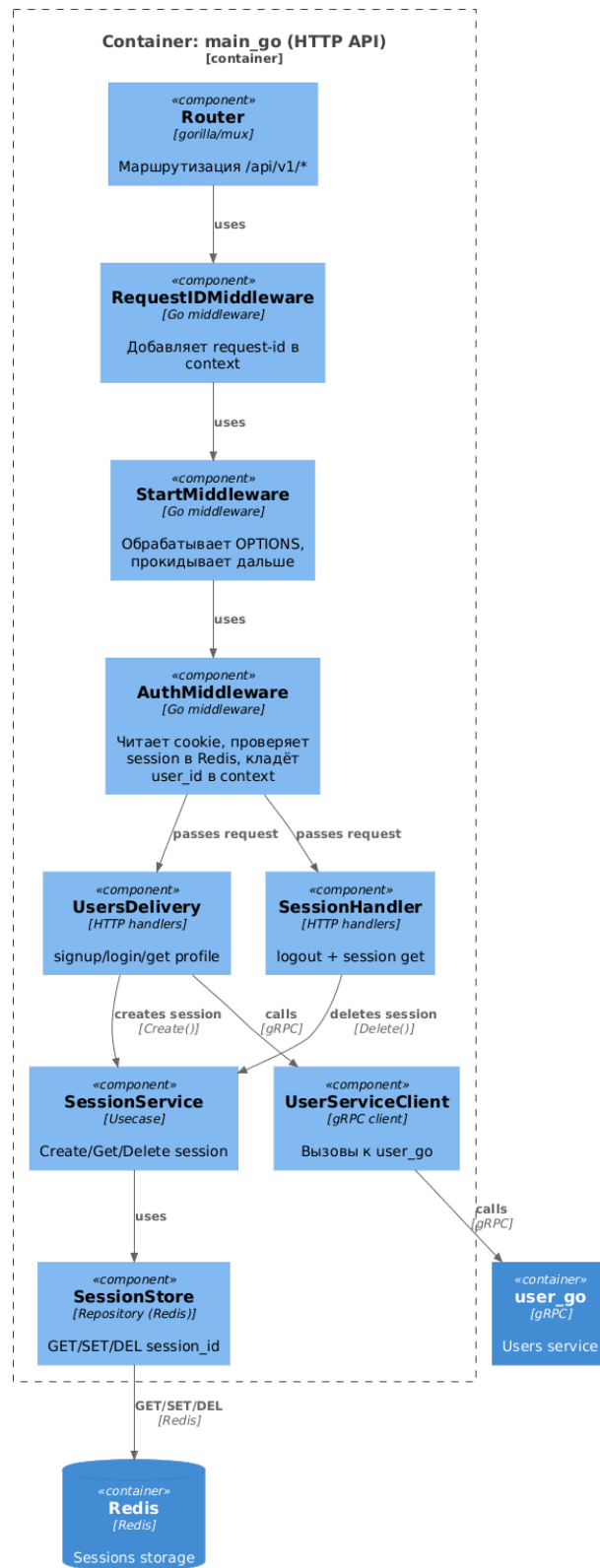


Рисунок 3 — C3(main_go): маршрутизация, middleware, handlers, Redis-сессии, gRPC-вызовы user_go

Контейнер user_go (gRPC Users)

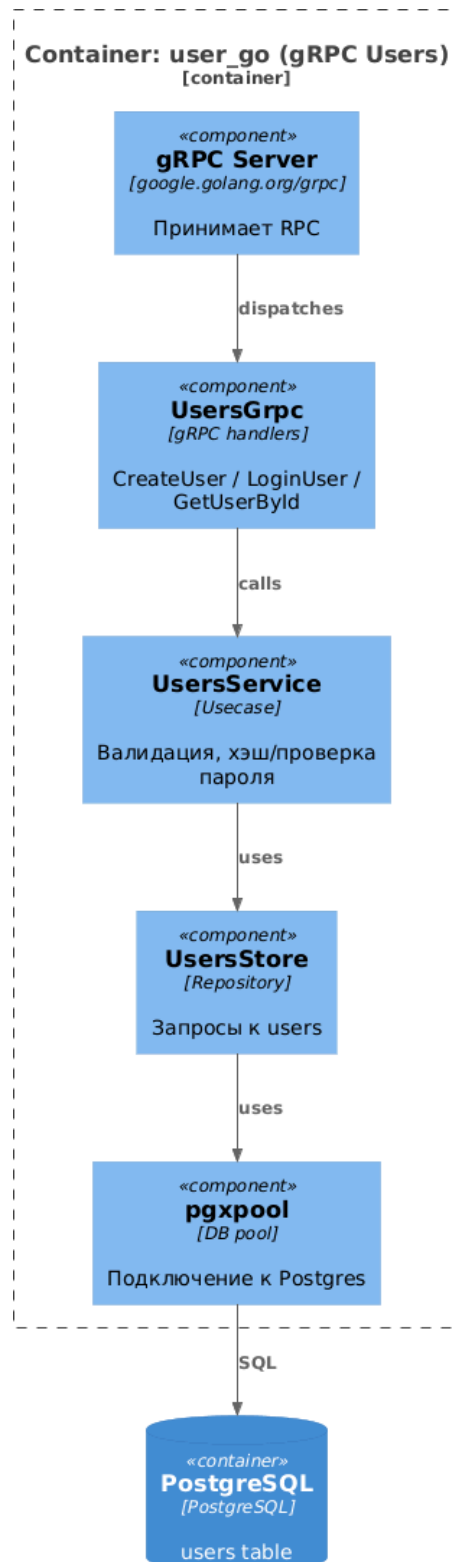


Рисунок 4 — C3(user_go): gRPC server, usecase-логика и доступ к PostgreSQL.

3.4. C4 — Code diagram (структура кода AuthService)

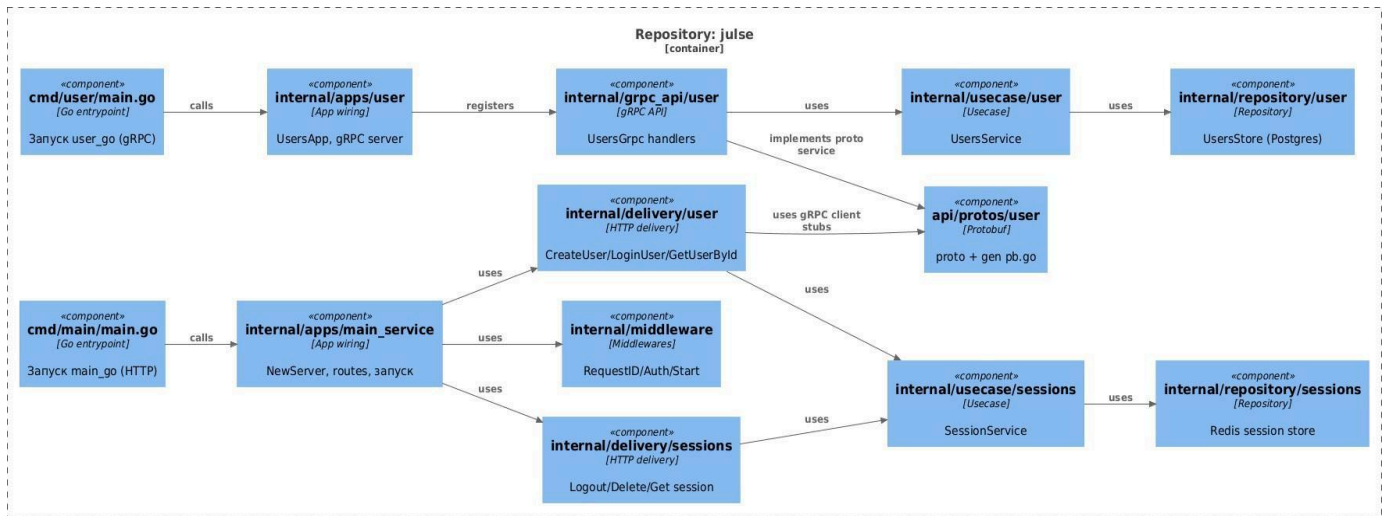


Рисунок 5 — C4: ключевые пакеты кода AuthService.

4. Модель данных и интерфейсы

Хранилище	Сущность/ключ	Содержимое
PostgreSQL	users	id, email (unique), username, city, password_hash, created_at
Redis	session	user_id + TTL (истечение)

AuthService использует PostgreSQL для пользователей и Redis для stateful-сессий.

API:

POST /api/v1/signup — регистрация и создание сессии; POST /api/v1/login — вход и создание сессии;

POST /api/v1/logout — удаление сессии;

5. Запуск (кратко)

AuthService поднимается в составе backend через Docker Compose:

docker compose up --build.

Сервис доступен на <http://localhost:3000>.