

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Бэк-энд разработка

**Отчет**

**Домашняя работа №1**

**Выполнил:**

**Русинов Василий**

**Группа К3440**

**Проверил:  
Добряков Д. И.**

**Санкт-Петербург**

**2025 г.**

## **Задача**

Необходимо спроектировать набор следующих диаграмм:

- общая архитектура решения (сервисы и их взаимосвязи, клиент-серверное взаимодействие);
- диаграмма компонентов;
- диаграммы БД по каждому сервису;
- диаграммы основных пользовательских сценариев (те сценарии, которые позволяют вашим приложением полноценно воспользоваться, пройти весь путь).

## **Ход работы**

Для выполнения данной лабораторной работы был выбран вариант 4 – сервис для аренды недвижимости.

- Вход
- Регистрация
- Личный кабинет пользователя (список арендованных и арендующихся объектов)
- Поиск недвижимости с фильтрацией по типу, цене, расположению
- Страница объекта недвижимости с фото, описанием и условиями аренды
- История сообщений и сделок пользователя

## Диаграмма общей архитектуры решения

Диаграмма отражает высокоуровневую архитектуру системы аренды недвижимости.

Показаны основные микросервисы, их зоны ответственности и взаимодействие с клиентским приложением через API Gateway.

Каждый сервис имеет собственную базу данных, что обеспечивает независимость, масштабируемость и изоляцию доменов.

Также отображено подключение сервиса хранения изображений для работы с фотографиями объектов недвижимости.

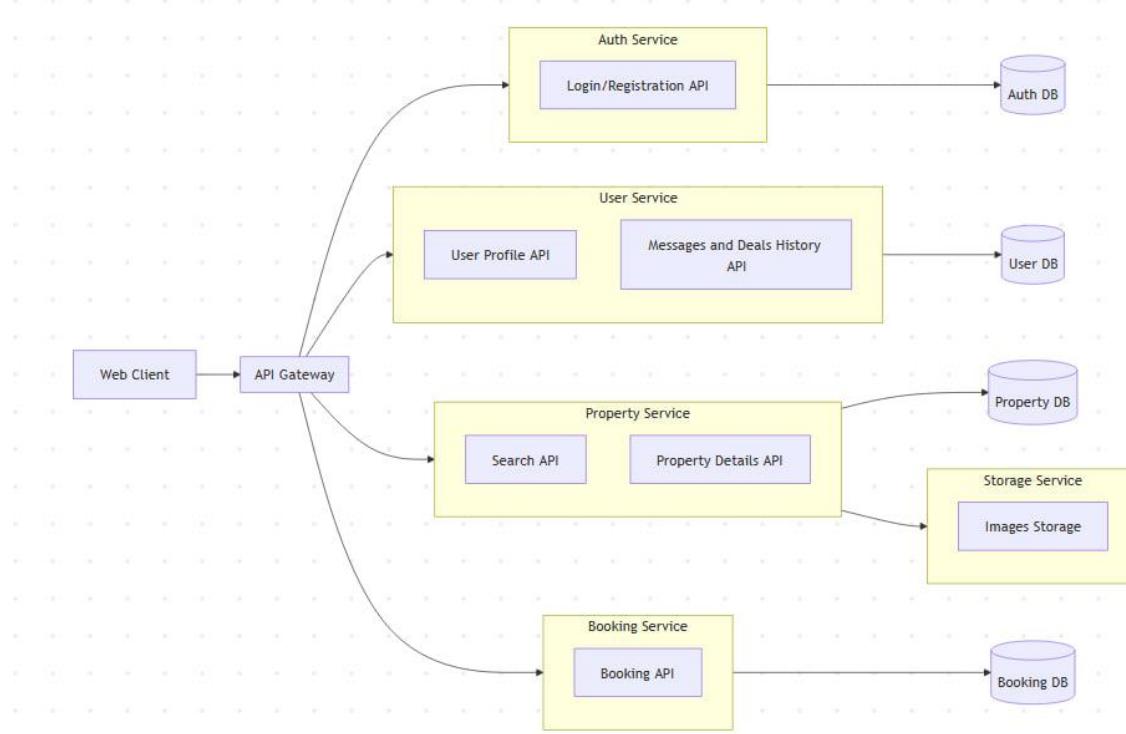


Рисунок 1 – Диаграмма общей архитектуры решения

## Диаграмма компонентов

Диаграмма иллюстрирует внутреннюю структуру приложения на уровне сервисов и модулей.

Frontend представлен набором интерфейсных компонентов. Backend разделён на несколько микросервисов (авторизация, пользователи, объекты недвижимости, бронирования), каждый из которых включает контроллеры и бизнес-логику.

Диаграмма демонстрирует модульную организацию и принципы разделения ответственности.

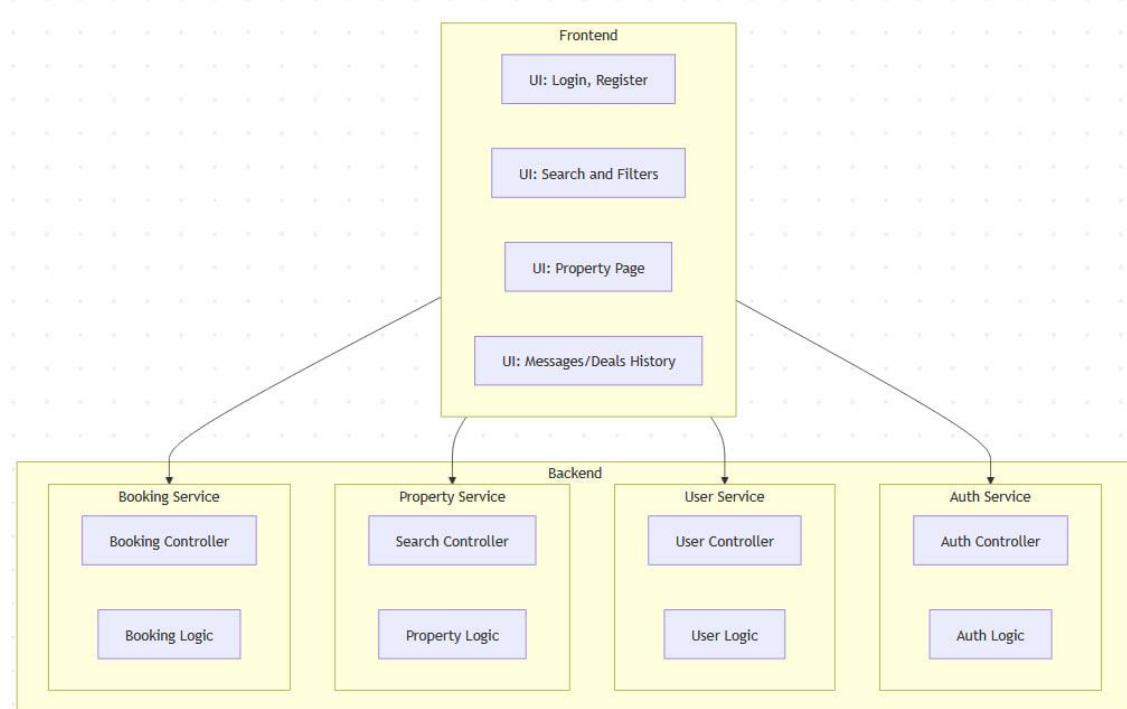


Рисунок 2 – Диаграмма компонентов

## Общая ER-диаграмма базы данных

Диаграмма показывает логические связи между сущностями системы: пользователями, профилями, сообщениями, сделками, объектами недвижимости, изображениями и бронированиями. Несмотря на микросервисную архитектуру, модель объединяет все данные в единую концептуальную схему, отражая связи «пользователь-объект», «объект-изображения», «пользователь-бронирование» и т.д. Диаграмма используется для понимания структуры данных и построения физических моделей отдельных сервисов.

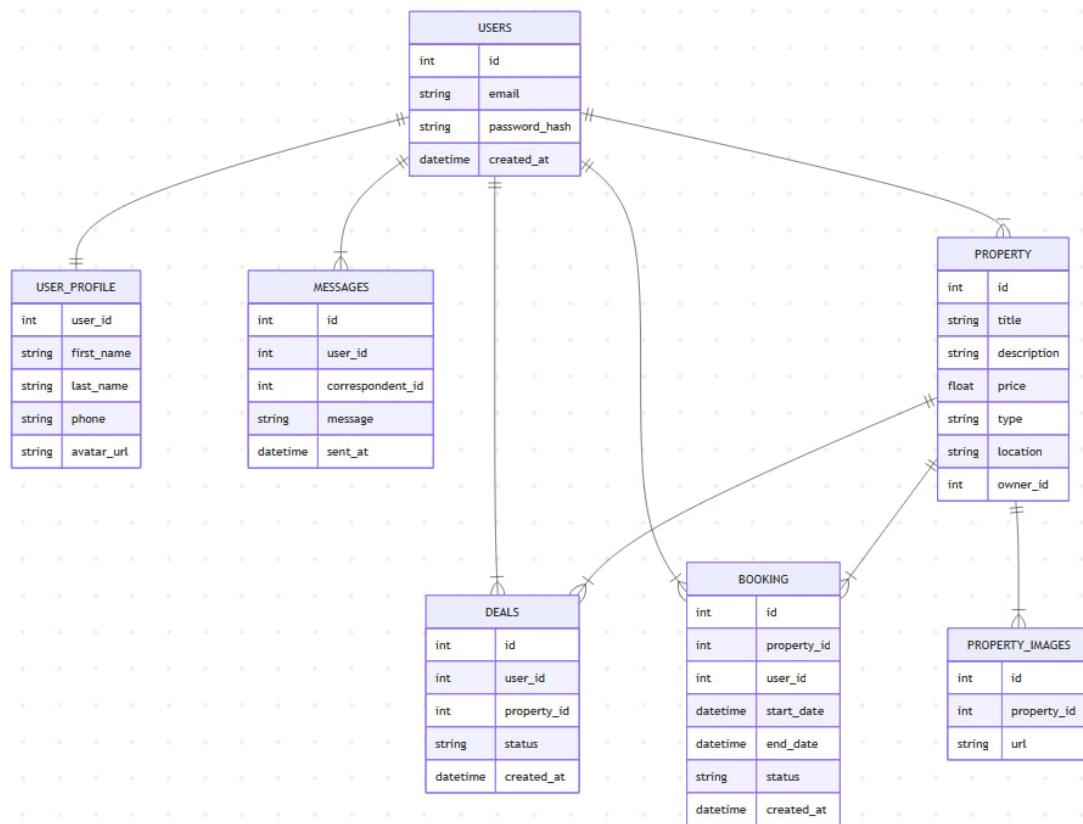


Рисунок 3 – Диаграмма БД

## Диаграммы БД по каждому сервису

Содержит только таблицу USERS.

Задача - хранить минимальную информацию для аутентификации: email, пароль.

USERS		
int	id	PK
string	email	
string	password_hash	
datetime	created_at	
datetime	updated_at	

Рисунок 4 – Диаграмма БД Auth Service

Содержит три сущности:

- USER\_PROFILE — персональные данные (ФИО, телефон, аватар)
- MESSAGES — переписка с арендодателями
- DEALS — история сделок (арендованные объекты, статусы)

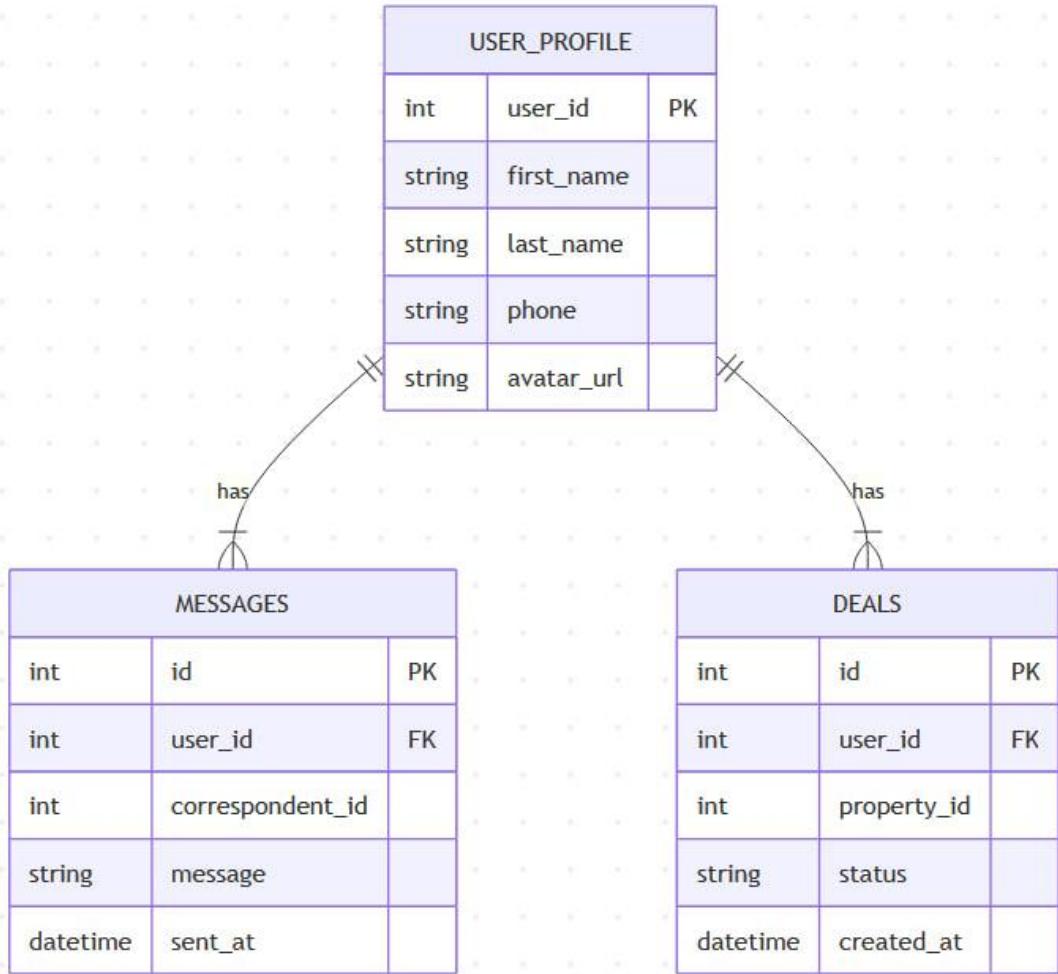


Рисунок 5 – Диаграмма БД User Service

Хранит информацию об объектах недвижимости.

- PROPERTY — описание квартиры/дома (тип, цена, расположение)
- PROPERTY\_IMAGES — галерея изображений

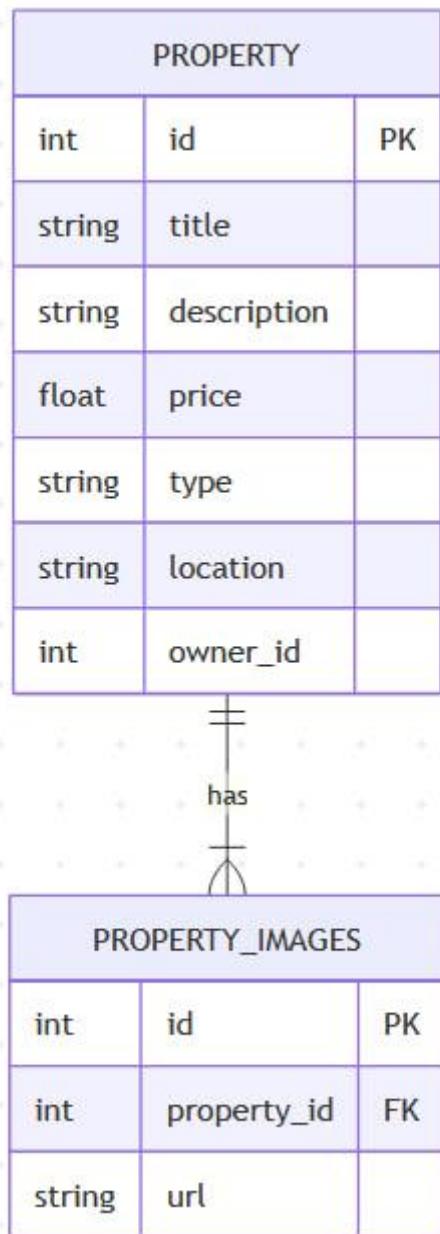


Рисунок 6 – Диаграмма БД Property Service

**BOOKING** — основная таблица с арендами, датами начала/конца, статусом бронирования.

BOOKING		
int	id	PK
int	property_id	FK
int	user_id	FK
datetime	start_date	
datetime	end_date	
string	status	
datetime	created_at	

Рисунок 7 – Диаграмма БД Booking Service

## Диаграммы пользовательских сценариев

Показывает процесс создания нового аккаунта: пользователь заполняет форму, фронтенд отправляет данные в сервис авторизации, происходит проверка и создание записи в БД, после чего возвращается токен.

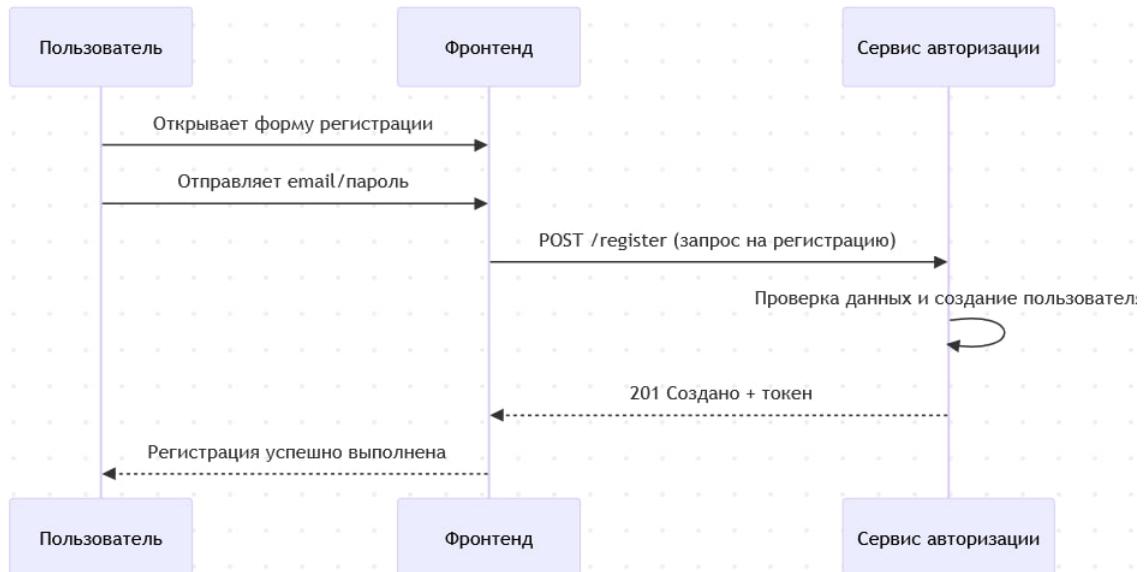


Рисунок 8 – Сценарий Регистрация

Диаграмма иллюстрирует процесс авторизации: пользователь вводит данные, сервис авторизации проверяет их и возвращает токен.

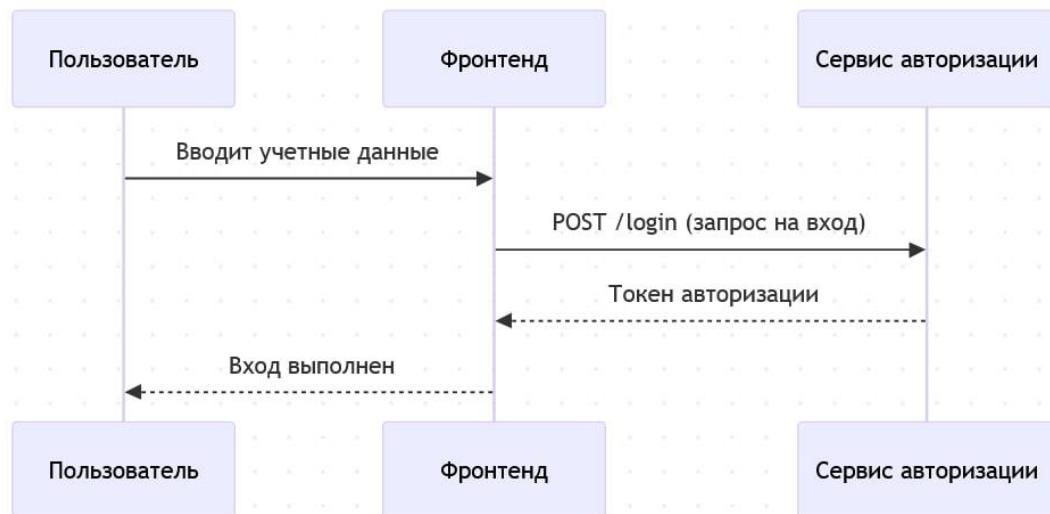


Рисунок 9 – Сценарий Вход

Отображает выполнение поиска: пользователь задаёт фильтры, фронтенд запрашивает данные у сервиса недвижимости, а затем выводит список подходящих объектов.

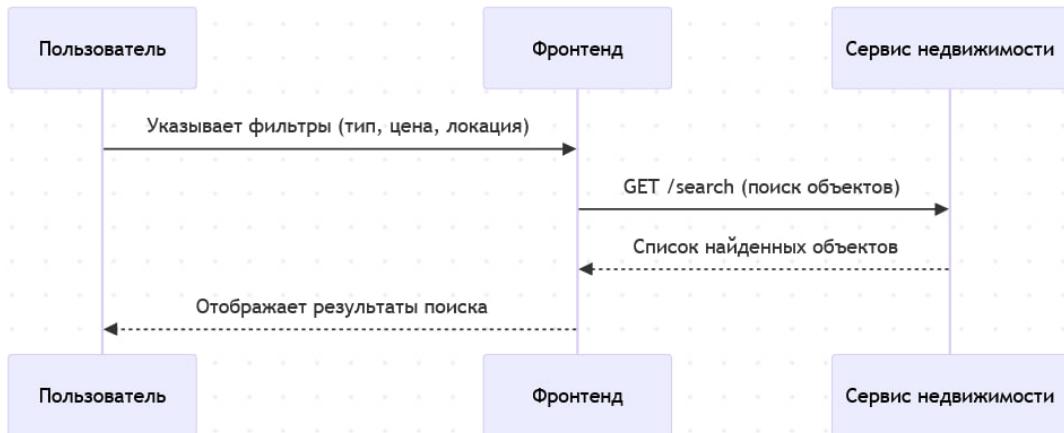


Рисунок 10 – Сценарий Поиск недвижимости

Показывает загрузку полной информации об объекте: фронтенд запрашивает данные у сервиса недвижимости, который дополнительно получает изображения из хранилища.

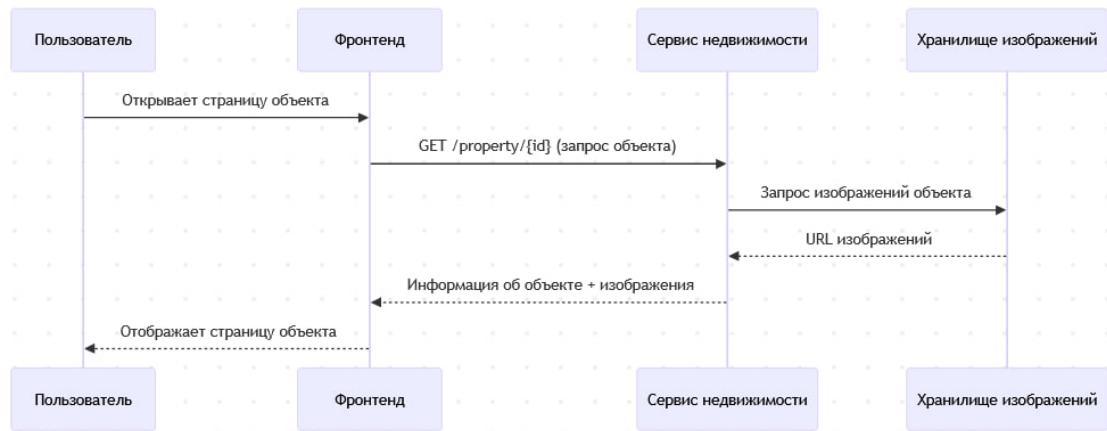


Рисунок 11 – Сценарий Просмотр объекта

Диаграмма показывает загрузку личной информации и списка бронирований: фронтенд обращается к сервису пользователей и сервису бронирований, после чего объединяет данные.

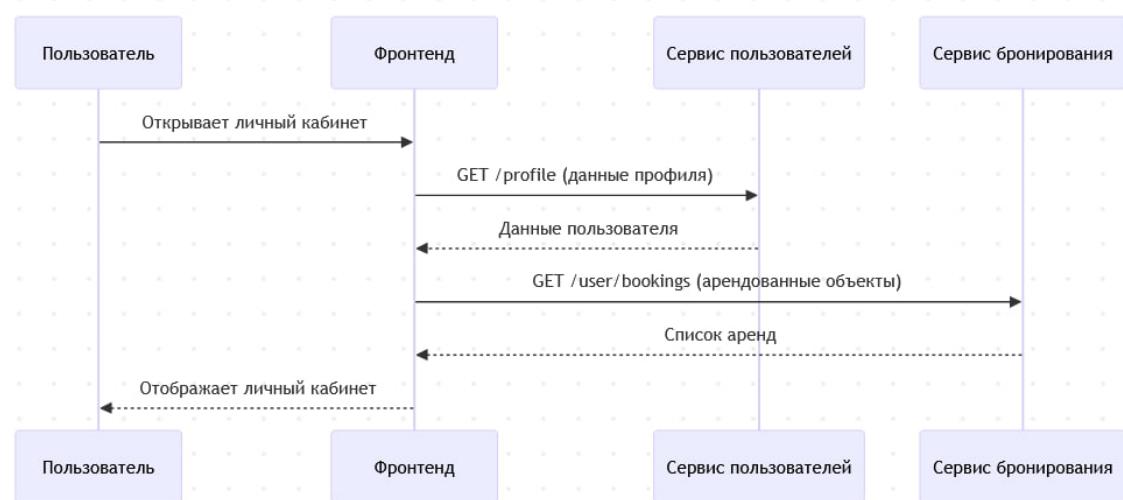


Рисунок 12 – Сценарий 5: Личный кабинет

Показывает получение списка сообщений и завершённых сделок из сервиса пользователей и отображение их пользователю.

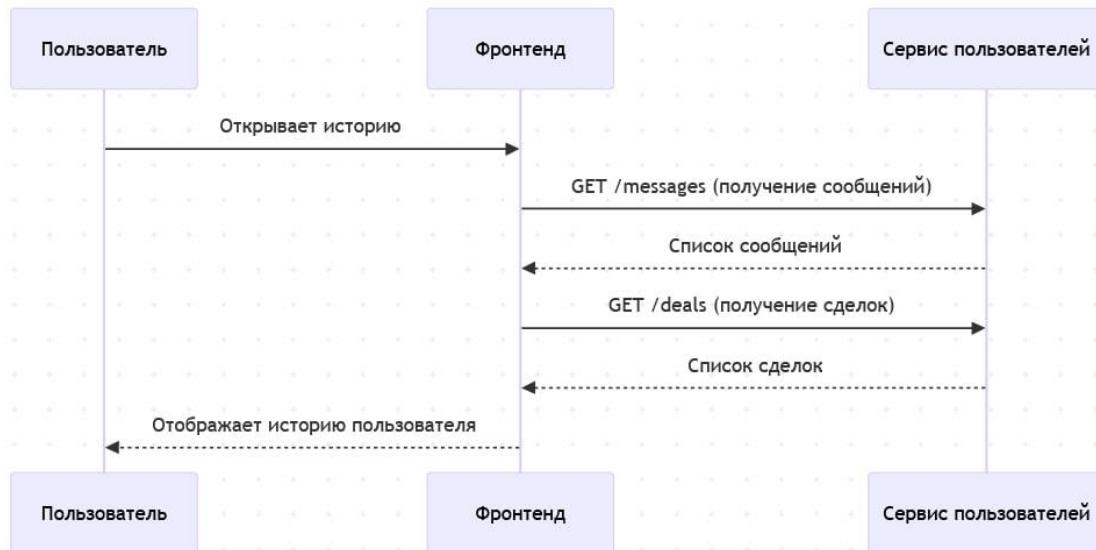


Рисунок 13 – Сценарий 6: История сообщений и сделок

## Вывод

В работе спроектирована архитектура сервиса аренды недвижимости: определены основные микросервисы, их взаимодействие и границы ответственности. Построены диаграммы архитектуры, компонентов,

пользовательских сценариев и ER-модели для каждого сервиса. Эти материалы формируют целостное понимание будущей системы и могут быть использованы как основа для её разработки и последующего масштабирования.