

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет

Домашняя работа №1

Выполнил:

Гуторова Инна

Группа К3441

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

1 Общая архитектура решения

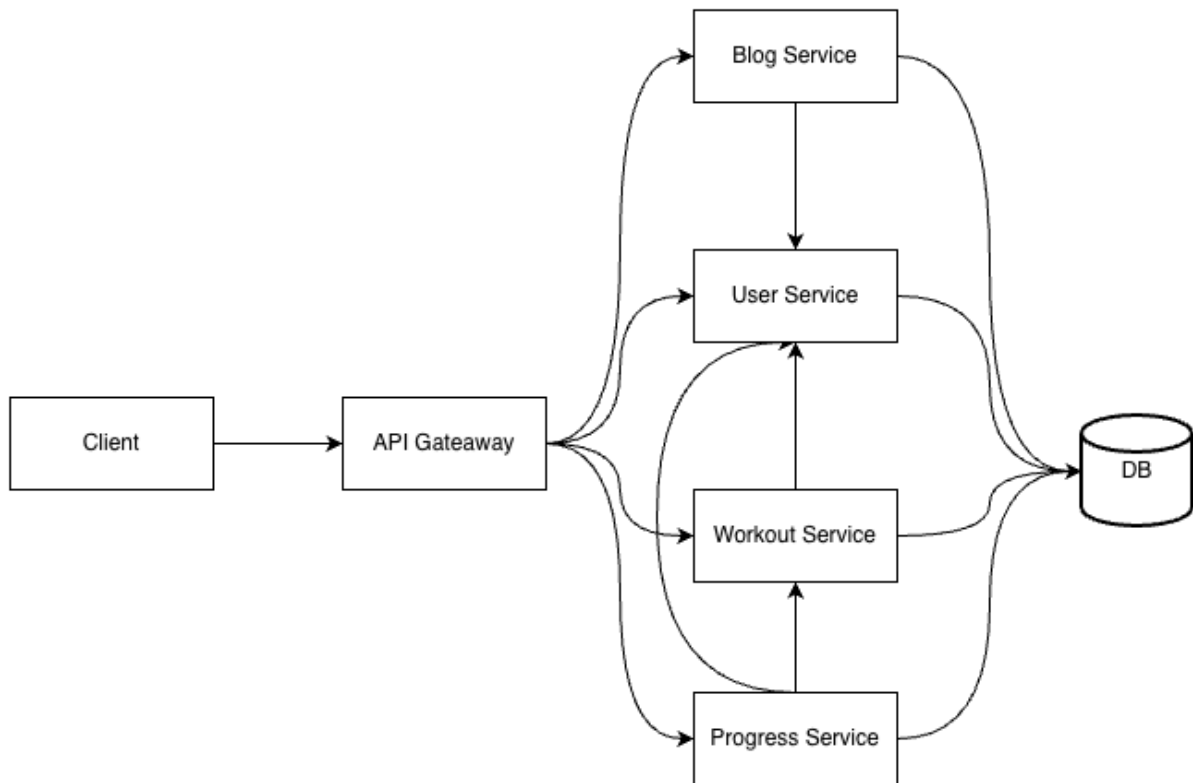
Приложение построено по микросервисной архитектуре. Основные сервисы:

Сервис	Основная задача
User Service	Управление пользователями (регистрация, авторизация, профиль)
Workout Service	Управление тренировками и упражнениями
Progress Service	Учет прогресса пользователей (история тренировок, упражнений, веса)
Blog Service	Ведение блога, публикация постов, лайки

Клиент-серверное взаимодействие:

Клиент общается с API Gateway, который маршрутизирует запросы к соответствующим сервисам. Сервисы взаимодействуют между собой напрямую через REST для получения необходимых данных.

Диаграмма общей архитектуры:



2. Диаграмма компонентов

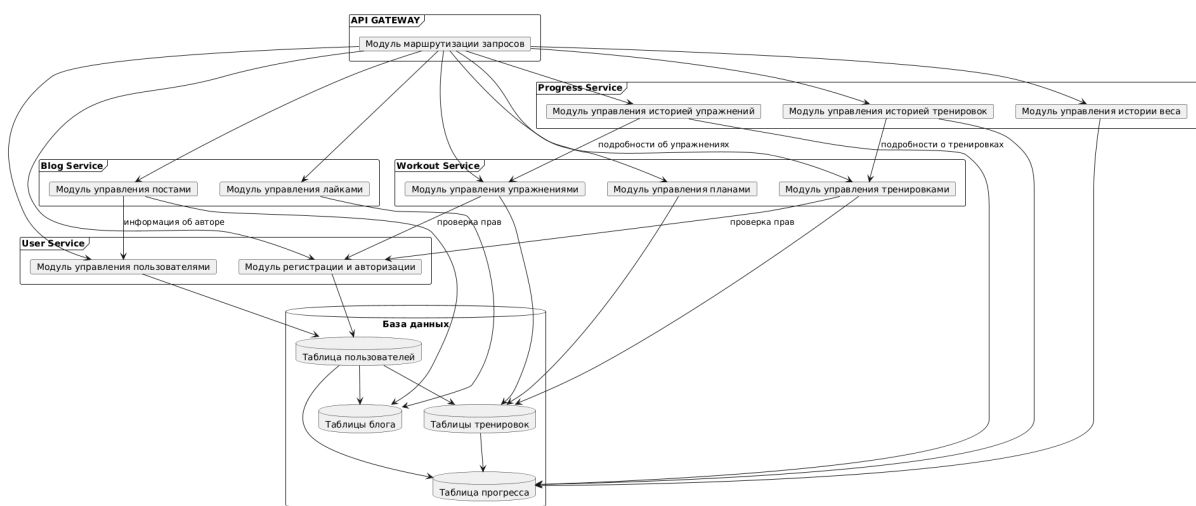


Диаграмма компонентов отражает структуру микросервисного приложения и показывает, как отдельные модули взаимодействуют между собой, с базой данных и клиентом через API Gateway.

1. API Gateway

- Модуль маршрутизации запросов принимает все входящие HTTP-запросы от клиента и перенаправляет их к соответствующим сервисам.
- Все микросервисы находятся за API Gateway, что обеспечивает единую точку входа и управление авторизацией, маршрутизацией и агрегированием данных.

2. User Service

Сервис управления пользователями состоит из двух основных модулей:

- Модуль регистрации и авторизации - отвечает за создание новых пользователей, проверку данных при входе и генерацию токенов доступа.
- Модуль управления пользователями - предоставляет функционал для просмотра и редактирования профиля пользователя, хранит информацию о росте, весе, дате рождения, роли и других данных.
- Оба модуля взаимодействуют с таблицей пользователей в базе данных.

3. Blog Service

Сервис ведения блога включает:

- Модуль управления постами - создание, редактирование, удаление и получение постов.
- Модуль управления лайками - учет и хранение лайков пользователей.
- Для работы оба модуля используют таблицы блога в базе данных.
- Модуль постов запрашивает у User Service информацию об авторе поста, обеспечивая корректное отображение данных на клиенте.

4. Workout Service

Сервис управления тренировками состоит из следующих модулей:

- Модуль управления тренировками - создание и редактирование тренировок.

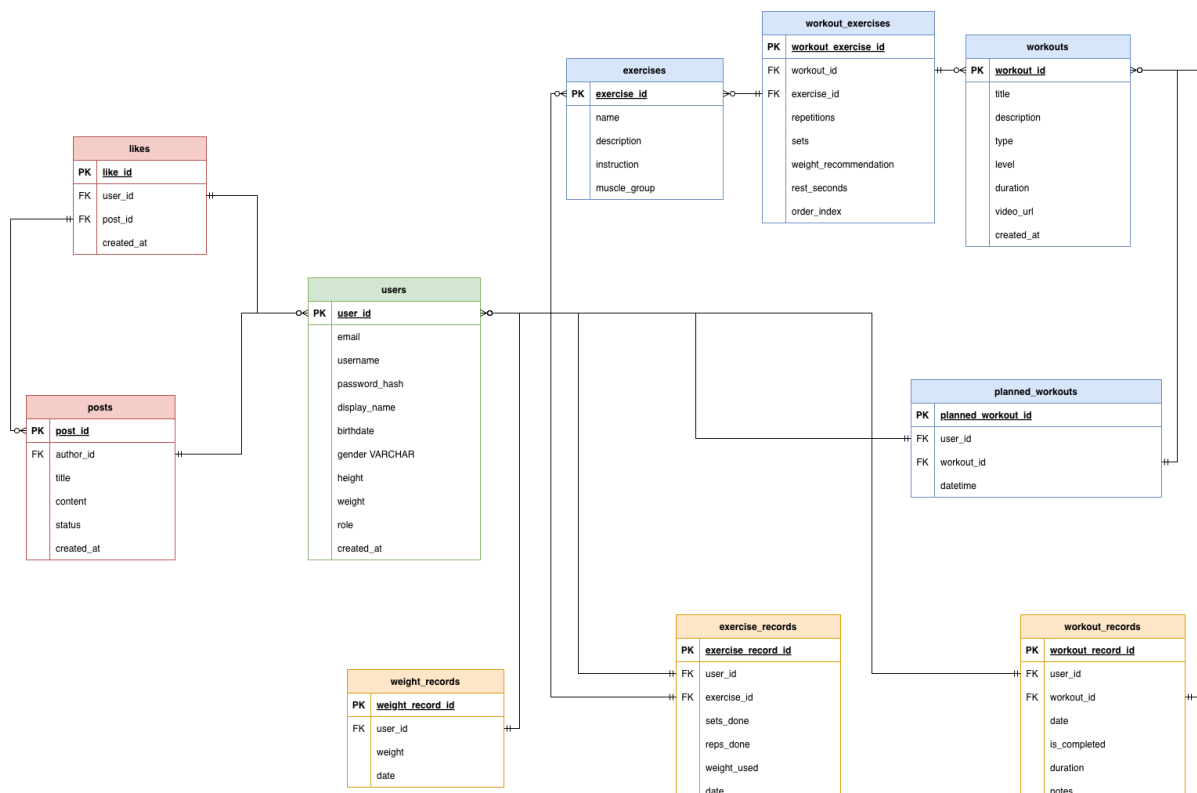
- Модуль управления упражнениями - добавление, редактирование и хранение упражнений.
- Модуль управления планами - планирование тренировок для пользователей.
- Все модули взаимодействуют с таблицами тренировок в базе данных и проверяют права пользователя через User Service.

5. Progress Service

Сервис учета прогресса включает четыре модуля:

- Модуль управления историей тренировок - хранение и получение данных о завершенных тренировках пользователя.
- Модуль управления историей упражнений - хранение и запись результатов выполненных упражнений.
- Модуль управления историей веса - учет изменения веса пользователя.
- Для получения деталей тренировок и упражнений модули истории запрашивают соответствующую информацию у Workout Service.

3. Диаграмма БД

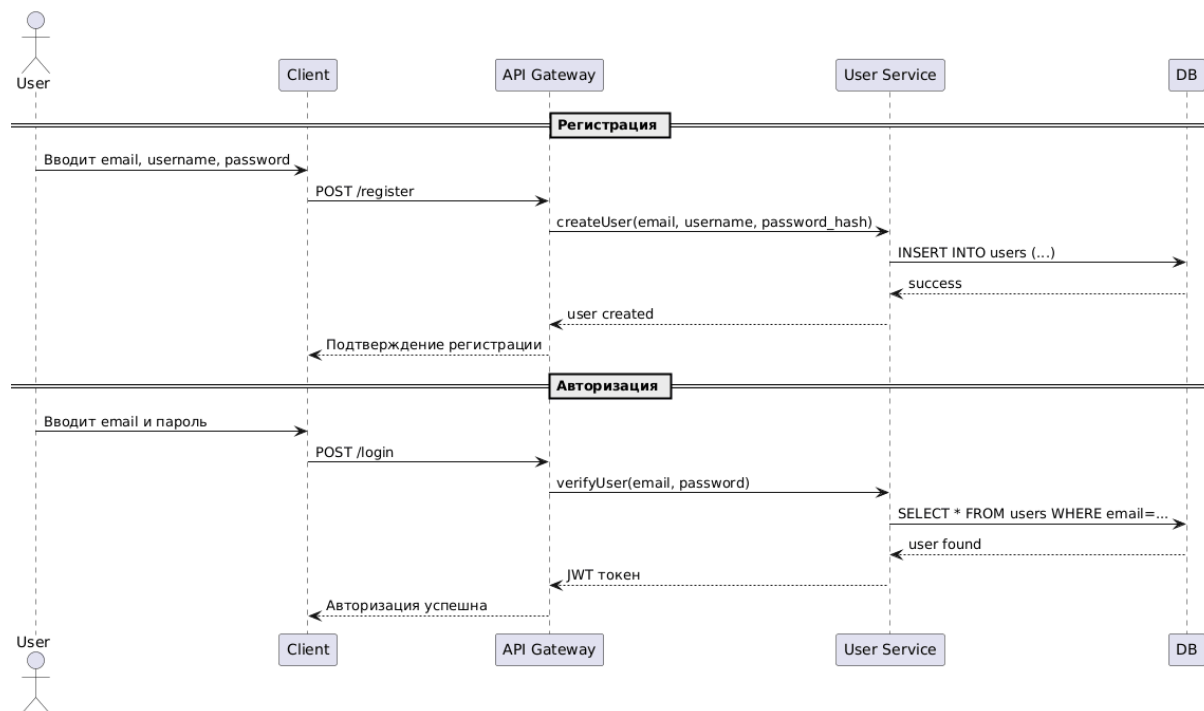


В диаграмме отмечено цветами:

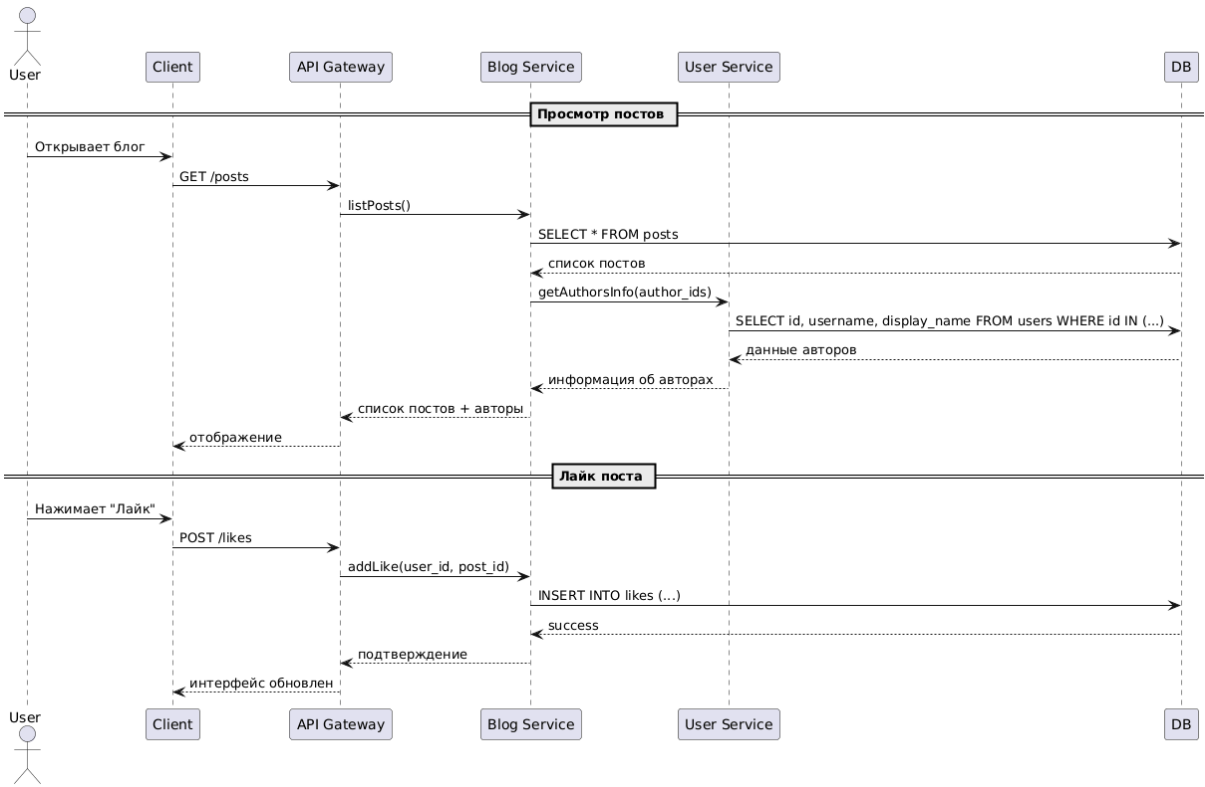
- Таблицы, связанные с управлением пользователями - зеленый
- Таблицы, связанные с ведением блога - красный
- Таблица, связанная с управлением тренировками - синий
- Таблицы, связанные с ведением прогресса - оранжевый

4. Диаграммы основных пользовательских сценариев

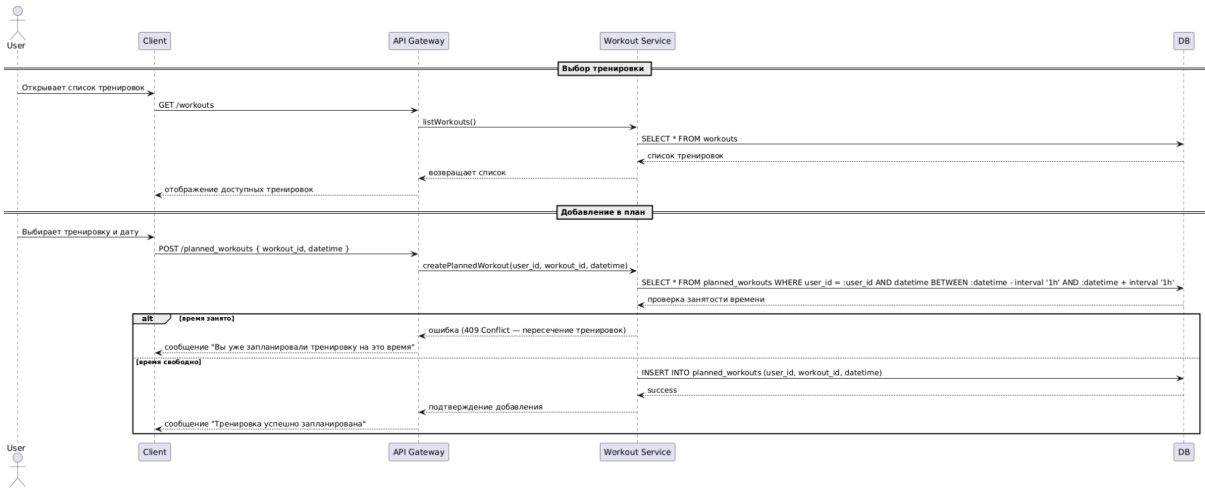
4.1 Регистрация и авторизация



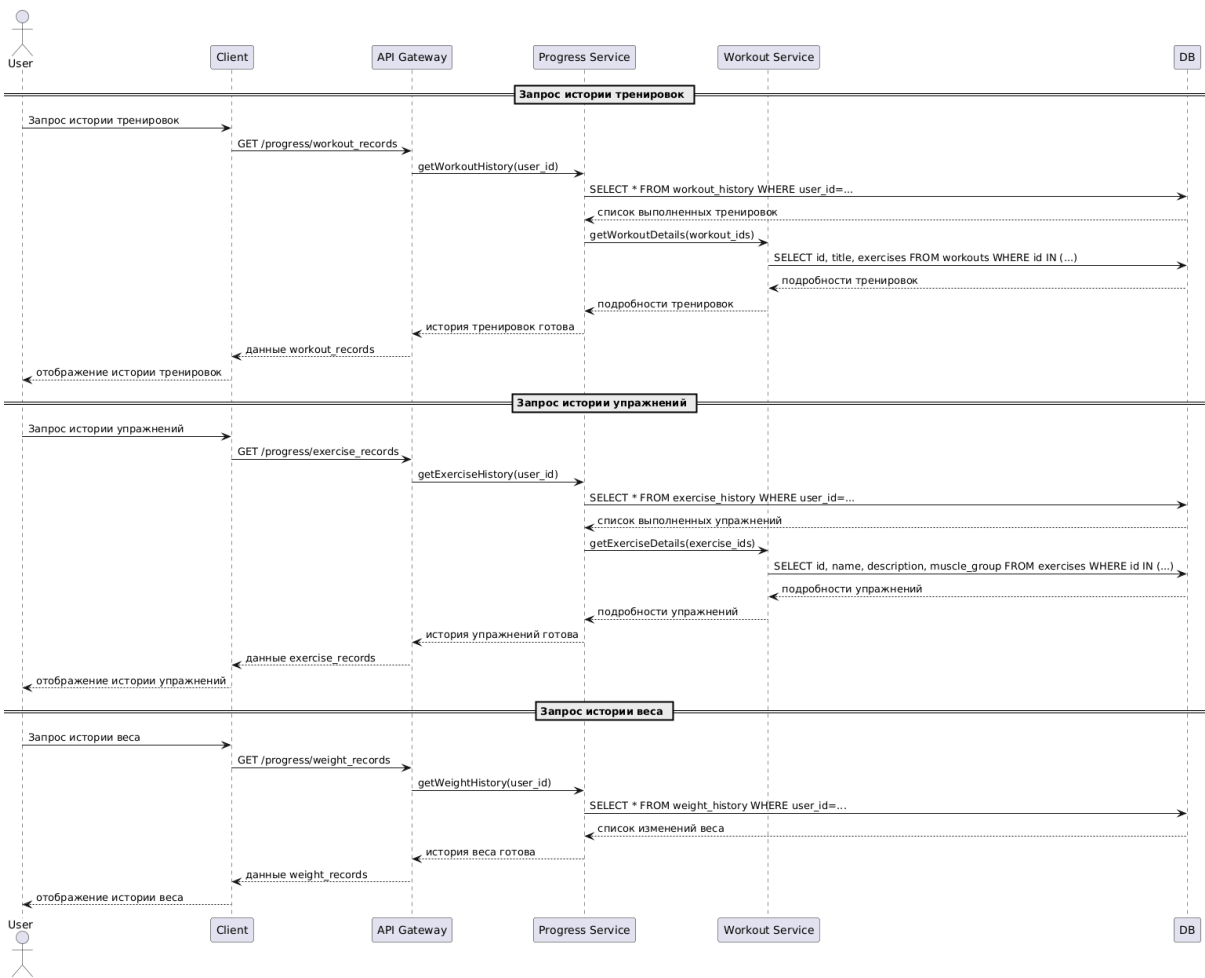
4.2 Взаимодействие с блогом



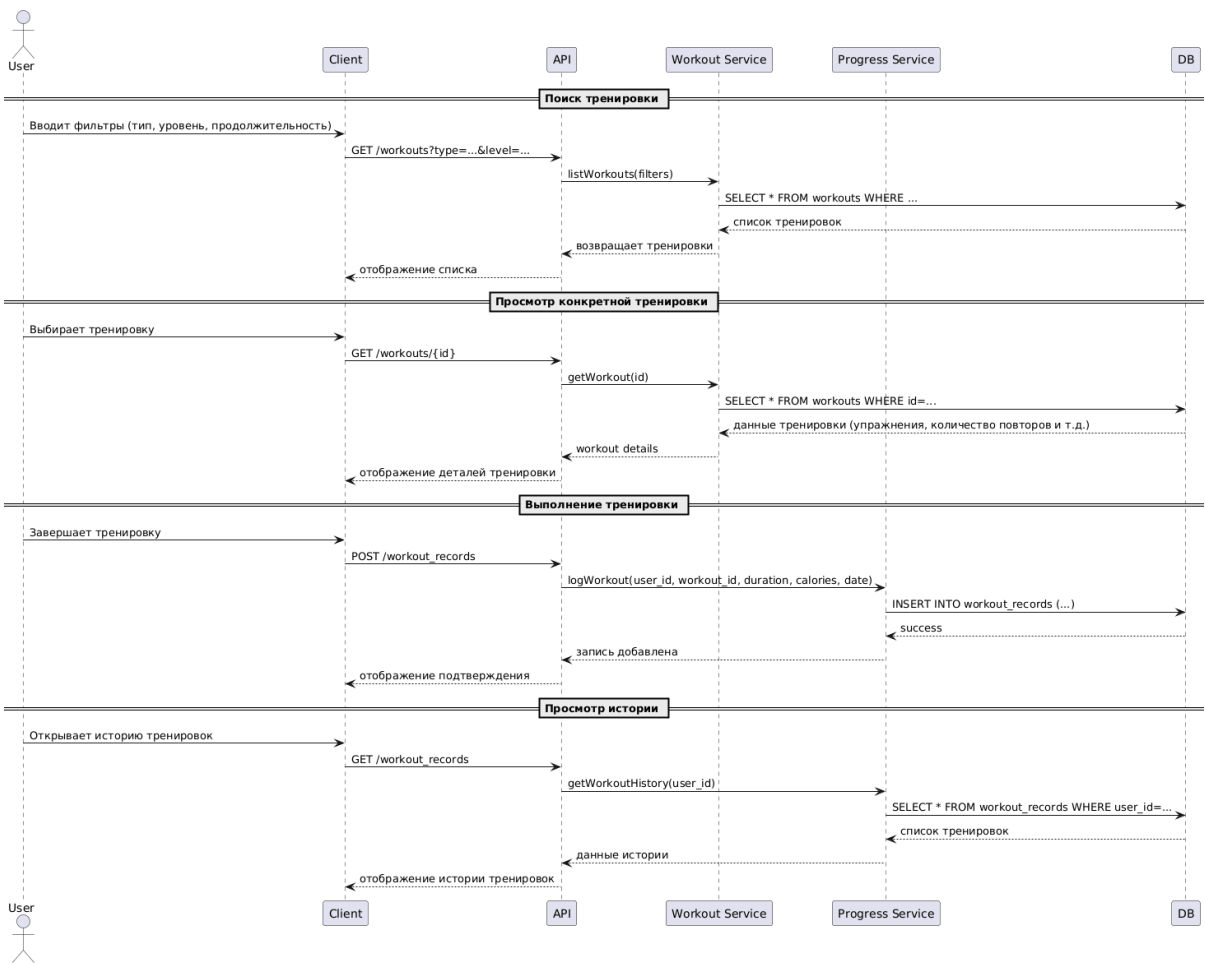
4.3 Планирование тренировки



4.4 Просмотр прогресса



4.5 Выполнение тренировки



Вывод

В ходе работы был выполнен технический дизайн микросервисного приложения, включающий:

- Архитектуру решения с клиентом, API Gateway и четырьмя основными сервисами.
- Диаграмму компонентов, показывающую внутренние модули каждого сервиса.
- Диаграммы баз данных, отражающие структуру таблиц и связи между ними.
- Подробные диаграммы пользовательских сценариев, учитывающие межсервисное взаимодействие и отдельные запросы пользователя для разных видов данных.

Результатом является полное представление о микросервисной архитектуре приложения, о последовательности действий пользователей, что позволяет приступить к реализации с ясным пониманием структуры и взаимосвязей.