

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Бэк-энд разработка

Отчет

Лабораторная работа №6

Выполнил:

Зотов Михаил

К3341

Проверил:

Добряков Д. И.

Санкт-Петербург

2025 г.

## Задача

- подключить и настроить rabbitMQ/kafka;
- реализовать межсервисное взаимодействие посредством rabbitMQ/kafka.

## Ход работы

1. В `docker-compose.yml` добавлен сервис `rabbitmq` с панелью управления.
2. В `auth-service` реализован обработчик сообщений: сервис подписывается на очередь `auth_queue`, принимает запросы типа `getUser`, ищет пользователя в БД и возвращает данные без пароля через механизм **RPC** (`correlationId` + `replyTo`).
3. В `subscription-service` реализован клиент RabbitMQ: создаётся временная очередь для ответа, отправляется сообщение в `auth_queue`, ожидается ответ по `correlationId`.
4. Методы получения подписчиков и подписок (`getFollowers`, `getFollowing`) используют RPC-запросы к `auth-service`, чтобы получить список пользователей.
5. Проверка: при запросе к `subscription-service` возвращается список пользователей, полученный через RabbitMQ.

## Результаты

- RabbitMQ успешно подключён и работает в отдельном контейнере.
- Настроено межсервисное взаимодействие по схеме RPC.
- `subscription-service` получает данные о пользователях через обмен сообщениями с `auth-service`.

- Сервисы функционируют независимо, взаимодействуя через очередь, что подтверждает корректность асинхронного взаимодействия.

## **Вывод**

Задача выполнена: RabbitMQ настроен, реализован обмен данными между сервисами с использованием очередей и RPC-механизма. Межсервисное взаимодействие работает стабильно и соответствует требованиям задания.