

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет

по первому этапу разработки новой программной системы по
дисциплине «Рефакторинг баз данных и приложений»

Этап 1

Сервис статистики

Автор: Иванов Андрей Вячеславович

Факультет: ПИиКТ

Группа: Р34101

Преподаватель: Логинов Иван Павлович



Санкт-Петербург, 2024

Техническое задание первого этапа

На этом этапе требуется реализовать сервис статистики. Спланировать архитектуру и контракты сервисов. Настроить систему контроля версий, разделить разработку на ветки. Настроить базу данных и миграции.

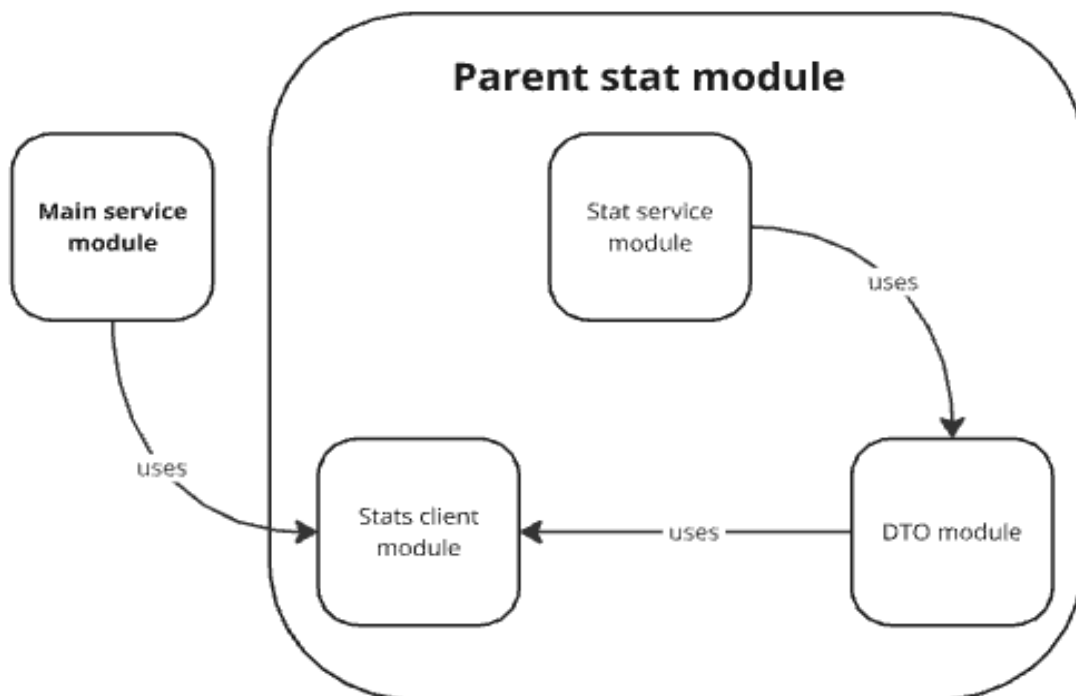
- Реализовать сервис статистики в соответствии со спецификацией [stats-service-spec.json](#);
- Реализовать HTTP-клиент для работы с сервисом статистики;
- Подготовить сборку проекта;
- Написать Postman-тесты для проверки работы функциональности;

Выполнение

Приложение состоит из двух отдельно запускаемых сервисов — в контексте сборки проекта при помощи Maven это означает, что проект многомодульный. Сами сервисы разбиты на подмодули.

Сервис статистики состоит из HTTP-сервиса и HTTP-клиента. Это значит, что модуль статистики разделён на два подмодуля. Механизм взаимодействия сервиса и клиента предполагает, что они используют одни и те же объекты для запросов и ответов. Исходя из этого, был выделен еще один подмодуль, в котором были размещены общие классы DTO.

Структуру модулей и зависимостей можно представить в виде такой схемы:



Модули основного сервиса и сервиса статистики содержат dockerfile. В корне проекта был создан файл docker-compose.yml, описывающий запуск контейнеров с сервисами проекта и базами данных для них. Файл pom.xml, описывающий сборку основного сервиса, на данном этапе содержит только указание на родительский модуль и идентификатор артефакта.

Для того, чтобы автоматически вносить изменения в базу данных, отслеживать историю этих изменений и упростить процесс синхронизации базы данных между различными средами используются миграции. В данном проекте используется Liquibase – инструмент для управления миграциями базы данных.

Для проверки работоспособности функционала были написаны [Postman-тесты](#).

Разработка данного функционала велась в ветке с именем stat_svc.

Вся разработка велась в отдельных фича-ветках, а по окончании вливалась в ветку main.

Для подробного ознакомления с кодом проекта можно перейти в [PR данного этапа](#).

Вывод

На первом этапе проекта успешно реализована модульная архитектура с использованием Liquibase для управления миграциями базы данных, настроено развертывание через Docker и проверена функциональность с помощью Postman-тестов. Работа велась в отдельных фича-ветках, что обеспечило чистоту основной ветки и удобство разработки. При разработке соблюдались общие принципы грамотной разработки ПО, такие как SOLID и DRY, обеспечивающие высокую читаемость и поддержку кода.