Федеральное государственное автономное образовательное учреждение высшего образования

«**Санкт-Петербургский национальный исследовательский**

**Университет ИТМО»**

**Факультет программной инженерии и компьютерной техники**

**Программирование**

**Лабораторная работа №7**

**Вариант №** 23344

Выполнил: Сарваров Тимур Фазаелович

Группа: Р3113

Проверил: Ермаков Михаил Константинович

Санкт-Петербург, 2025г.

**Задание:**

Доработать программу из [лабораторной работы №6](https://se.ifmo.ru/courses/programming#lab6) следующим образом:

1. Организовать хранение коллекции в реляционной СУБД (PostgresQL). Убрать хранение коллекции в файле.
2. Для генерации поля id использовать средства базы данных (sequence).
3. Обновлять состояние коллекции в памяти только при успешном добавлении объекта в БД
4. Все команды получения данных должны работать с коллекцией в памяти, а не в БД
5. Организовать возможность регистрации и авторизации пользователей. У пользователя есть возможность указать пароль.
6. Пароли при хранении хэшировать алгоритмом MD2
7. Запретить выполнение команд не авторизованным пользователям.
8. При хранении объектов сохранять информацию о пользователе, который создал этот объект.
9. Пользователи должны иметь возможность просмотра всех объектов коллекции, но модифицировать могут только принадлежащие им.
10. Для идентификации пользователя отправлять логин и пароль с каждым запросом.

Необходимо реализовать многопоточную обработку запросов.

1. Для многопоточного чтения запросов использовать Cached thread pool
2. Для многопотчной обработки полученного запроса использовать Cached thread pool
3. Для многопоточной отправки ответа использовать ForkJoinPool
4. Для синхронизации доступа к коллекции использовать синхронизацию чтения и записи с помощью java.util.concurrent.locks.ReadWriteLock

**Порядок выполнения работы:**

1. В качестве базы данных использовать PostgreSQL.
2. Для подключения к БД на кафедральном сервере использовать хост pg, имя базы данных - studs, имя пользователя/пароль совпадают с таковыми для подключения к серверу.

**UML-диаграмма:**

Изображение выглядит как текст, снимок экрана, диаграмма, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

**Код программы:**

<https://github.com/Tim4ik77/prog7>

**Вывод:**

В ходе выполнения лабораторной работы приложение разделено на два приложения – клиентское и серверное. Реализована отправка и получение команд клиентом и сервером с помощью протокола UDP, сериализация и десериализация классов коллекции. Были изменены команды в соответствии с заданием. Также было реализовано логирование ключевых событий в серверной части (запуск, приём запроса, отправка ответа и т.д.) с использованием библиотеки Logback, что облегчает отладку и сопровождение проекта. Все операции обработки коллекции на сервере были реализованы с использованием Stream API и лямбда-выражений.