Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 «Программная инженерия» –

Системное и прикладное программное обеспечение

**Отчёт**

**По лабораторной работе №7**

**По программированию**

**Вариант: 3511**

Выполнили:

студенты 1 курса

Батманов Даниил Евгеньевич

и Соколов Андрей Дмитриевич

Группа: Р3107

Принял:

Гаврилов Антон Валерьевич

Отчёт принят «\_\_»\_\_\_\_\_2023 г.

Оценка: \_\_\_\_\_\_\_\_\_\_\_

г. Санкт-Петербург, 2023

**Оглавление**

[Задание 3](#_Toc138079560)

[Исходный код программы и uml-диаграмма 6](#_Toc138079561)

[Вывод 8](#_Toc138079562)

[Список литературы 9](#_Toc138079563)

# 

# 

# 

# Задание

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

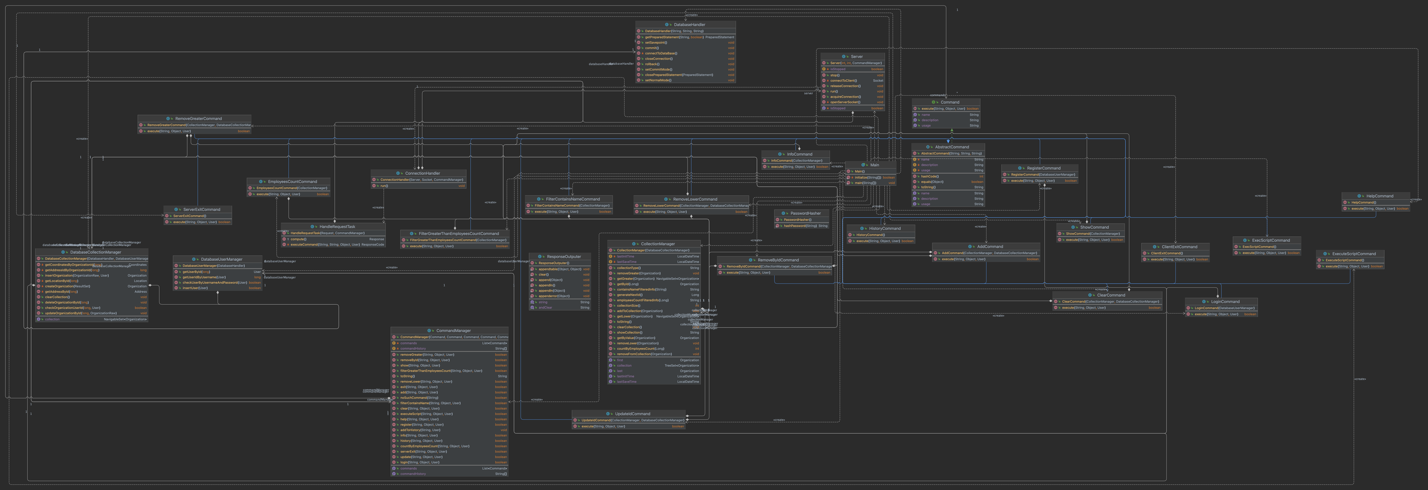
1. Организовать хранение коллекции в реляционной СУБД (PostgresQL). Убрать хранение коллекции в файле.
2. Для генерации поля id использовать средства базы данных (sequence).
3. Обновлять состояние коллекции в памяти только при успешном добавлении объекта в БД
4. Все команды получения данных должны работать 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 SHA-224
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.

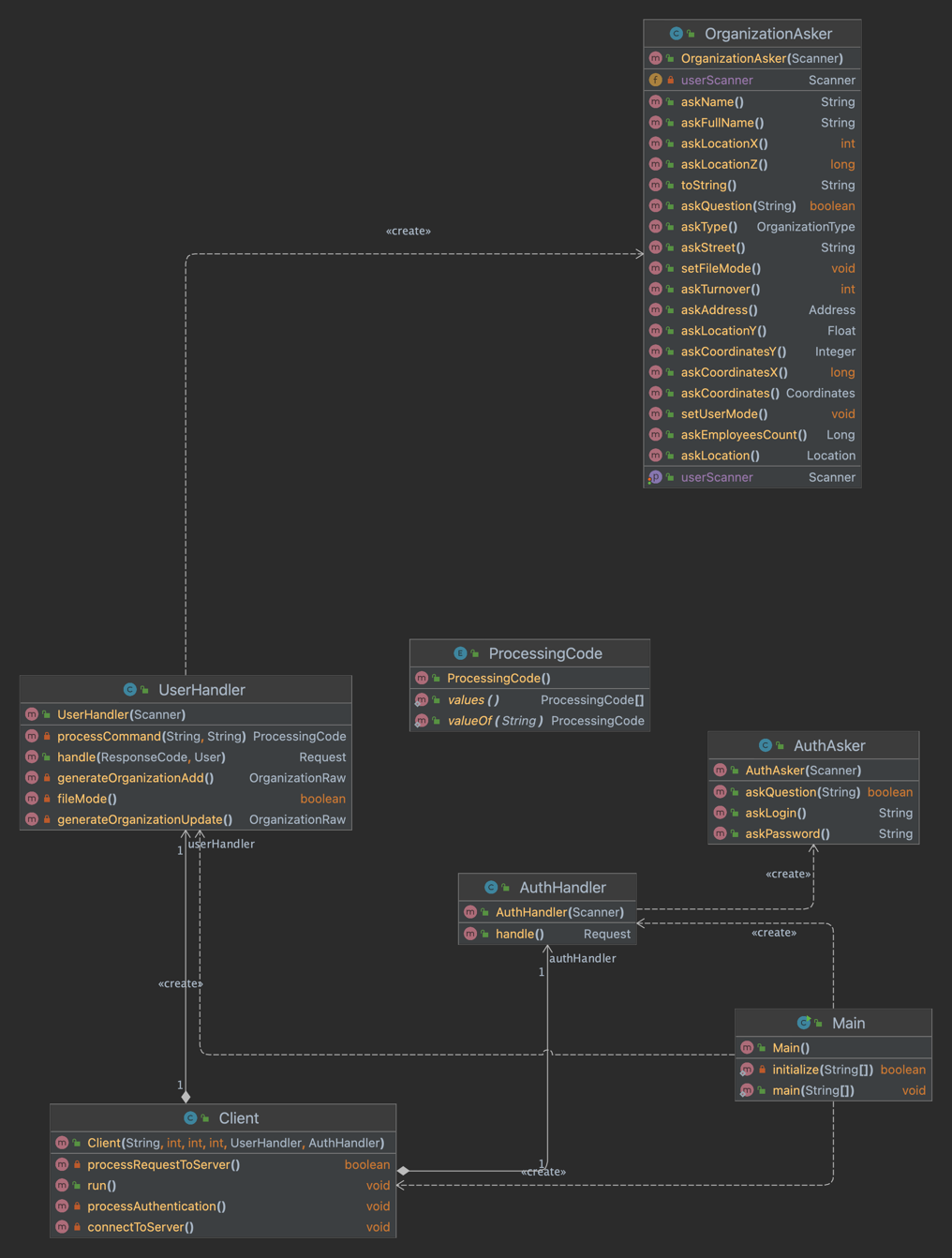
1. Для многопоточного чтения запросов использовать создание нового потока (java.lang.Thread)
2. Для многопотчной обработки полученного запроса использовать Fixed thread pool
3. Для многопоточной отправки ответа использовать Cached thread pool
4. Для синхронизации доступа к коллекции использовать синхронизацию чтения и записи с помощью java.util.concurrent.locks.ReadWriteLock

# Исходный код программы и uml-диаграмма

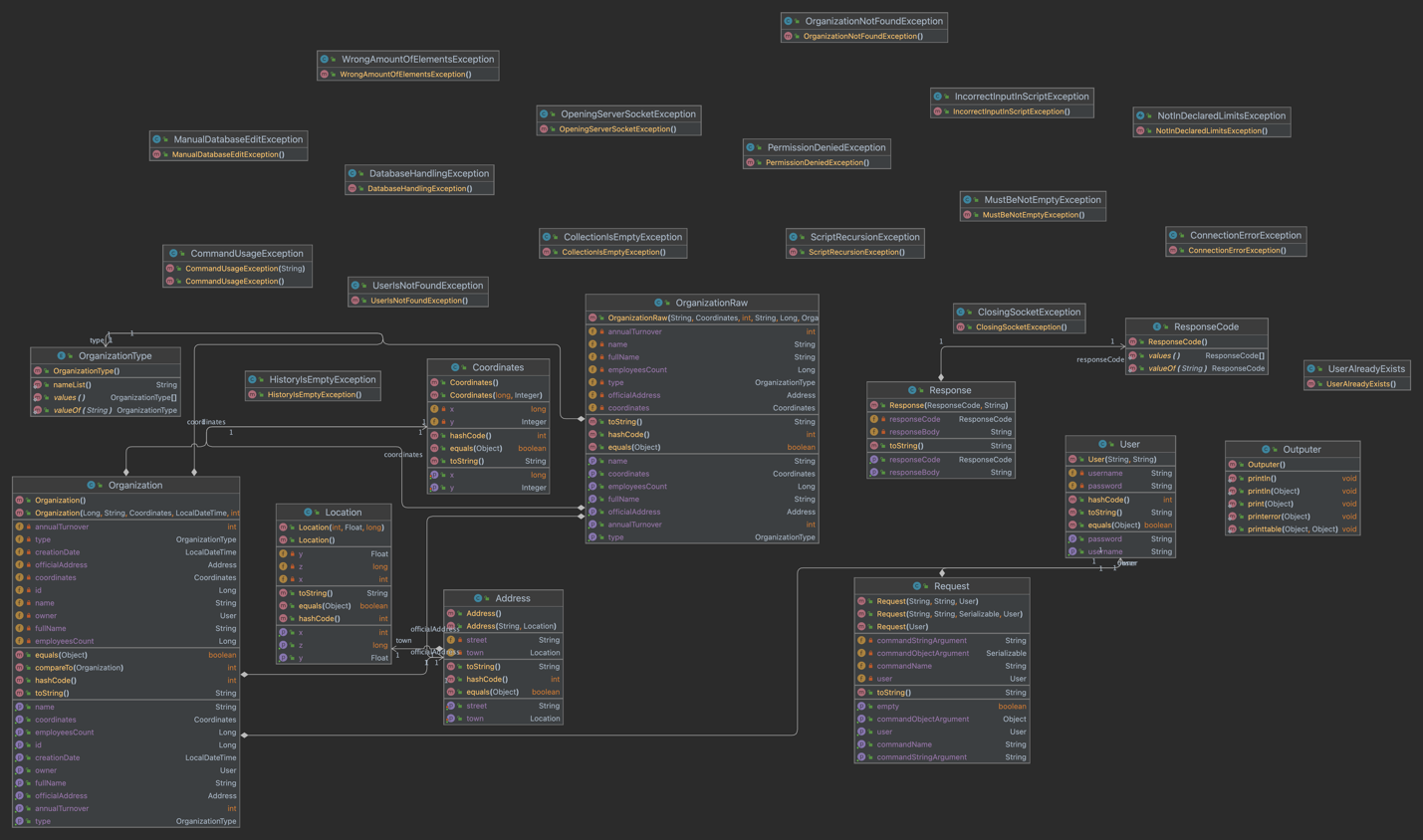
**Server:**

****

**Client:**

****

**Common:**

****

https://github.com/Sayronar/Lab7

# Вывод

В ходе выполнения данной лабораторной работы мы научились интегрировать базу данных в клиент-серверное приложение.

# Список литературы

**Academy Vertex Computer Science** Генерация случайных чисел в Java. - 2016 г.. - https://vertex-academy.com/tutorials/ru/generaciya-sluchajnyx-chisel-v-java/.

**Oracle** Java Documentation. - https://docs.oracle.com/en/java/.

**Михалевский Петр** Легкий старт в Java <...>. - Stepik. - https://stepik.org/course/90684/syllabus.