Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Программирование

**Лабораторная работа № 7**

Вариант № 3113800

Выполнил:

Сандов Кирилл Алекссевич

Группа:

P3113

Проверил:

преподаватель практики Письмак Алексей Евгеньевич

Санкт-Петербург

2023

# Оглавление

[Оглавление 2](#_Toc119848939)

[Задание 3](#_Toc119848940)

[Диаграмма классов модели 5](#_Toc119848941)

[Исходный код программы 6](#_Toc119848942)

[Результат работы программы 7](#_Toc119848943)

[Заключение 10](#_Toc119848944)

# Задание

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

* Организовать хранение коллекции в реляционной СУБД (PostgresQL). Убрать хранение коллекции в файле.
* Для генерации поля id использовать средства базы данных (sequence).
* Обновлять состояние коллекции в памяти только при успешном добавлении объекта в БД
* Все команды получения данных должны работать с коллекцией в памяти, а не в БД
* Организовать возможность регистрации и авторизации пользователей. У пользователя есть возможность указать пароль.
* Пароли при хранении хэшировать алгоритмом SHA-512
* Запретить выполнение команд не авторизованным пользователям.
* При хранении объектов сохранять информацию о пользователе, который создал этот объект.
* Пользователи должны иметь возможность просмотра всех объектов коллекции, но модифицировать могут только принадлежащие им.
* Для идентификации пользователя отправлять логин и пароль с каждым запросом.

Необходимо реализовать многопоточную обработку запросов.

* Для многопоточного чтения запросов использовать Fixed thread pool
* Для многопотчной обработки полученного запроса использовать создание нового потока (java.lang.Thread)
* Для многопоточной отправки ответа использовать создание нового потока (java.lang.Thread)
* 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.

# Диаграмма классов модели

## Приложение-сервер

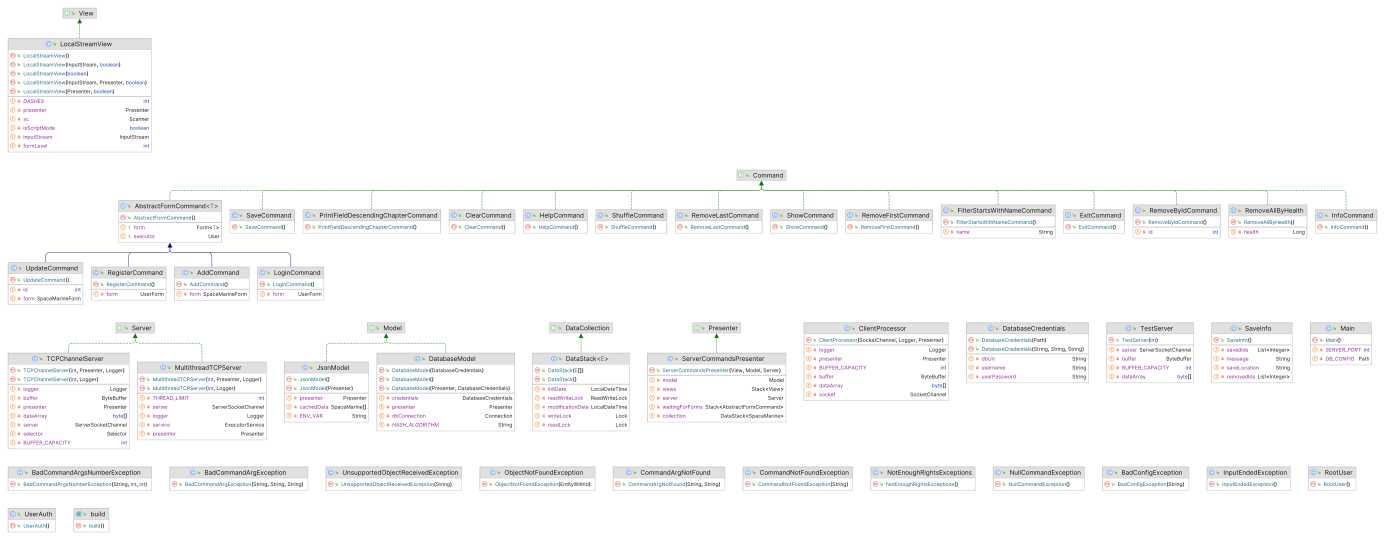


Рисунок 1 – «UML-диаграмма классов сервера»

## Приложение-клиент

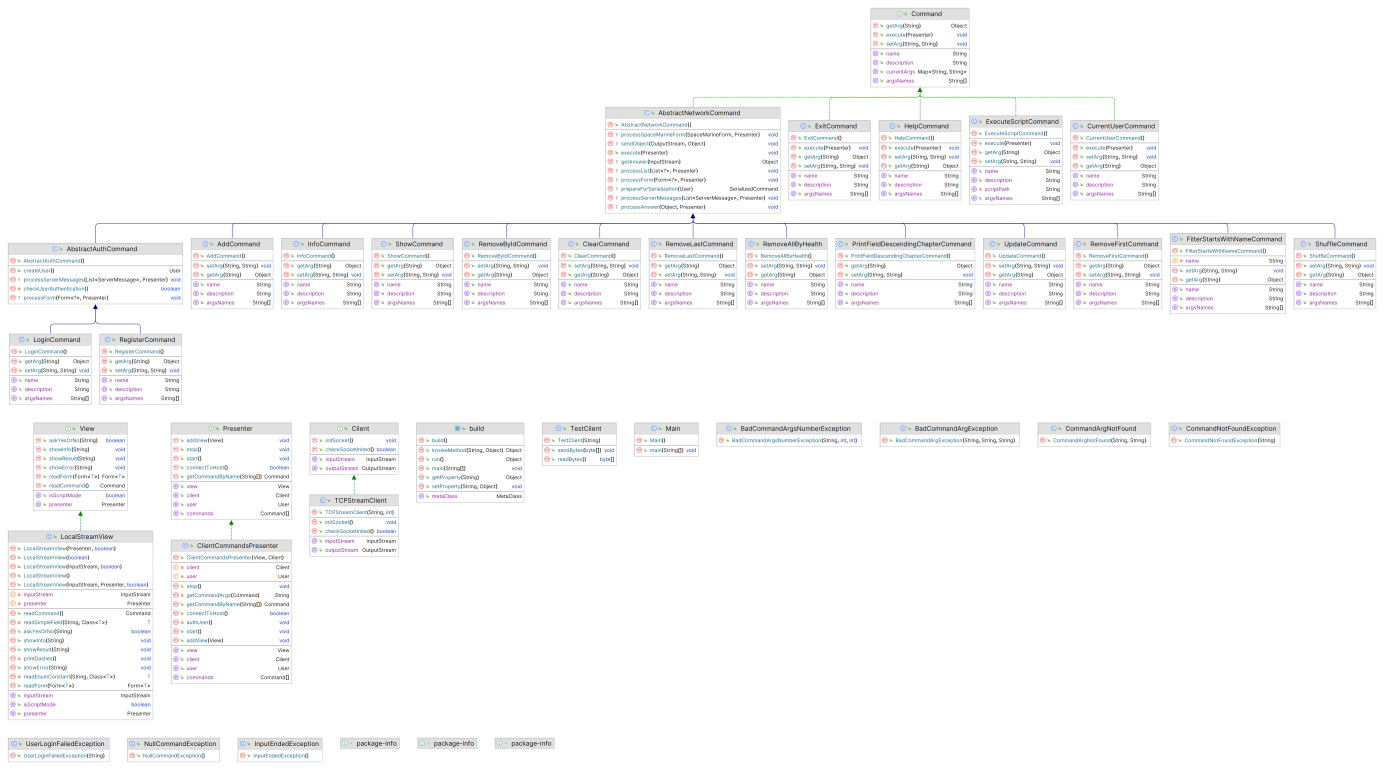


Рисунок 2 – «UML-диаграмма классов клиента»

# Исходный код программы

Ссылка на GitHub-репозиторий с реализацией:

<https://github.com/amphyxs/vt-labas/tree/main/sem-2/prog/lab-7>

# **Заключение**

В результате выполнения данной лабораторной работы изучены методы подключения БД к Java-приложению и взаимодействия с ней. Получены базовые знания об обеспечении безопасности хранения данных (например, хранение паролей в захешированном виде с «солью» и «перцем»). Также изучена организация многопоточности в программах: создание новых поток, пулы потоков, синхронизированный доступ к объектам, потокобезопасные коллекции.