

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники
(ФПИиКТ)

Программирование
Лабораторная работа №7
Вариант 311601

Выполнил

Григорьев Даниил Александрович

Группа Р3116

Санкт-Петербург 2025

Текст задания

Внимание! У разных вариантов разный текст задания!

Доработать программу из [лабораторной работы №6](#) 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-256**
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.

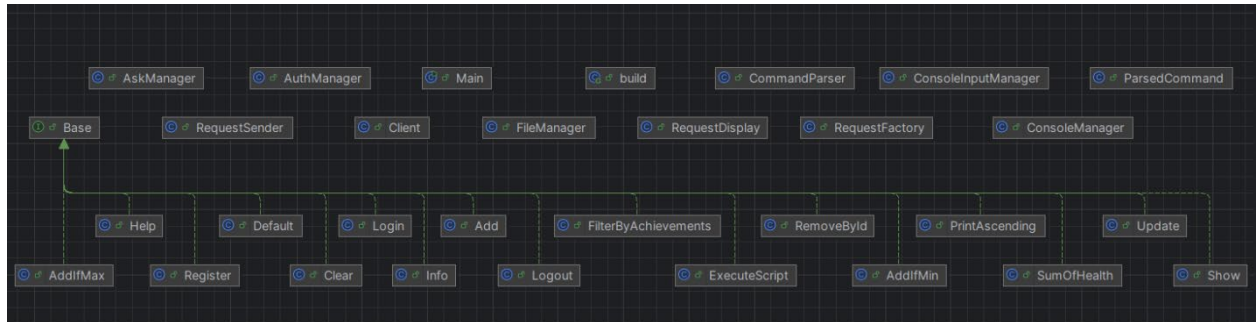
1. Для многопоточного чтения запросов использовать **создание нового потока (java.lang.Thread)**
2. Для многопоточной обработки полученного запроса использовать **Fixed thread pool**
3. Для многопоточной отправки ответа использовать **Cached thread pool**
4. Для синхронизации доступа к коллекции использовать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

Порядок выполнения работы:

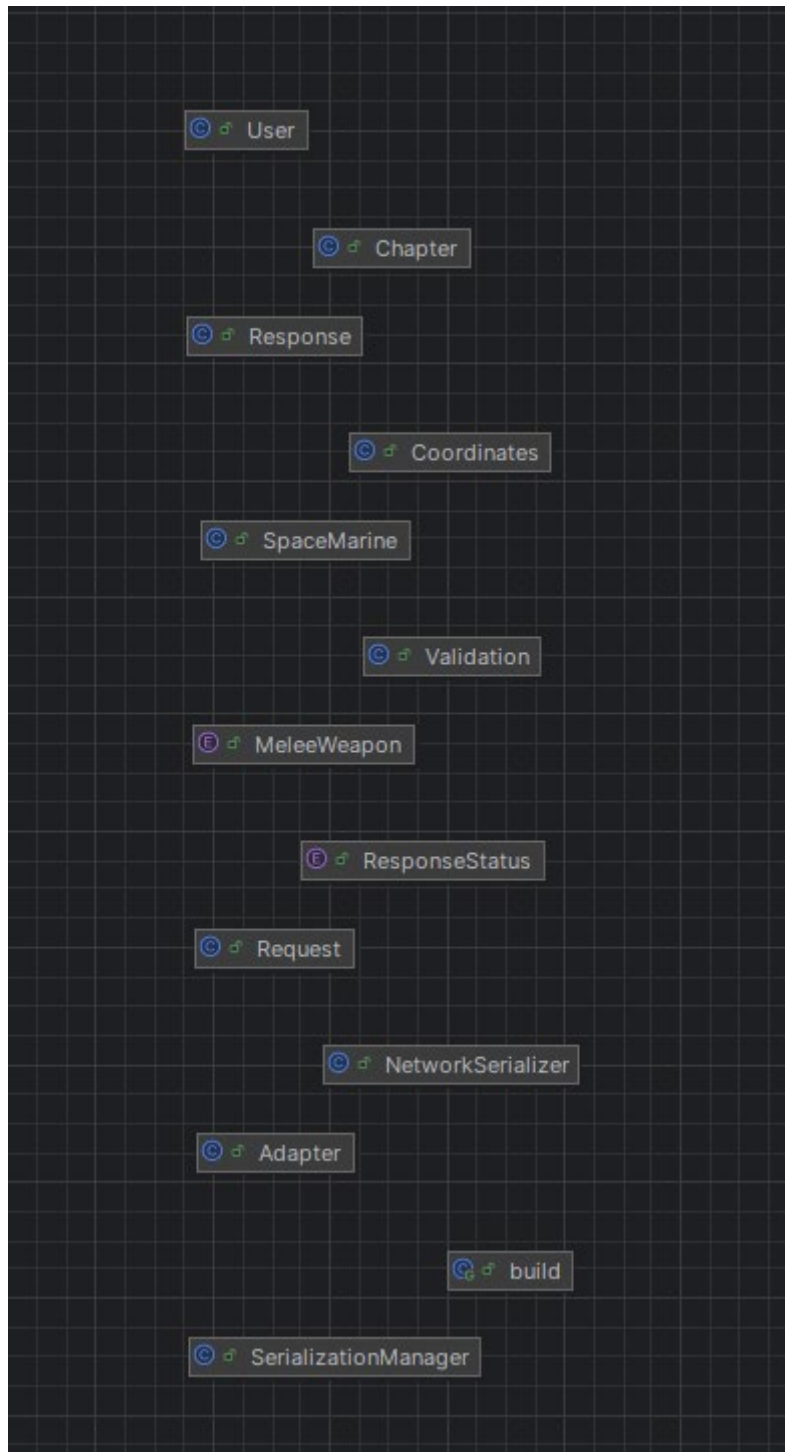
1. В качестве базы данных использовать PostgreSQL.
2. Для подключения к БД на кафедральном сервере использовать хост **pg**, имя базы данных - **studs**, имя пользователя/пароль совпадают с таковыми для подключения к серверу.

Диаграмма классов объектной модели

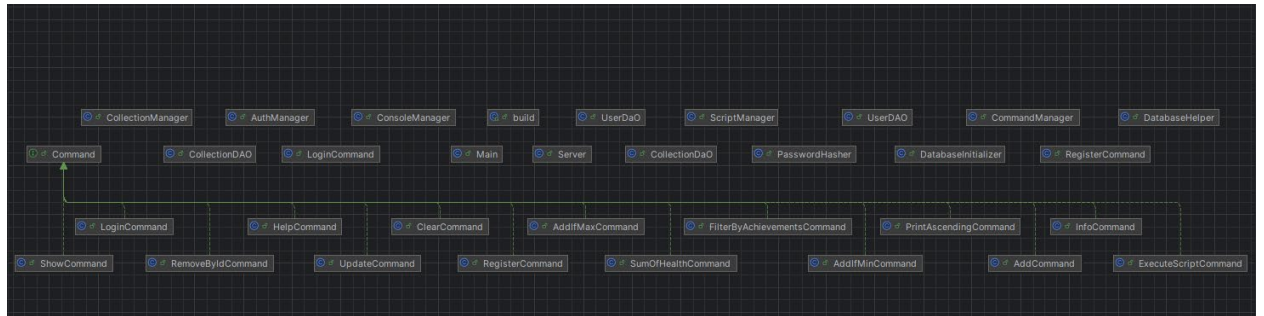
Client:



Common:



Server:



Исходный код программы

Представлен по ссылке:

<https://github.com/special-meme/labs-Megadeth2006>

Вывод

Я изучил многопоточность, разобрался с тем, как работать с базой данных и использовать ее в своей логике, я разобрался с пулом потоков, с коллекциями из пакета `java.util.concurrent`, познакомился с методами `wait()`, `notify()`, `notifyall()` класса `Object`, с интерфейсами `lock` и `condition`, с шаблонами проектирования.