**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**КЕМЕРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Институт цифры**

**ОТЧЕТ**

**По проекту**

По дисциплине «Языки программирования»

Студента 2 курса   
группы МОА-221   
Иванова Сергея

Направление 02.03.03 – «Математическое обеспечение и администрирование информационных систем»

Руководитель:

А.И. Зимин

Кемерово 2024

**Название проекта**:   
ChatMVC: Многопользовательский чат с графическим интерфейсом

**Состав участников**:  
Иванов Сергей Александрович, код и оформление

**Тема проекта**Проект представляет собой многопользовательский чат с графическим интерфейсом, реализованный на языке Java с использованием модели MVC (Model-View-Controller). Основной целью проекта является создание удобного и интуитивно понятного интерфейса для обмена сообщениями в реальном времени между пользователями. Пользователи могут подключаться к чату, отправлять и получать сообщения, а также видеть список активных участников чата.

Ключевые возможности включают:

1. Отправка и получение сообщений в режиме реального времени.
2. Поддержка нескольких пользователей одновременно.
3. Возможность запуска и остановки сервера через графический интерфейс.
4. Автоматическое обновление списка активных пользователей.
5. Логирование серверных сообщений с отображением их в отдельном окне консоли сервера.

**Архитектура**

Проект построен на архитектуре MVC, что разделяет логику приложения, пользовательский интерфейс и управление данными. Основные классы включают:

1. ChatModel: Отвечает за управление данными и логикой приложения, хранит список сообщений и активных пользователей.
2. ChatView: Предоставляет графический интерфейс для взаимодействия с пользователем. Включает окна для ввода и отображения сообщений, список пользователей, а также кнопки для подключения, отключения и управления сервером.
3. ChatController: Обрабатывает пользовательские действия, координирует работу модели и представления. Реализует логику подключения к серверу, отправки сообщений и управления состоянием приложения.
4. ChatServer: Реализует серверную часть приложения, обрабатывает входящие подключения и передает сообщения между клиентами.
5. ServerConsoleView: Отдельное окно для отображения логов сервера в реальном времени.

Использованы многопоточность для обработки нескольких подключений клиентов одновременно и межсетевое взаимодействие на основе сокетов для обмена сообщениями.

**Технологии**

Java: Основной язык программирования для реализации клиентской и серверной части.

Swing: Библиотека для создания графического пользовательского интерфейса.

Java Networking (java.net): Библиотека для реализации межсетевого взаимодействия с использованием сокетов.

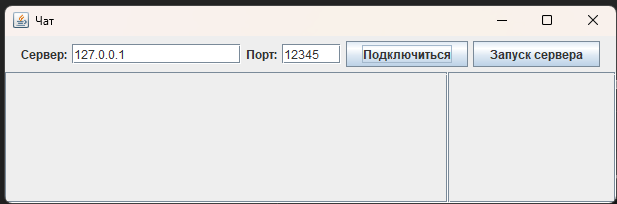
MVC (Model-View-Controller): Архитектурный паттерн, используемый для разделения логики приложения, интерфейса и управления данными.

Multithreading: Подход для обеспечения многопоточности, позволяющий серверу обрабатывать несколько клиентских подключений одновременно.

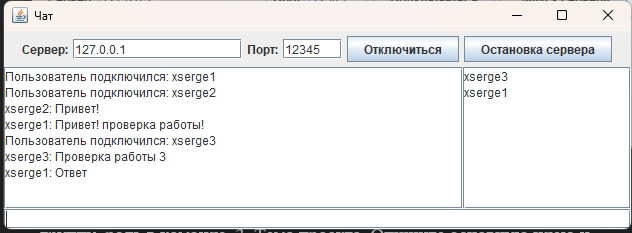
JOptionPane: Класс для реализации диалоговых окон, используемых для ввода никнейма пользователя и отображения сообщений об ошибках.

**Демонстрация работы**

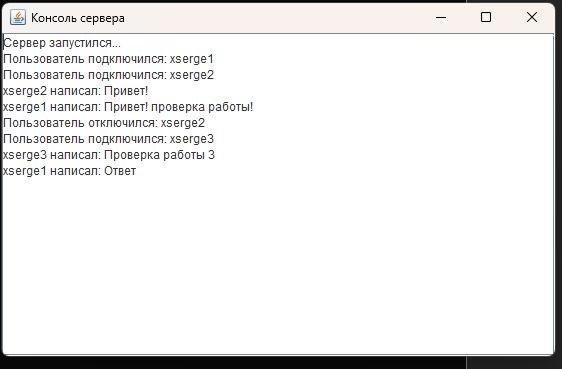
Окно приложения до подключения и запуска сервера:

****

После запуска сервера и подключения нескольких участников:

****

Консоль с логами сервера:

****

**Вклад каждого участника.**

<https://github.com/xiserge/kemsu-java>  
1 коммит, ник: xiserge, весь код и оформление проекта