

Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

по курсу «Основы операционных систем» на
тему «Система программ обмена сообщениями»

Handwritten signature in blue ink, followed by the date "22.06.2020" written vertically.

Выполнил:

студент группы 22ВВП2

Перкин П. О.

Принял:

к.т.н., доцент Егоров В.Ю.

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет Вычислительной техники

Кафедра "Вычислительная техника"

"УТВЕРЖДАЮ"

Зав. кафедрой ВТ

«___» _____ 20__

ЗАДАНИЕ

на курсовое проектирование по курсу

Основы операционных систем
Студенту Перингу Вадиму Олеговичу Группа 22ВВП2
Тема проекта система программ обмена
сообщений

Исходные данные (технические требования) на проектирование

Разработать систему программ клиент-серверной архитектуры, взаимодействующей по средствам вьюв разработанного протокола передачи данных.

Функционал системы включает в себя:

1. Идентификация пользователя на сервере по средствам клиента.
2. Передача текстовых сообщений между пользователями (индивидуальных и широковещательных, передача файлов)

Функционал сервера:

1. Организация взаимодействия между клиентами (передача сообщений, файлов)
2. Отношения передачи данных.

Функционал клиента:

1. Графический оконный интерфейс пользователя
2. Взаимодействие с сервером

Языки программирования: C, C++
Допускается расширение функционала существующих программ.

Объем работы по курсу

1. Расчетная часть

1. Разработка протокола взаимодействия программы (процессов)
2. Разработка программы.
3. Отладка программы и протокола

2. Графическая часть

Разработка не менее пяти UML диаграмм с описанием работы системы

3. Экспериментальная часть

Тестирование системы программ в локальной вычислительной сети

Срок выполнения проекта по разделам

- 1 В соответствии с графиком выполнения
- 2 работы.
- 3
- 4
- 5
- 6
- 7
- 8

Дата выдачи задания "27" сентября

Дата защиты проекта " " "

Руководитель

Задание получил "27" сентября 2024 г.

Студент Яерин П.О.

Содержание

Введение	7
1. Постановка задачи	9
2. Выбор решения	11
3. Описание разработки программы	Ошибка! Закладка не определена.
4. Отладка и тестирование	Ошибка! Закладка не определена.
5. Описание программы	Ошибка! Закладка не определена.
6. Руководство пользователя	Ошибка! Закладка не определена.
Заключение	13
Список используемых источников	14
Приложение А	Ошибка! Закладка не определена.
Приложение Б	Ошибка! Закладка не определена.
Результаты работы программы	Ошибка! Закладка не определена.

Введение

Клиент-серверная архитектура является одним из ключевых подходов в организации вычислительных систем и сетевых приложений, обеспечивающим распределение задач и ресурсов между клиентами и серверами. В рамках данной модели серверы предоставляют услуги или данные, а клиенты отправляют запросы на их получение. Эта архитектура используется в различных сферах, включая веб-разработку, системы мгновенного обмена сообщениями, обработку данных и другие приложения.

Клиент-серверная модель обеспечивает эффективное использование ресурсов и упрощает централизованное управление данными. Однако наряду с преимуществами, такими как высокая защищенность серверов и снижение требований к клиентским устройствам, она имеет свои недостатки, включая зависимость от производительности и доступности серверов.

В рамках данной курсовой работы разработано клиент-серверное приложение, предназначенное для обмена текстовыми сообщениями и передачи файлов. Основой проекта стала архитектура клиент-сервер, где взаимодействие реализовано через сетевые соединения с использованием библиотеки Qt и класса QTcpSocket.

Клиентская часть приложения разработана в среде Qt Creator на языке C++, с применением средств для создания графического интерфейса и обработки событий. Серверная часть, также написанная на языке C++, создана с использованием инструментов библиотеки Qt. Обе части взаимодействуют посредством TCP-соединений, что обеспечивает надежный и стабильный обмен данными.

Данная работа направлена на развитие практических навыков в проектировании клиент-серверных систем, использовании сетевых интерфейсов,

предоставляемых Qt, а также в разработке удобного и функционального пользовательского интерфейса

1. Постановка задачи

Разработать систему программ клиент-серверной архитектуры, взаимодействующей посредством вновь разработанного протокола, представленного на рисунке 1.

Функционал системы включает в себя:

1. Идентификацию пользователя по средствам протокола;
2. Передачу текстовых сообщений между пользователями (индивидуальных и широковещательных), пересылку файлов.

Функционал сервера включает в себя:

1. Организация взаимодействия между клиентами (передача сообщений, файлов и т.д.);
2. Отложенная передача данных.

Функционал клиента:

1. Графический оконный интерфейс пользователя;
2. Взаимодействие с сервером.

Используемый в разработке язык программирования C++. Для реализации графического оконного интерфейса использовать технологию Qt Creator.

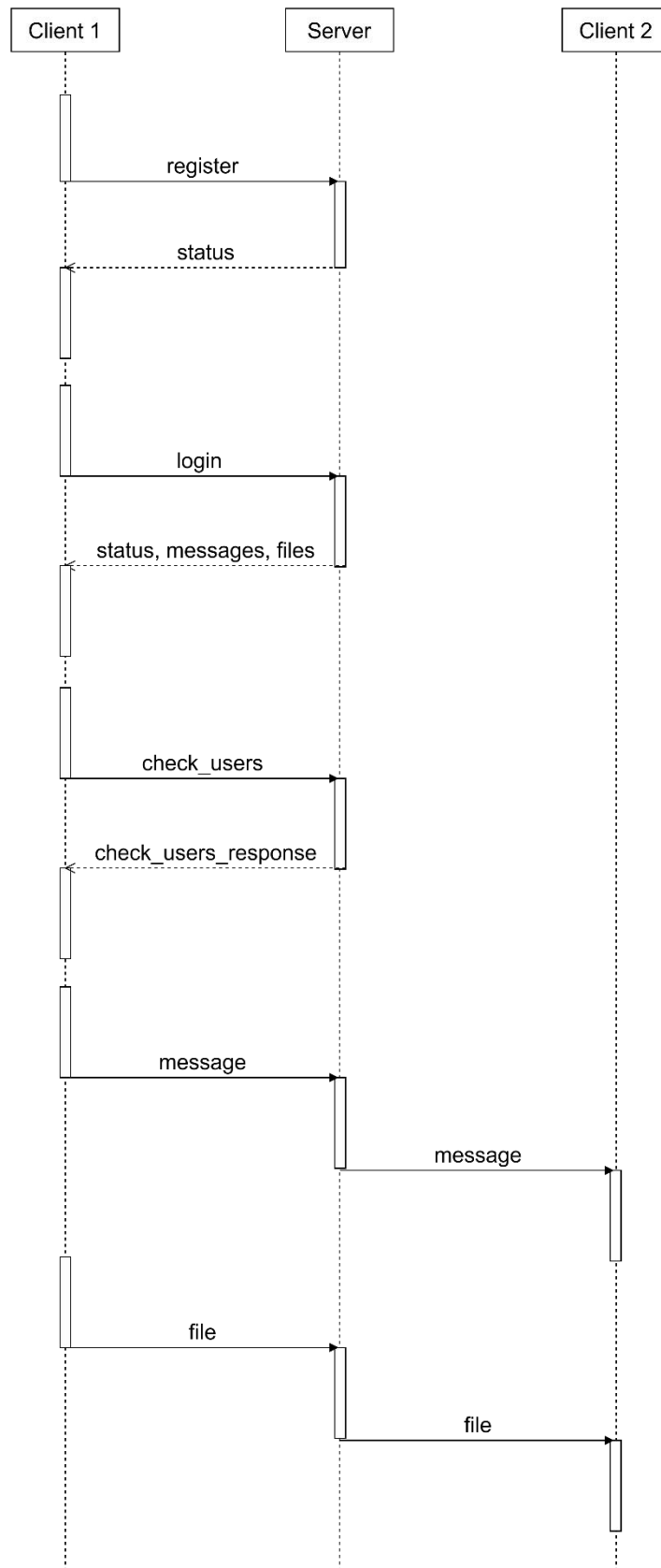


Рисунок 1 – UML-Диаграмма протокола взаимодействия клиента и сервера

2. Выбор решения

Приложение реализовано на языке C++ с использованием интерфейса Qt Creator.

C++ является компилируемым и статически типизированным языком программирования, который сочетает в себе преимущества высокоуровневых и низкоуровневых языков. Его возможности включают поддержку процедурного, объектно-ориентированного и обобщенного программирования, что позволяет разрабатывать как сложные, так и производительные приложения.

Ключевым аргументом в пользу использования C++ в данной работе стала его высокая производительность, что особенно важно для реализации серверной части приложения, обрабатывающей множество запросов от клиентов. Богатая стандартная библиотека языка, включающая контейнеры, алгоритмы и средства многопоточности, упрощает реализацию базовых функций и повышает стабильность программы.

Кроме того, C++ предоставляет гибкость в управлении ресурсами, что важно для сетевых приложений, работающих с сокетами и передачей данных. Для реализации сетевого взаимодействия и управления базой данных использовалась библиотека Qt, которая интегрируется с C++ и предоставляет мощные инструменты, такие как QTcpSocket и QSqlDatabase.

Для разработки клиентской части приложения была выбрана интегрированная среда разработки Qt Creator. Этот выбор обусловлен ее мощными возможностями, специально адаптированными для работы с фреймворком Qt, а также ее удобством и функциональностью.

Qt Creator поддерживает как классическое программирование на C++. Одним из ключевых преимуществ этой IDE является встроенный дизайнер

интерфейсов, позволяющий быстро и эффективно разрабатывать графические интерфейсы с использованием QtWidgets. Это особенно важно для приложений, где пользовательский интерфейс играет ключевую роль.

Qt Creator обеспечивает тесную интеграцию с фреймворком Qt, что позволило максимально эффективно использовать такие инструменты, как QTcpSocket для сетевого взаимодействия и QSqlDatabase для работы с базами данных.

Основная логика системы программы будет написана на языке C++. Графическая часть будет реализована с помощью интерфейса Qt Creator, обмен данными будет происходить с помощью QTcpSocket, а для хранения и управления данными будет использоваться QSqlDatabase с использованием СУБД SQLite.

Заключение

В процессе выполнения курсовой работы была реализована система программ обмена текстовыми сообщениями и файлами, были получены следующие навыки: программирования на языке C++, работы с фреймворком Qt, использования сетевых классов QTcpSocket и QTcpServer для организации сетевого взаимодействия, проработки архитектуры клиент-серверного программного обеспечения, разработки протоколов взаимодействия между клиентами и сервером, работы с СУБД SQLite, создания графического пользовательского интерфейса с помощью Qt Widgets, организации взаимодействия компонентов программы с использованием сигналов и слотов Qt.

Список используемых источников

1. Рихтер Дж. Windows для профессионалов: создание эффективных Win32-приложений с учетом специфики 64-разрядной версии Windows / Пер. с англ. – 4-е изд. – СПб.: Питер; М.: Издательство «Русская Редакция»; 2008. – 720 стр.
2. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектноориентированного проектирования. Паттерны проектирования. – СПб.: Питер, 2007.– 256 с. Калиткин Н.Н. Численные методы. – М.: Наука, 1978. – 267 с.
3. Керниган Брайан У., Ритчи Деннис М. Язык программирования С, 2-е изд. СПб: ООО «Диалектика», 2020. – 288 с.
4. Подбельский В.В., Фомин С.С. Курс программирования на языке Си. Учебник. Изд. 2-е, переработанное. М: ДМК Пресс, 2018 — 384 с.
5. Павловская Т.А. С/С++. Программирование на языке высокого уровня. Из серии «300 лучших учебников», 2003 — 461 с.