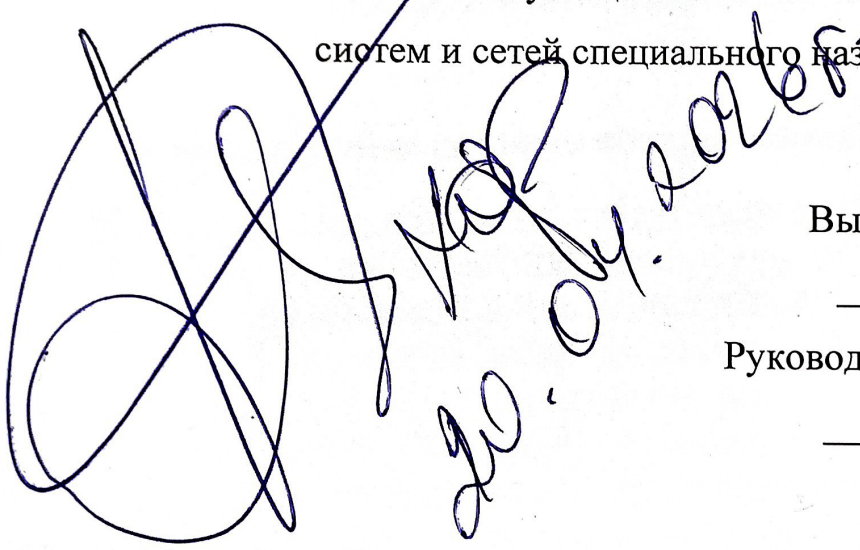
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Кафедра «Вычислительная техника»

Курсовая работа
по дисциплине «Системное программирование»
на тему «Система программ обмена сообщениями»

ПГУ 1.09.05.01

Специальность подготовки: 09.05.01 «Применение и эксплуатация автоматизированных систем специального назначения»
Специализация «Эксплуатация вычислительных машин, комплексов, систем и сетей специального назначения»



Выполнил: студент гр. 22ВОЭ1
_____Нагаев Д.А.
Руководитель: к.т.н., доцент каф. ВТ
_____Егоров В.Ю.

Работа защищена с оценкой _____
Преподаватели _____
_____
_____
Дата защиты _____

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет Вычислительной техники

Кафедра “Вычислительная техника”

“УТВЕРЖДАЮ”

Зав. кафедрой ВТ

профессор М. А. Митрохин

«____»_____20__г

ЗАДАНИЕ

на курсовое проектирование по курсу

Системное программирование

Студенту Ногаеву Дмитрию Алексеевичу Группа 22ВОЭ1

Тема проекта: Система программ обмена сообщениями

Исходные данные (технические требования) на проектирование

Разработать систему программ клиент-серверной архитектуры, позволяющую осуществлять взаимодействие пользователей на основе обмена сообщениями в рамках локальногй вычислительной сети. Взаимодействие программ осуществляется на основе вновь разрабатываемого протокола взаимодействия. Протокол взаимодействия должен включать в себя команды управления, команды передачи данных, информационные команды. Может быть представлен и состоять из серверной и клиентской частей. Клиентская программа включает в себя графический интерфейс пользователя и реализацию протокола взаимодействия. Серверная программа может вообще не иметь интерфейса, либо иметь консольный интерфейс реализующий серверную часть протокола взаимодействия. Необходимо предусмотреть передачу текстовых сообщений (личные и общие), файлов. Опционально идентификация, аунтефикация пользователя, обмен мультимедия информацией.

Язык программирования – C, C++, C#

Среда программирование – Visual Studio

Операционная система – любая

Объем работы по курсу

1. Расчетная часть

Разработка протоколов взаимодействия программ
Разработка и покомпонентная отладка программ
Комплексная отладка системы

2. Графическая часть

Диаграммы языка UML:
1) диаграмма классов;
2) диаграмма деятельности;
3) диаграмма вариантов использования;
4) диаграмма развёртывания;
5) диаграмма последовательности.

3. Экспериментальная часть

Проведение тестирования на институтской локальной вычислительной сети

Срок выполнения проекта по разделам

В соответствии с графиком выполнения работы

Дата выдачи задания "19" сентября 2025 г.

Дата защиты проекта "____"_____

Руководитель_____

Задание получил "31" октября 2025 г.

Студент Попов Дмитрий Алексеевич

Реферат

Пояснительная записка состоит из 123 листов, 22 рисунков, 1 таблицы и 5 источников.

Цель работы – закрепление знаний и приобретение практических навыков разработки системы программ клиент-серверной архитектуры, позволяющей осуществлять взаимодействие пользователей на основе обмена сообщениями в рамках локальной вычислительной сети.

Курсовая работа направлена на разработку системы программ клиент-серверной архитектуры, позволяющей осуществлять взаимодействие пользователей на основе обмена сообщениями в рамках локальной вычислительной сети.

Объект исследования для данной курсовой работы – это система программ клиент-серверной архитектуры, позволяющей осуществлять взаимодействие пользователей на основе обмена сообщениями в рамках локальной вычислительной сети

В результате курсовой работы будет разработано приложение "Текстовый чат".

Содержание

Введение.....6
1 Постановка задачи.....9
2 Выбор решения 10
3 Описание программы 12
4 Описание организации сетевого взаимодействия приложения.....15
5 Описание способа организации пользовательского интерфейса.....17
6 Описание результатов работы программы.....25
Заключение.....32
Список используемых источников.....34
Приложение А. Листинг программы.....35
Приложение Б. UML-диаграммы.....118

Введение

В современном мире, где общение и работа все чаще переходят в цифровую среду, высокопроизводительные и надежные системы обмена сообщениями становятся ключевой инфраструктурой. Для их создания необходим язык программирования, который обеспечивает максимальную эффективность, контроль над ресурсами и работу на низком сетевом уровне. Таким языком является C++. Его принцип «нулевых накладных расходов», прямое управление памятью, низкоуровневые возможности и кроссплатформенность делают его идеальным выбором для разработки приложений, где критически важны скорость, минимальные задержки и эффективное использование ресурсов.

Цель работы — практическая разработка клиент-серверной системы для обмена сообщениями с графическим интерфейсом на C++. Эта задача позволит глубоко освоить сетевое и многопоточное программирование, проектирование архитектуры распределенных приложений, создание протоколов и пользовательских интерфейсов под Windows.

Объект исследования — процесс создания клиент-серверных систем для обмена данными в реальном времени. Предмет исследования — методы и инструменты разработки таких систем на C++ с применением низкоуровневых сетевых интерфейсов и объектно-ориентированного подхода.

Методы, применяемые в работе:

- Анализ литературы и документации по сетевым технологиям (Winsock API) и многопоточному программированию.
- Проектирование архитектуры системы и формата данных для передачи.
- Поэтапная реализация и тестирование функциональных модулей.

Структура системы. Приложение будет построено по модульному принципу и состоять из трех основных компонентов:

Серверный модуль: отвечает за управление подключениями и маршрутизацию данных.

Клиентский модуль: предоставляет графический интерфейс для взаимодействия с пользователем.

Модуль протокола: обеспечивает надежную передачу сообщений и файлов.

Такая архитектура повышает читаемость и сопровождаемость кода, упрощает тестирование и позволяет повторно использовать компоненты.

Данная курсовая работа направлена на комплексное решение практической задачи. Ее результат — не только теоретическое понимание принципов построения распределенных систем, но и готовое к использованию клиент-серверное приложение, сочетающее производительное сетевое ядро с удобным интерфейсом.

1 Постановка задачи

Основная задача проекта заключается в создании системы программ клиент-серверной архитектуры, позволяющую осуществлять взаимодействие пользователей на основе обмена сообщениями в рамках локальной вычислительной сети. Взаимодействие программ осуществляется на основе вновь разрабатываемого протокола взаимодействия. Протокол взаимодействия должен включать в себя команды управления, команды передачи данных, информационные команды. Может быть представлен в текстовом или бинарном виде и состоять из серверной и клиентской частей.

Клиентская программа включает в себя графический интерфейс пользователя и протокол взаимодействия. Серверная программа может вообще не иметь интерфейса. Необходимо предусмотреть передачу текстовых сообщений, файлов.

Средой разработки была выбрана Visual Studio — свободная интегрированная среда разработки приложений (IDE) на языках программирования C, C++, C#, которая будет использоваться для написания, компиляции и отладки кода.

Заключение

В ходе выполнения курсовой работы была успешно разработана и реализована клиент-серверная система обмена сообщениями «Текстовый чат», предназначенная для взаимодействия пользователей в рамках локальной вычислительной сети. Поставленная цель — закрепление знаний и приобретение практических навыков в области сетевого и многопоточного программирования, проектирования архитектуры распределённых приложений и создания графических интерфейсов — достигнута в полном объёме.

Основные результаты работы:

Разработано клиент-серверное приложение на языке C++, состоящее из двух независимых модулей: серверной части, управляющей подключениями и маршрутизацией данных, и клиентской части с интуитивно понятным графическим интерфейсом на базе WinAPI.

Реализован собственный коммуникационный протокол на основе TCP, обеспечивающий надёжную передачу текстовых сообщений и файлов, поддержку приватных и групповых чатов, а также динамическое обновление списка пользователей.

Проведено тестирование работы системы в локальной сети, подтвердившее корректность взаимодействия клиентов и сервера, стабильность передачи данных и отказоустойчивость при разрывах соединения.

В процессе работы были изучены и применены ключевые технологии и инструменты: Winsock API для сетевого взаимодействия, многопоточное программирование для обработки одновременных подключений, объектно-ориентированный подход для структурирования кода, а также принципы проектирования пользовательских интерфейсов.

Таким образом, разработанная система является законченным, функциональным решением, демонстрирующим возможность создания производительных и масштабируемых клиент-серверных приложений на C++. Работа подтвердила теоретические знания, полученные в рамках дисциплины «Системное программирование», и предоставила ценный практический опыт, который может быть использован в дальнейшем для решения более сложных задач в области сетевых технологий и распределённых систем.

Список использованных источников

1. WinAPI: создание окон и элементов управления [Электронный ресурс]. — Режим доступа: https://docs.microsoft.com/en-us/windows/win32/learnwin32/creating-a-window (дата обращения: 6.11.2025).

2. Создание Mutex C++ [Электронный ресурс]. — Режим доступа: https://www.reddit.com/r/C_Programming/comments/135udh5/how_do_you_implement_a_mutex/?tl=ru (дата обращения: 8.11.2025).

3. Разработка своего сетевого протокола C++ [Электронный ресурс]. — Режим доступа: https://programmersforum.ru/showthread.php?t=302051 (дата обращения: 10.11.2025).

4. Microsoft Docs. Winsock (Windows Sockets) [Электронный ресурс]. — Режим доступа: https://docs.microsoft.com/en-us/windows/win32/winsock/windows-sockets-start-page-2 (дата обращения: 12.11.2025).

5. Передача файлов по протоколу TCP [Электронный ресурс]. — Режим доступа: https://www.binarytides.com/code-a-simple-socket-client-class-in-c/ (дата обращения: 15.11.2025).