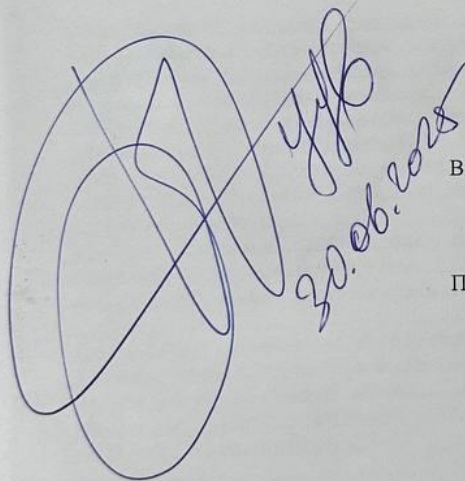


Министерство науки и высшего образования Российской Федерации
Пензенский государственный университет
Кафедра «Вычислительная техника»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе
по курсу «Архитектура операционных систем»
на тему «Система программ обмена сообщениями»



Handwritten signature and date: 20.06.2025

Выполнил:

студент группы 22BVB1

Советкин Д.Д.

Принял:

к.т.н., доцент Егоров В.Ю.

Пенза 2025

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

Архитектура операционных систем

Студенту Саваткину Владимиру Викторовичу Группа 22 ВВ01

Тема проекта Система управления обменом сообщениями

Исходные данные (технические требования) на проектирование

Разработать систему управления кешем - серверной архитектурой взаимодействия для поддержки работы параллельного программного приложения. Приложение должно обеспечивать взаимодействие на сервере с кешем кешем, передачу текстовых сообщений между пользователями, передачу файлов.

Функции сервера: организация взаимодействия между клиентами (передача сообщений, файлов, отложенная передача данных).

Функции клиента: графический оконный интерфейс пользователя, взаимодействие с сервером.

Языки программирования: С, С++, С#.

Функциональные расширения функционала системы управления, в том числе: маршрутизация потоков данных, передача мультимедийной информации, нахождение статистических данных на сервере потоков работы, ведение протоколов работы, нормализация С/Т/Р для хранения информации о потоках.

Объем работы по курсу

1. Расчетная часть

1. Разработка потока взаимодействия программы
2. Разработка программы
3. Описание программы и протокола

2. Графическая часть

Работа с UML диаграммами с графическими работами

3. Экспериментальная часть

Испытание системы программы в локальной информационной среде.

Срок выполнения проекта по разделам

1. В соответствии с утвержденным графиком работ
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.

Руководитель

Задание получил

Студент

Дата выдачи задания "24" сентября 2024г.

Дата защиты проекта " " "

"24" сентября

Соловьев Д.В.

2024г.

Введение.....

Основная часть.....

1. Постановка задачи.....

2. Теоретические сведения.....

3. Выбор решения.....

4. Описание Протокола.....

5. Описание программы.....

6. Руководство пользователя.....

Заключение.....

Список используемых источников.....

Приложение А.....

Листинг программы клиента

AuthForm.cs.....

Ip.cs.....

Messenger.cs.....

Program.cs.....

Листинг программы сервера

Server.cs.....

Приложение В.....

Содержание

Введение.....	4
Основная часть.....	5
1. Постановка задачи.....	5
2. Теоретические сведения.....	6
3. Выбор решения.....	7
4. Описание Протокола.....	8
5. Описание программы.....	11
6. Руководство пользователя.....	19
Заключение.....	24
Список используемых источников.....	26
Приложение А.....	27
Листинг программы клиента.....	27
AuthForm.cs.....	27
Ip.cs.....	28
Messenger.cs.....	29
Program.cs.....	39
Листинг программы сервера.....	41
Server.cs.....	41
Приложение В.....	56

Введение

Современное общество трудно представить без технологий обмена данными и эффективной коммуникации. В эпоху стремительного развития информационных технологий мессенджеры занимают важное место среди средств общения, предлагая пользователям возможность обмениваться сообщениями, файлами и мультимедиа в режиме реального времени.

Данная курсовая работа посвящена созданию программного обеспечения, позволяющего осуществлять обмен текстовыми сообщениями и файлами в локальной сети. Основная цель работы – разработка удобного и функционального мессенджера, способного удовлетворить базовые потребности пользователей в коммуникации. Приложение ориентировано на упрощение взаимодействия между пользователями и реализацию надежного способа передачи данных.

В процессе выполнения работы будет рассмотрено множество аспектов, связанных с разработкой приложения, таких как проектирование архитектуры, реализация многопользовательского доступа, обеспечение авторизации и аутентификации пользователей, а также обработка и передача файлов. Особое внимание будет уделено созданию удобного интерфейса и обеспечения безопасности передаваемой информации.

Результатом работы станет полноценное приложение, способное обеспечивать удобное и безопасное взаимодействие пользователей в локальной сети. Полученный опыт в проектировании и разработке мессенджеров может быть полезен не только для дальнейшего совершенствования программы, но и для разработки аналогичных решений в других областях.

Основная часть

1. Постановка задачи

Цель работы заключается в разработке клиент-серверной архитектуры, предназначенной для обмена текстовыми сообщениями в локальной вычислительной сети. Необходимо создать систему, в которой клиенты могут взаимодействовать с сервером на основе вновь создаваемого протокола обмена данными. Необходимой частью является наличие графического интерфейса пользовательского приложения и консольного интерфейса сервера.

Требуется разработать протокол обмена данными, который обеспечит передачу информации между клиентами и сервером. Графический интерфейс пользовательского приложения должен поддерживать отправку сообщений. Консольный интерфейс сервера предоставит возможность мониторинга обмена сообщениями.

Так же дополнительно система может предоставлять возможности идентификации и авторизации пользователей, что повышает уровень безопасности и обеспечивает защищенный доступ к обмену данными.

Язык программирования – C/C++/C#.

Операционная система – windows.

2. Теоретические сведения

Программы, которые функционируют в Интернете, обмениваются данными через специальные наборы правил и инструкций, известные как протоколы. Эти протоколы определяют способы передачи информации между устройствами и порядок действий для отправки и получения данных.

Одним из основных протоколов, обеспечивающих работу Интернета, является протокол TCP/IP. Он состоит из двух компонентов: TCP (протокол управления передачей) и IP (интернет-протокол). TCP отвечает за управление процессом передачи данных: разбивает информацию на пакеты, отправляет их и обеспечивает их правильную доставку. Кроме того, TCP осуществляет контроль над ошибками передаваемых данных и гарантирует последовательную сборку информации на стороне получателя.

IP, в свою очередь, занимается маршрутизацией пакетов. Каждый пакет информации содержит IP-адреса как отправителя, так и получателя. Исходя из этих адресов, маршрутизаторы — специализированные сетевые устройства — определяют наиболее оптимальный маршрут для доставки данных к целевому пункту.

Для работы прикладных программ, таких как мессенджеры или клиенты электронной почты, применяются дополнительные протоколы, которые регулируют процессы передачи данных и авторизации пользователей. Например, чтобы получить доступ к электронной почте, программа может запросить ввести пароль, что подразумевает выполнение заранее установленной последовательности действий.

Таким образом, благодаря множеству различных протоколов обеспечивается согласованное и эффективное взаимодействие программ и устройств в Интернете, что делает использование современных сетевых технологий удобным и доступным.

3. Выбор решения

Для реализации графического интерфейса клиентской части использовалась технология WinForms. Она предоставляет удобные инструменты для разработки приложений под Windows, позволяя легко интегрировать такие элементы управления, как текстовые поля, кнопки и списки, в функциональную структуру мессенджера.

Взаимодействие между сервером и клиентами осуществляется с использованием протокола TCP/IP, который гарантирует надежную передачу данных благодаря механизму подтверждений и повторной отправки в случае ошибок. Этот протокол широко используется в приложениях, таких как HTTP, FTP, SMTP и Telnet. TCP требует предварительной установки соединения, где сервер ожидает входящих запросов, а клиент инициирует связь, отправляя синхронизирующий порядковый номер.

В системе применяется топология "звезда": каждый клиент подключается к серверу, который выступает в роли центрального узла. Для хранения данных пользователей используется база данных SQLite, что позволяет эффективно управлять информацией о пользователях и их состоянии.

Для обеспечения безопасного обмена данными между клиентами и сервером был реализован механизм идентификации и авторизации пользователей. Это позволяет контролировать доступ к системе и предотвращать несанкционированное использование.

Дополнительно для повышения удобства работы с приложением были внедрены функции, такие как отображение статуса пользователей в сети.

✓

Заключение

В результате реализации данной курсовой работы был создан мессенджер, который обеспечивает эффективное и безопасное взаимодействие среди пользователей в локальной сети. Ключевым достижением стало разработка клиент-серверного приложения, способного передавать текстовые сообщения и файлы с использованием надежного протокола TCP/IP.

В процессе разработки приложения были успешно решены важные задачи: проектирование архитектуры системы, создание протокола для обмена данными и реализация процессов авторизации и аутентификации пользователей. Значительное внимание было уделено обеспечению удобства взаимодействия пользователей с приложением.

Клиентская часть программы была создана на базе технологии WinForms, что дало возможность разработать простой и интуитивно понятный интерфейс. Серверная часть, в свою очередь, обеспечивает административные инструменты для управления и мониторинга состояния системы через консольный интерфейс, что повысило гибкость и удобство в работе с приложением.

Разработанный протокол обмена данными стал основой для эффективной передачи сообщений и файлов между клиентами. Надежность протокола достигается за счет наличия механизмов контроля ошибок и повторных попыток передачи, что является важным аспектом для стабильной работы системы в условиях локальной сети.

Опыт, полученный в ходе разработки мессенджера с возможностью обмена текстовыми сообщениями и файлами, может оказать большую пользу для дальнейших улучшений программы, а также для создания аналогичных решений в других областях. В будущем предполагается дополнить приложение новыми функциональными возможностями, включая поддержку



мультимедийных сообщений, усовершенствованные механизмы защиты данных и расширенную функциональность для администраторов сети.

В завершение, выполненная работа позволила не только получить теоретические знания, но и практические навыки в разработке многопользовательских приложений с использованием клиент-серверной архитектуры, а также углубило понимание процессов передачи данных и сетевых технологий.