Белорусский государственный университет

Институт непрерывного образования

Кафедра прикладной математики и информатики

КУРСОВОЙ ПРОЕКТ

на тему

Программное средство «Сетевой чат»

Слушатель группы 15ПО11-04з

по специальности 1-40 01 73

«Программное обеспечение

информационных систем»

Селюк Иван Чеславович

Руководитель Кравцова А.А.

Допущен к защите:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, дата)

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Зав. кафедрой Лесун Б.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, ФИО, дата)

Минск 2013

Белорусский государственный университет

Институт непрерывного образования

Кафедра прикладной математики и информатики

|  |  |
| --- | --- |
| Группа 15ПО11-04з | УТВЕРЖДАЮ  Зав. кафедрой  Б.В. Лесун  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_» \_\_\_\_\_\_\_\_\_ 20\_\_ г. |

ЗАДАНИЕ

на курсовое проектирование

Слушателю Селюку Ивану Чеславовичу

1 Тема проекта Программное средство «Сетевой чат»

2 Срок сдачи законченного проекта «08» июля 2013 г.

3 Исходные данные по проекту:

Программное средство разрабатывается в архитектуре клиент-сервер на объектно-ориентированном языке программирования. Необходимо разработать собственную иерархию классов с использованием сокрытия данных (инкапсуляция), наследования, виртуальных методов (полиморфизм), перегрузки и переопределения методов. Так же можно использовать интерфейс программирования приложений (API) предоставляемый операционной системой. Предусмотреть обработку исключительных ситуаций.

Необходимо:

Разработанное программное средство позволяет обмениваться текстовыми сообщениями в локальной сети, поддерживает функции регистрации нового пользователя, входа и выхода из системы зарегистрированных пользователей, имеет возможность настройки сетевых подключений.

4 Состав проекта:

а) пояснительная записка:

Введение

Основная часть

- Анализ и разработка требований

- Проектирование программного средства

- Реализация программного средства

- Тестирование программного средства

- Руководство пользователя

Заключение

б) графическая часть проекта:

- Диаграмма вариантов использования

- Диаграмма взаимодействия

- Диаграмма классов

- Информационная модель

- Интерфейс пользователя и др.

- Листинг реализации отправки сообщения

- Листинг реализации получения сообщения

5 Календарный график работы на весь период проектирования:

17.06.2013 - 20.06.2013 – анализ и разработка требований;

20.06.2013 – 23.062013 – программное проектирование;

23.06.2013 – 30.06.2013 – реализация программного средства;

30.06.2013 – 02.07.2013 – тестирование программного средства;

02.07.2013 – 07.07.2013 – оформление пояснительной записки;

Руководитель проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.А. Кравцова

Дата выдачи задания «17» июня 2013 г.

Задание принял к исполнению: «17» июня 2013 г.

Подпись слушателя: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

АННОТАЦИЯ

к курсовому проекту Селюка Ивана Чеславовича

слушателя группы 15ПО11-04з по специальности 1-40 01 73   
«Программное обеспечение информационных систем»

на тему

«Программное средство «Сетевой чат»

Ключевые слова: ПРОГРАММНОЕ СРЕДСТВО, СООБЩЕНИЕ, СОКЕТ, ЛОКАЛЬНАЯ СЕТЬ, СЕТЕВОЙ ПРОТОКОЛ, СТЕК СЕТЕВЫХ ПРОТОКОЛОВ.

Предметной областью курсового проектирования является общение пользователей в локальной сети, с помощью передачи текстовых сообщений.

Целью курсового проекта является создание программного средства, которое позволит обмениваться текстовыми сообщениями в локальной сети и, тем самым, организовать систему общения между работниками конкретного предприятия.

При выполнении курсового проектирования были использованы среда визуального программирования Visual Studio 2010, среда проектирования информационных систем Rational Rose 2000.

Пояснительная записка к курсовому проекту включает перечень сокращений, перечень определений, введение, пять глав и заключение, содержит 56 страниц, в том числе 3 приложения, 4 таблицы, 2 схемы, 54 литературных источника.

**СОДЕРЖАНИЕ**

|  |  |
| --- | --- |
| Определения…………………………………………………………………..... | 6 |
| Введение………………………………………………………………………... | 8 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**ОПРЕДЕЛЕНИЯ**

Локальная вычислительная сеть (англ. Local Area Network, LAN) – компьютерная сеть, покрывающая обычно относительно небольшую территорию или небольшую группу зданий (дом, офис, фирму, институт).

Ethernet – пакетная технология передачи данных преимущественно локальных компьютерных сетей. Стандарты Ethernet определяют проводные соединения и электрические сигналы на физическом уровне, формат кадров и протоколы управления доступом к среде — на канальном уровне модели OSI.

Протокол передачи данных – набор соглашений интерфейса логического уровня, которые определяют обмен данными между различными программами. Эти соглашения задают единообразный способ передачи сообщений и обработки ошибок при взаимодействии программного обеспечения, разнесённой в пространстве аппаратуры, соединённой тем или иным интерфейсом.

Стек протоколов – иерархически организованный набор сетевых протоколов, достаточный для организации взаимодействия узлов в сети.

Межпроцессное взаимодействие (англ. Inter-Process Communication, IPC) – набор способов обмена данными между множеством потоков в одном или более процессах. Процессы могут быть запущены на одном или более компьютерах, связанных между собой сетью. IPC-способы делятся на методы обмена сообщениями, синхронизации, разделяемой памяти и удаленных вызовов (RPC). Методы IPC зависят от пропускной способности и задержки взаимодействия между потоками и типа передаваемых данных.

Сокет (англ. socket – разъём) – название программного интерфейса для обеспечения обмена данными между процессами. Процессы при таком обмене могут исполняться как на одной ЭВМ, так и на различных ЭВМ, связанных между собой сетью. Сокет – абстрактный объект, представляющий конечную точку соединения.

UDP (англ. User Datagram Protocol – протокол пользовательских датаграмм) – один из ключевых элементов Internet Protocol Suite, набора сетевых протоколов для Интернета. С UDP компьютерные приложения могут посылать сообщения (в данном случае называемые датаграммами) другим хостам по IP-сети без необходимости предварительного сообщения для установки специальных каналов передачи или путей данных.

Дейтаграмма (англ. datagram), также датаграмма – блок информации, посланный как пакет сетевого уровня через передающую среду без предварительного установления соединения и создания виртуального канала. Датаграмма представляет собой единицу информации в протоколе (protocol data unit, PDU) для обмена информацией на сетевом и транспортном (в случае протокола UDP, UDP-датаграммы) уровнях эталонной модели OSI.

Многопоточность – свойство платформы (например, операционной системы, виртуальной машины и т. д.) или приложения, состоящее в том, что процесс, порождённый в операционной системе, может состоять из нескольких потоков, выполняющихся «параллельно», то есть без предписанного порядка во времени. При выполнении некоторых задач такое разделение может достичь более эффективного использования ресурсов вычислительной машины.

Чат – средство обмена сообщениями по компьютерной сети в режиме реального времени, а также программное обеспечение, позволяющее организовывать такое общение.

Система обмена мгновенными сообщениями (англ. Instant messaging, IM) – служба мгновенных сообщений (Instant Messaging Service, IMS), программы онлайн-консультанты (OnlineSaler) и программы-клиенты (Instant Messenger, IM) для обмена сообщениями в реальном времени через Интернет. Могут передаваться текстовые сообщения, звуковые сигналы, изображения, видео, а также производиться такие действия, как совместное рисование или игры. Многие из таких программ-клиентов могут применяться для организации групповых текстовых чатов или видеоконференций.

**ВВЕДЕНИЕ**

С развитием информационных технологий стали возможными ещё более глобальные коммуникации. Историческим «докомпьютерным» предшественником чатов, несомненно, был телефон. Ни почта, ни телеграф не позволяли общаться в режиме реального времени, и не были доступны в домашней обстановке. Изобретение и распространение телефона по планете вызвало настоящую революцию в средствах и способах общения. Возможность поговорить с собеседником на другой стороне Земли казалась настоящим чудом.

Во второй половине XX века начали бурно развиваться компьютеры. Однако долгое время они были большими и слишком дорогими, что препятствовало тому, чтобы расходовать драгоценное машинное время на забавы с обменом сообщениями вместо расчётов атомных бомб. К тому же до конца 60-х годов они не были связаны друг с другом. Предок Интернета, сеть ARPANET, в 1969 году насчитывала только четыре связанных друг с другом научных компьютера. Чуть позже, в 1971 году, была придумана электронная почта, которая стала необычайно популярна ввиду своего удобства. Постепенно появились новые службы сообщений, такие, как списки почтовой рассылки, новостные группы и доски объявлений. Однако в то время сеть ARPANET ещё не могла легко взаимодействовать с другими сетями, построенными на других технических стандартах, что затрудняло её распространение.

Программы для обмена текстовыми строками, несмотря на простоту самой идеи, появились не сразу. Примерно в 1974 году для мэйнфрейма PLATO был разработана программа Talkomatic, потенциально позволявшая общаться между тысячей терминалов системы. В 1980-x появилась система Freelancing' Round table. Однако по-настоящему популярным стал разработанный в 1988 году протокол, названный Internet Relay Chat (IRC), что примерно можно перевести как ретранслируемый интернет-разговор. Где-то в это же время появилось и распространилось само понятие «чат».

Обычно под словом «чат» подразумевается обмен текстовыми сообщениями, реже встречаются голосовой чат, видеочат (или видеоконференция). Существует огромное множество средств общения с использованием сетевых технологий. Например: интернет форумы, Web-чаты, программы для обмена мгновенными сообщениями через Интернет в реальном времени такие, как ICQ, Jabber, AIM, MSN, Yahoo, Google Talk и соответствующие сетевые протоколы. Эти протоколы используют протоколы более низкого уровня, например TCP/IP, как транспорт для передачи данных. Передаваться могут текстовые сообщения, звуковые сигналы, картинки, видео. Такие программы могут применяться для организации телеконференций.

Для этого вида коммуникации необходима клиентская программа, так называемый мессенджер (англ. message – сообщение). Он отличается от электронной почты тем, что позволяет обмениваться сообщениями в реальном времени (instant – мгновенно). Большинство программ позволяет видеть, подключены ли в данный момент абоненты, занесенные в список контактов. В ранних версиях программ все, что печатал пользователь, тут же передавалось. Если он делал ошибку и исправлял ее, это тоже было видно. В таком режиме общение напоминало телефонный разговор. В современных программах сообщения появляются на мониторе собеседника только по окончанию редактирования и отправке.

Как правило, мессенджеры не работают самостоятельно, а подключаются к главному компьютеру данной сети обмена сообщениями, называемому сервером. Поэтому мессенджеры называют ещё клиентскими программами или клиентами (термин взят из понятий о клиент-серверных технологиях).

Широкому кругу пользователей известно некоторое количество популярных сетей обмена сообщениями, таких как ICQ, MSN, Yahoo!. Каждая из этих сетей разработана отдельной группой разработчиков, имеет отдельный сервер, отличается своими правилами и особенностями. Между различными сетями обычно нет никакой взаимосвязи. Таким образом, пользователь сети ICQ не может связаться с пользователем сети MSN. Однако, ничто не мешает быть одновременно пользователем нескольких сетей.

Для каждой из сетей есть свой мессенджер, разработанный той же командой разработчиков. Так, для пользования вышеуказанными сетями разработчиками предлагаются программы с одноименными названиями: ICQ, MSN Messenger, Yahoo! Messenger.

Большинство IM-сетей используют закрытые протоколы, поэтому альтернативные клиенты обычно обладают меньшим количеством функций, чем официальные. Также, в связи с изменениями протоколов на стороне сервера, альтернативные клиенты могут внезапно переставать работать.

Целью данной курсовой работы является разработка программного средства «Сетевой чат» для локальной сети не требующего собственного сервера.

**1 АНАЛИЗ И РАЗРАБОТКА ТРЕБОВАНИЙ   
К ПРОГРАММНОМУ СРЕДСТВУ**

**1.1 Анализ предметной области.**

Создание компьютерных сетей вызвано практической потребностью пользователей удаленных друг от друга компьютеров в одной и той же информации. Сети предоставляют пользователям возможность быстрого обмена информацией, совместной работы на сетевых устройствах, и даже одновременной обработки документов.

Развитие компьютерных сетей привело к появлению различных средств общения, так называемых чатов. Обычно под словом «чат» подразумевается обмен текстовыми сообщениями, реже встречаются голосовой чат, видеочат.  
Разрабатывались не только определенные программные средства, но так же и специализированные протоколы передачи сообщений. Одним из самых первых и популярных стал разработанный в 1988 году протокол, названный Internet Relay Chat (IRC), что примерно можно перевести как ретранслируемый интернет-разговор. Впоследствии появились многие другие.

Существующие программные средства для передачи текстовых сообщений можно условно разделить на 2 группы:

- приложения, сочетающие в себе функциональные возможности, как сервера, так и клиента для приема передачи сообщений. Данные программные средства обычно используются в локальных сетях.

- приложения, выполняющие только роль клиента при передаче сообщений. Сервер чаще всего является удаленным и выполняет функции транзита переданной информации от одного клиента к другому. Данные программы-клиенты называются мессенджерами.

Разрабатываемое программное средство «Сетевой чат» относится к первой группе сетевых приложений, т.е. сочетать в себе функции, как сервера для приема сообщений, так и клиента для их отправки. Таким образом, не будет необходимости устанавливать в сети дополнительное серверное приложение.

Чаты в домашних локальных сетях чаще всего используются для общения с друзьями и знакомыми. В корпоративных и офисных сетях чаты предназначены для:

- рассылки объявлений или заданий коллегам и подчиненным;

- напоминаний работникам о проведении совещаний;

- получения оповещений от администратора сети;

Архитектура сетевых чатов, как и архитектура других сетевых приложений, разделяют прикладную систему на службы, которые могут совместно и многократно использоваться множеством приложений. Во время выполнения сетевого приложения поток управления реализуется на одном или на нескольких хостах. Все компоненты такой системы обмениваются информацией, передавая друг другу данные и управление потоком по мере необходимости. Если использовать совместимые протоколы обмена информацией, то можно добиться взаимодействия отдельных компонентов, даже если базовые сети, операционные системы, аппаратные средства и языки программирования являются неоднородными.

**1.2 Разработка требований к программному средству**

**1.2.1 Общее описание**

Программное средство «Сетевой чат» представляет собой оконное приложение, предназначенное для отправки и получения текстовых сообщений между пользователями (компьютерами) в локальной сети.

Уровень владения персональным компьютером пользователей данного программного средства должен соответствовать уровню «Опытный пользователь».

Программное средство функционирует в среде под управлением операционной системы Windows.

Компьютеры, использующие сетевой чат должны быть подключены к локальной сети Ethernet. При подключении чат использует UDP и TCP/IP стеки сетевых протоколов.

**1.2.2 Функциональность системы**

Программное средство поддерживает функцию регистрации нового пользователя. При регистрации необходимо ввести имя пользователя и пароль. В дальнейшем введенные данные изменить будет невозможно.

Функцию входа в систему (логина) пользователей осуществляется каждый раз при запуске программы. При этом пользователю необходимо ввести имя и пароль. При наличии ошибок в вводимых данных пользователь будет оповещен об этом. Программа имеет возможность смены текущего пользователя во время работы. Только вошедший в системы пользователь имеет возможность принятия и отправки сообщений.

Отправка сообщений осуществляется путем ввода текста и нажатием на кнопку «Отправить».

Отправленные сообщения помещаются в окно отправленных и полученных сообщений.

Все вошедшие в систему клиенты в сети отображаются в списке пользователей. Список пользователей отображает только онлайн пользователей в данный момент времени. Для каждого пользователя в списке предусмотрено свое окно отправленных и полученных сообщений.

При выходе из системы или отключении пользователя он автоматически удаляется из списка пользователей у других клиентов.

Предусмотрена возможность настройки программного средства для работы в определенной сети (при наличии у пользователя нескольких сетевых интерфейсов) и/или с использованием различных портов для приема и отправки сообщений.

При приеме и передаче сообщений программное средство использует сетевой протокол UDP.

**1.2.3 Требования к внешним интерфейсам**

Главное окно приложения имеет возможность сворачивания на панель задач, и разворачивания на всю ширину рабочего стола. Пользовать может изменить размер главного окна в зависимости от своих предпочтений.

Главное окно разделяется на три области:

- список онлайн пользователей, представляет собой вертикальный список пользователей, отображает их имена, указанные при регистрации.

- окно для ввода новых сообщений, имеет функцию редактирования.

- окно для отображения принятых и отправленных сообщений, функция редактирования отключена.

Размер каждой области можно изменить относительно других.

В правом нижнем углу главного окна расположена кнопка отправки сообщений.

Главное меню расположено вверху окна под заголовком. Меню имеет следующую структуру:

- Вход в систему (Sign In);

Вход в систему (Sign In);

Выход из системы (Sign Out);

Закрытие приложения (Exit);

- Настройки (Settings);

- Информация о программе (About);

Программное средство предназначено для работы в операционных системах Windows XP, Windows Vista, Windows 7.

Требования к оборудованию (определены экспериментально, зависят от выбранной операционной системы):

- CPU AMD Duron 1000 Mhz;

- оперативная память 256 Mb;

- 50 Mb свободного места на жестком диске;

- сетевой адаптер Ethernet;

- подключение к локальной сети.

**2 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО СРЕДСТВА**

**2.1 Общие сведения**

Использование объектно-ориентированное проектирования на основе графических диаграмм языка UML позволяет:

- сокращение цикла разработки приложения «заказчик – программист – заказчик»;

- увеличение продуктивности работы программистов вследствие уменьшения ошибок и времени на отладку;

- улучшение потребительских качеств программ за счет ориентации на пользователей и бизнес;

- возможность вести большие проекты и группы проектов;

- возможность повторного использования уже созданного ПО за счет упора на разбор их архитектуры и компонентов;

- использование языка UML как универсального «мостика» между разработчиками из разных подразделений.

В качестве программного пакета для визуального объектно-ориентированного моделирования систем на основе классов и их взаимодействия выбран визуальный редактор Rational Rose.

**2.2 Разработка диаграммы вариантов использования**

Одним из основных преимуществ применения диаграммы вариантов использования – это предоставляет важной информации. Она показывает, какие функциональные возможности будут заложены в систему, кто конкретно будет с ней взаимодействовать. Изучение всего множества вариантов использования и действующих лиц, позволяет определить сферу применения системы.

При проектировании диаграммы вариантов использования для разрабатываемого программного средства были определены следующие варианты использования:

- отправка текстовых сообщений;

- получение текстовых сообщений;

- настройка сетевого подключения;

- выбор сетевого адаптера;

- регистрация;

- вход и выход из системы;

Диаграмма вариантов использования для программного средства «Сетевой чат» представлена на рисунке 2.1.



**Рисунок 2.1 –** **Диаграмма вариантов использования   
программного средства «Сетевой чат»**

**2.3 Разработка диаграммы взаимодействия**

На диаграмме взаимодействия отображается один из процессов обработки информации в варианте использования. Различают два типа диаграмм взаимодействия – диаграммы последовательности и кооперативные диаграммы. Диаграммы первого типа организованы по времени, они заостряют внимание на управлении. Кооперативные диаграммы отображают поток данных.

С помощью диаграмм взаимодействия мы можем определить классы, которые нужно создать, связи между ними, а также операции и ответственности каждого класса. Диаграммы последовательности полезны для того, что бы понять логическую последовательность событий в сценарии. Кооперативные диаграммы полезны в тех случаях, когда нужно оценить последствия сделанных изменений.

Диаграммы последовательностей используются для моделирование конкретных экземпляров классов, интерфейсов, компонентов и узлов, а также сообщений, которыми они обмениваются, – и все это в контексте сценария, иллюстрирующего данное поведение. Диаграммы взаимодействий могут существовать автономно и служить для визуализации, специфицирования, конструирования и документирования динамики конкретного сообщества объектов, а могут использоваться для моделирования отдельного потока управления в составе прецедента.

Диаграмма последовательности при отправке текстового сообщения пользователю в сети представлена на рисунке 2.2.



**Рисунок 2.2 –** **Диаграмма последовательности   
при отправке сообщения пользователю в сети**

На основании построенной в Rational Rose диаграммы последовательности можно автоматически получить кооперативную диаграмму. Кооперативная диаграмма при отправке сообщения пользователю в сети представлена на рисунке 2.3.



**Рисунок 2.3 –** **Кооперативная диаграмма   
при отправке сообщения пользователю в сети**