

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ
Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра дискретной математики и
информационных технологий

**Разработка сетевого приложения с помощью кроссплатформенного
инструментария Qt**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 5 курса 521 группы
направления 09.03.01 «Информатика и вычислительная техника»
факультета компьютерных наук и информационных технологий
Исаева Дмитрия Львовича

Научный руководитель

к. ф.-м.н., доцент

А.Д. Панферов

Зав. кафедрой

к. ф.-м.н., доцент

Л.Б. Тяпаев

Саратов 2016 г.

СОДЕРЖАНИЕ

| | |
|---|----|
| СОДЕРЖАНИЕ | 2 |
| ВВЕДЕНИЕ | 3 |
| 1 Реализация проекта | 4 |
| 1.1 Создание проекта | 4 |
| 1.2 Создание окна | 4 |
| 1.3 Реализация передачи сообщений..... | 5 |
| 1.4 Реализация передачи файла..... | 6 |
| 2 Демонстрация проекта | 7 |
| 2.1 Главное окно | 7 |
| 2.2 Окно настройки и окно отправки файла..... | 8 |
| 2.3 Тестирование работы сетевой части | 9 |
| ЗАКЛЮЧЕНИЕ..... | 11 |

ВВЕДЕНИЕ

Сегодня практически невозможно представить себе приложение без возможности использования сетевой среды и взаимодействия с модулями или данными через сетевые интерфейсы.

В данном реферате кратко рассматривается построение сетевого приложения и принцип его работы. Во второй части реферата изложена демонстрация работы приложения.

В дипломной работе была поставлена задача изучить возможности практической реализации сетевых обменов в разрабатываемых прикладных программах. В качестве простейшей модели такого сетевого обмена мною рассматривались текстовый чат и обмен файлами.

Для реализации программ с указанными модельными функциями был выбран язык программирования C++. Поскольку сами процедуры организации сетевого обмена достаточно трудоёмки, было решено воспользоваться существующими библиотечными решениями.

В качестве универсального инструмента была выбрана хорошо зарекомендовавшая себя кроссплатформенная библиотека Qt. Она уменьшает объем работы программистам, пишущим на C++ приложения для Windows, Linux и других ОС. Базовые функции сетевого обмена были реализованы с помощью библиотеки QtNetwork. Графический интерфейс разработанных программ строился с помощью редактора форм QtDesigner. В качестве среды разработки использовался встроенный IDE QtCreator с компилятором MinGW 5.5.0 на языке C++.

1 Реализация проекта

1.1 Создание проекта

Создание любого приложения начинается с его проектировки. Для разработки приложения, создается проект в QtCreator. В Qt проект привязывается к файлу .pro. Файлы проекта содержат всю необходимую информацию для того, чтобы qmake собрал приложение, библиотеку или подключаемый модуль (plugin). Ресурсы, используемые приложением, определяются в основном с помощью последовательностей объявлений, но, кроме того, поддержка простой программной структуры позволяет вам описывать различные процессы компоновки для различных платформ и сред.

1.2 Создание окна

Для проектировки и создания окна использовался инструмент QtDesigner. Qt Designer - инструмент для проектирования и создания графических пользовательских интерфейсов (GUI) из компонентов Qt. Имеет возможности создавать и настраивать виджеты или диалоги. Общий вид инструмента QtDesigner отображен на рисунке 1.

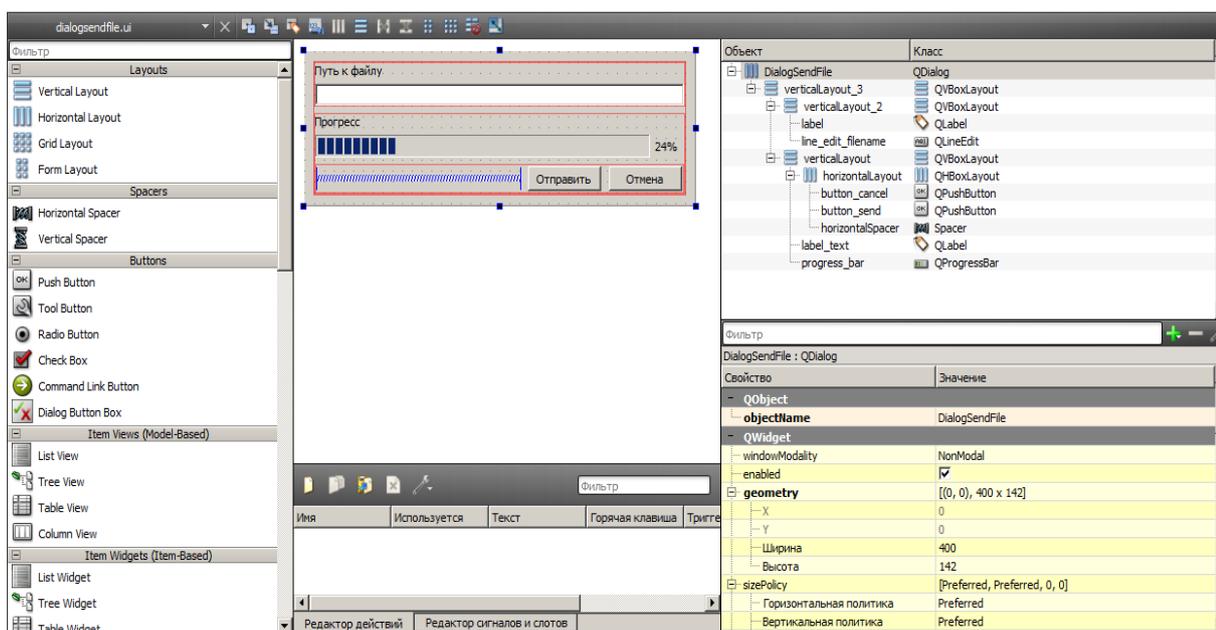


Рисунок 1 – Общий вид инструмента QtDesigner

Как видно из рисунка 1 слева находится список виджетов, которые можно

использовать в нашем приложении. Чтобы добавить виджет достаточно перетащить его мышью на наше «рабочее поле», в котором отображается будущее окно или диалог приложения. Справа находится список уже добавленных виджетов, которые, как видно из рисунка 1, группируются слоями (например QVBoxLayout). Ниже находится настройка виджета, в котором можно настроить множество параметров виджета, такие как размер, позиция, шрифт и многое другое.

После того проектировки окна, каждый виджет программируется вручную. Для программирования взаимодействия виджетов между собой используются сигналы и слоты. Сигналы и слоты это специальная система взаимодействий между виджетами Qt. Сигнал вырабатывается когда происходит определенное событие. Слот это функция, которая вызывается в ответ на определенный сигнал. Виджеты Qt имеют много предопределенных сигналов и слотов, но мы всегда можем сделать дочерний класс и добавить наши сигналы и слоты в нем.

1.3 Реализация передачи сообщений

В нашем приложении чат является общим для всех пользователей. Конкретнее говоря, мы принимаем все сообщения пользователей и показываем их в QTextEdit виджете, т.е. окне чата.

Чтобы реализовать такой чат, удобнее всего использовать UDP – протокол, нежели TCP, так как UDP работает быстрее и нам не требуется надежная доставка пакета.

При отправлении сообщения создается дейтаграмма UDP, в которой содержится тип сообщения (в данном случае «простое сообщение»), имя пользователя, отправившего сообщения, и само сообщение. Дейтаграмма отправляется по широковещательному адресу (255.255.255.255) по конкретному порту, установленному в настройках приложения. Пользователь, получивший дейтаграмму, обратно «разбирает» ее на части – тип сообщения, имя отправителя и сообщение. Когда видно что тип сообщения является «обычное

сообщение», то выводится в окно чата имя пользователя и сообщение.

В нашем приложении существуют разные типы сообщений. Для отображения списка пользователей сети, используется тип сообщения «вопрос». Пользователь получивший такое сообщение отправляет обратно сообщение «ответ». Соответственно тот кто получил «ответ», отображает в списке того кто отправил «ответ».

Также в приложении есть тип сообщения «оповещение», которые отправляется при включении приложения и выключении, тем самым уведомляя других участников чата о статусе.

1.4 Реализация передачи файла

Для реализации передачи файла используется ТСР-протокол, так как надежность в этом деле главное. Он обеспечивает гарантированную доставку с установлением соединений в виде байтовых потоков. Для передачи данных ТСР реализует модель клиент-сервер.

При передаче файла, пользователь, отправляющий файл, становится сервером и отправляет дейтаграмму пользователю, которому собирается передавать файл. Дейтаграмма сообщает получавшему ип-адрес отправителя, по которому получающий подключается к серверу для подключения файла.

Так как файл может оказаться размером больше чем это позволяет сокет, то файл передается циклично частями.

2 Демонстрация проекта

2.1 Главное окно

Главное окно представляется двумя полями для текста, одно из которых (верхнее) служит для чтения чата, второе (нижнее) для ввода текста в чат. Слева отображается список доступных пользователей. Также ниже списка пользователей имеется кнопка «Передать файл», которая служит для передачи файла.

Для отправки сообщение в чат используется нижнее поле для ввода. Чтобы отправить сообщение, можно нажать клавишу «Enter» либо нажать на кнопку «Отправить» в самом приложении. Кнопки слева от поля ввода являются вспомогательными кнопками. Кнопка «Копировать» копирует набираемый текст в буфер обмена, кнопка «Очистить» очищает поле ввода. На рисунке 2 показано главное окно приложения.

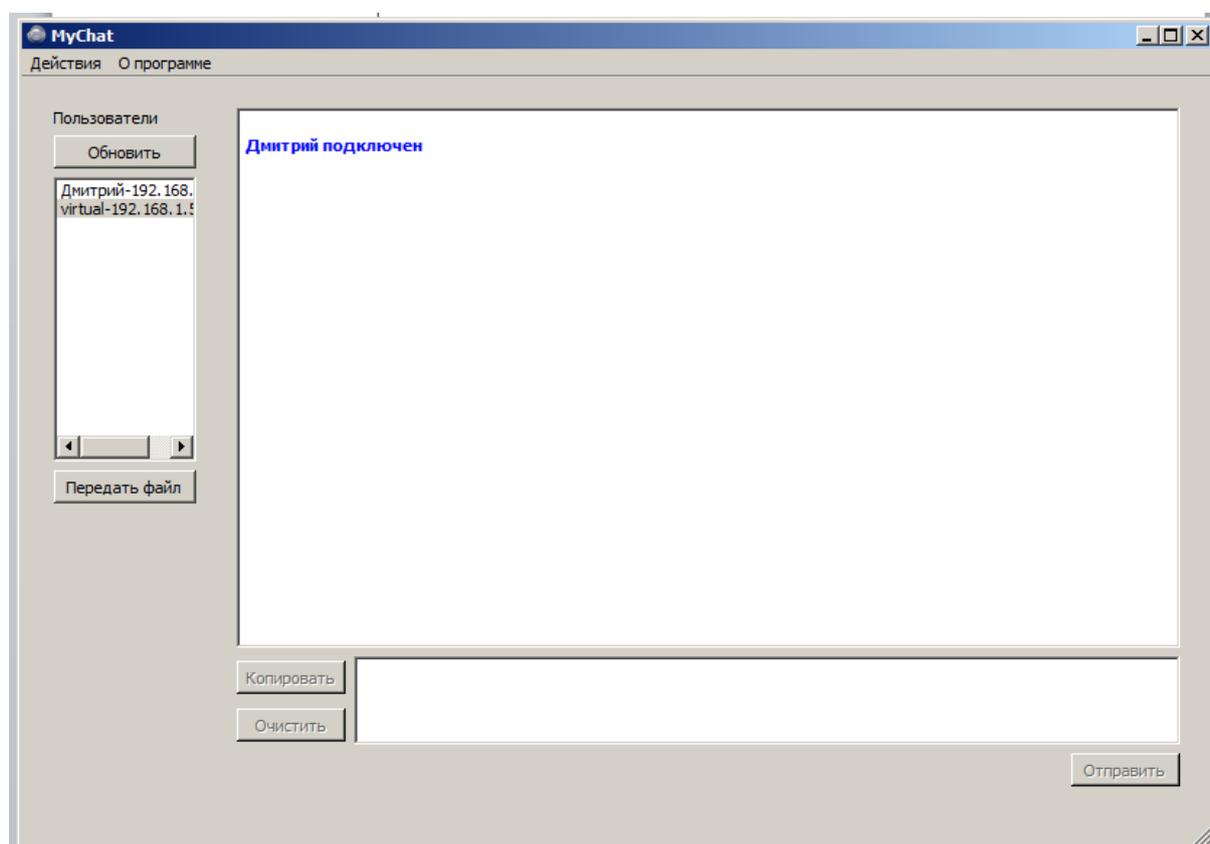


Рисунок 2 – Главное окно

2.2 Окно настройки и окно отправки файла

В окне настройки можно изменить параметры приложения, такие как IP-адрес, имя пользователя и порты. Окно представляет из себя набор из линий редактирования текста и кнопок «Отмена» и «Ок». После нажатия клавиши «Ок» параметры считываются с линий редактирования и заносятся в файл настройки приложения(config.ini) и окно закрывается. Общий вид окна настройки показан на рисунке 3.

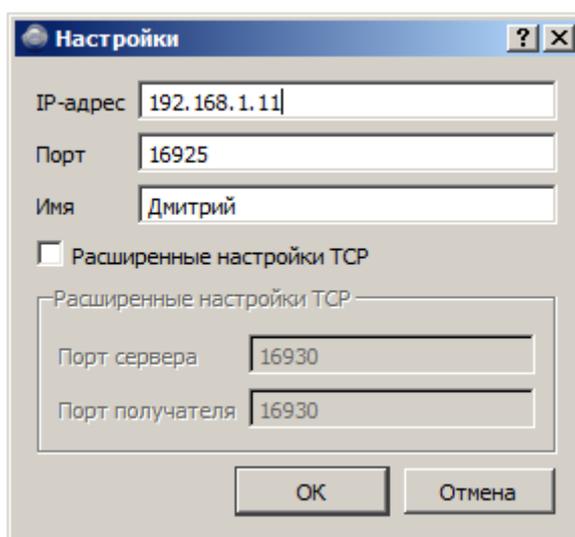


Рисунок 3 – Окно настройки

Чтобы открыть окно передачи файла, необходимо выделить пользователя в списке, и тогда активируется кнопка «Передать файла». После нажатия кнопки откроется окно передачи файла.

Окно передачи файла представляет собой линию редактирования и прогресс-бара. В линию редактирования необходимо ввести полное имя передаваемого файла. Если файл, введённый в линии редактирования, существует то приложение начнет передачу файла и активируется прогресс-бар, который будет показывать состояние передачи файла. В противном случае, приложение сообщит о несуществующем файле. Общий вид окна показан на рисунке 4.

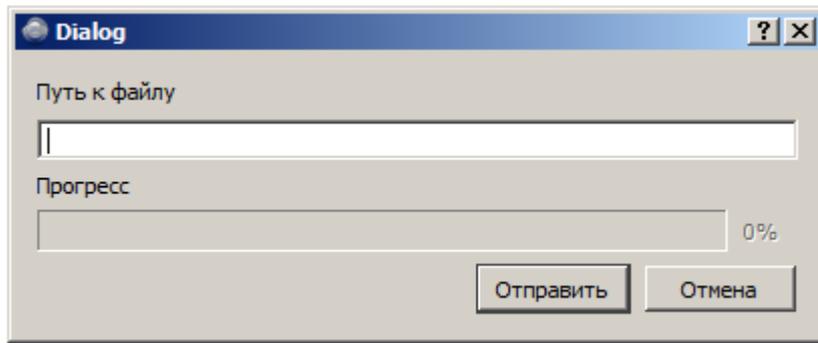


Рисунок 4 – Окно передачи файла

2.3 Тестирование работы сетевой части

Тестирование работы сетевой части проводилось с помощью программы Virtual Box. С помощью Virtual Box мною была создана виртуальная машина с виртуальным сетевым адаптером. На рисунке 5 показан чат с сообщениями обеих сторон.

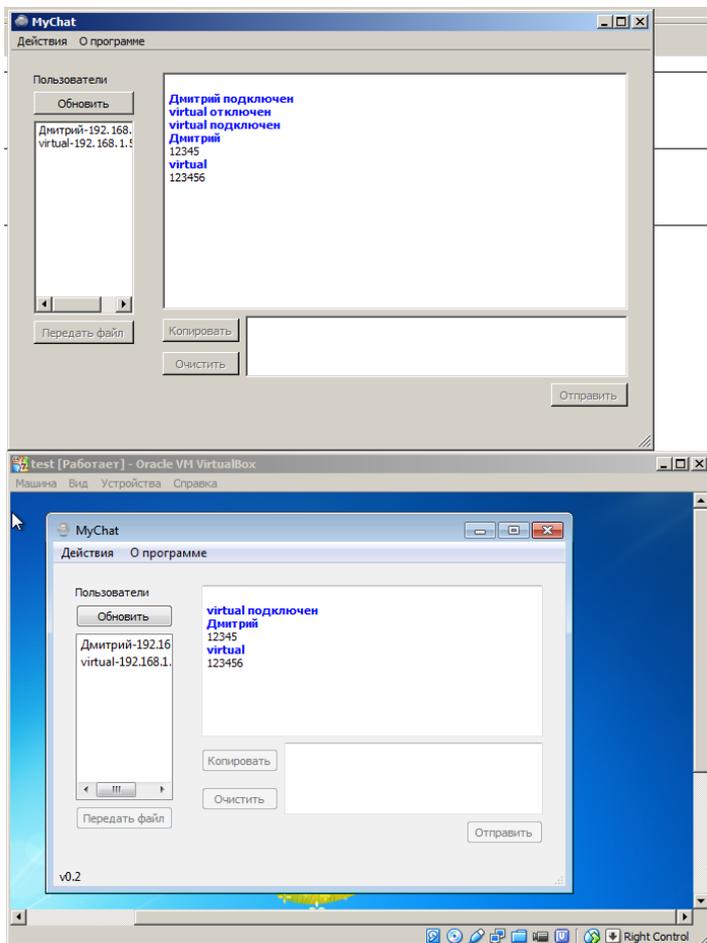


Рисунок 5 – Чат с виртуальной машиной

На рисунке 6 показана передача файла.

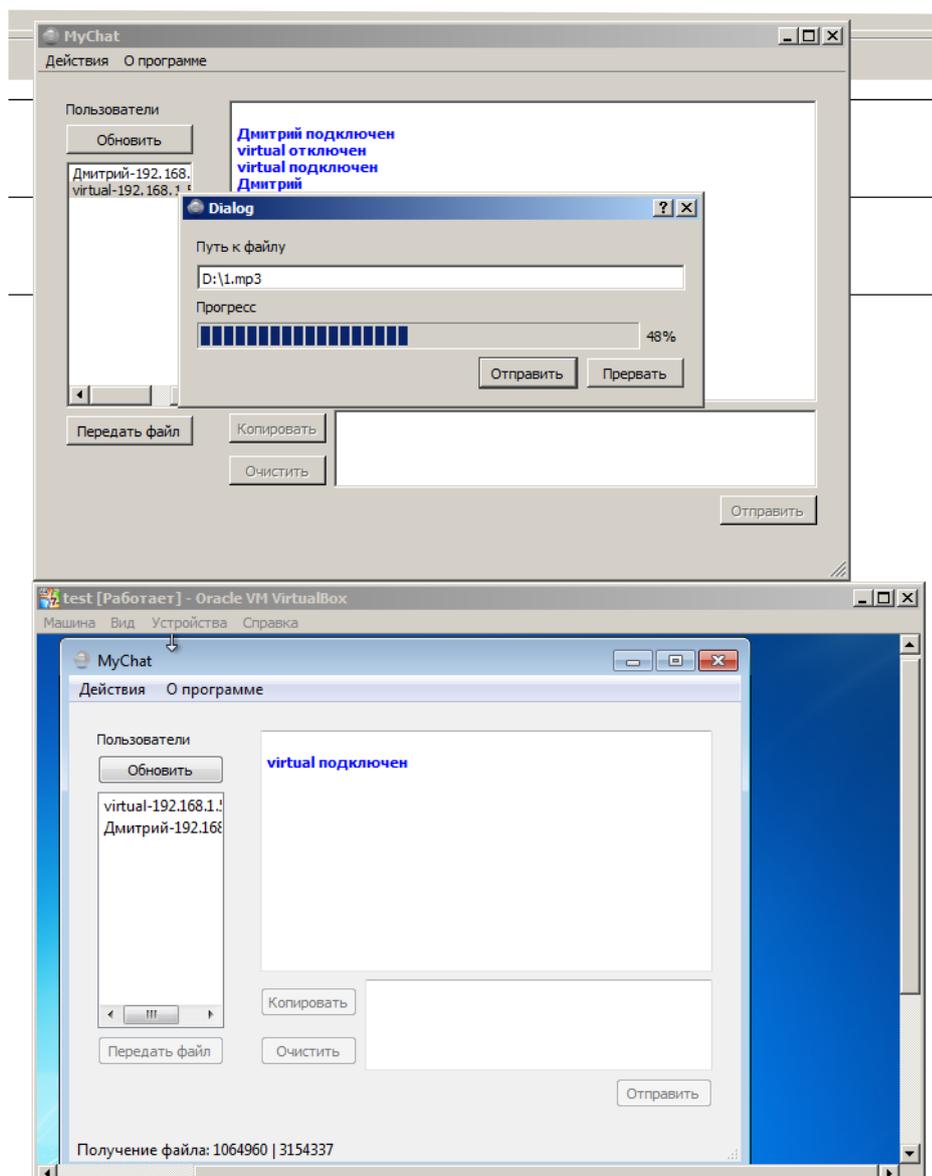


Рисунок 6 – Передача файла

ЗАКЛЮЧЕНИЕ

В ходе работы был представлен инструмент QtDesigner, с помощью которого были спроектированы окна приложения. Также был показан принцип построения сетевой части приложения. Для реализации сетевой части программы использовалась библиотека QtNetwork, в частности с классами работающими с TCP и UDP протоколами, которые в последующем были использованы в построении приложения.

Результатом данной работы является приложение чат-клиент. Данное приложение имеет возможность обмениваться пользователям сообщениями в окне чата, и также передавать файлы.