**МиНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение

высшего образования

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ**

**УНИВЕРСИТЕТ им. В.Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

**Институт** *Энергетики, информационных технологий и управляющих систем*

**Кафедра** *Программного обеспечения вычислительной техники и автоматизированных систем*

**Шифр, название направления, специальности**

09.03.01 Информатика и вычислительная техника

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

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

(образовательная программа)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

на тему:

Мобильное приложение на платформе Android для получения удаленного доступа к компьютеру

**Дипломник** *Плотникова А. О.*

**Зав.кафедрой** *Поляков В. М.*

**Руководитель** *Федотов Е. А.*

**К защите допустить**

**Зав. кафедрой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/**Поляков В.М**/**

**«\_\_\_\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_г.**

**Белгород 20\_\_\_\_\_г.**

**МиНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение

высшего образования

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ**

**УНИВЕРСИТЕТ им. В.Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

**Институт** Энергетики, информационных технологий и управляющих систем

**Кафедра**Программного обеспечения вычислительной техники и автоматизированных систем

**Шифр, название направления, специальности**

09.03.01 Информатика и вычислительная техника

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

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

(образовательная программа)

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

|  |  |
| --- | --- |
|  | Утверждаю:  Зав. кафедрой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «17»\_\_\_\_\_\_\_\_июня\_\_\_\_\_\_2016г. |

Задание

на выпускную квалификационную работу студента (ки)

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*\_\_\_Плотниковой Алёны Олеговны \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1.Вид выпускной квалификационной работы\_\_\_\_\_\_\_\_\_бакалаврская работа\_\_\_\_\_\_\_\_\_\_\_\_

(дипломный проект (работа), бакалаврская работа, магистерская диссертация)

2. Тема работы\_\_\_\_\_\_\_\_Мобильное приложение на платформе Android для получения удаленного доступа к компьютеру\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_утверждено приказом по университету от «\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2017г. №\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. Срок сдачи студентом законченной работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. Исходные данные\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

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

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

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

5.Содержание выпускной квалификационной работы (перечень подлежащих разработке

|  |  |
| --- | --- |
| разделов) | *1. Описание предметной области, анализ и выбор методов решения задачи* |
| *2. Проектирование программного обеспечения* | |
| *3. Программная реализация* | |
|  | |

6.Перечень графического материала\_\_\_\_ Презентация 11 слайдов, рисунков 7, таблиц 3 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

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

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

Консультанты по работе с указанием относящихся к ним разделов

|  |  |  |  |
| --- | --- | --- | --- |
| Раздел | Консультант | Задание выдал  (подпись, дата) | Задание принял  (подпись, дата) |

Дата выдачи задания «\_\_» \_\_\_\_\_\_\_\_ 2017г.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Федотов Е. А.

(подпись руководителя)

Задание принял к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Плотникова. А. О. (подпись выпускника)

**КАЛЕНДАРНЫЙ ПЛАН**

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Наименование этапов работы | Срок выполнения  этапов работы | Примечание |
| 1 | Описание предметной области, анализ и выбор методов решения задачи |  |  |
| 2 | Проектирование программного обеспечения |  |  |
| 3 | Проектирование и реализация алгоритма |  |  |
| 4 | Программная реализация |  |  |
| 5 | Оформление записки, презентация |  |  |

Дипломник\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Плотникова А. О.

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Федотов Е. А.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ……………………………………………………………………….5

**1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ, АНАЛИЗ И ВЫБОР**

**МЕТОДОВ РЕШЕНИЯ ЗАДАЧИ**……….…………………………………….7

**1.1. Постановка задачи**………………………………………………………...7

**1.2. Обзор предметной области**……………………………………………….8

**1.3. Обоснование актуальности разрабатываемого клиент-серверного приложения**………………………………………………………………………9

**1.4. Обзор существующих аналогов разрабатываемого приложения**.....10

**1.4.1. TeamViewer**…………………………………………………………….11

**1.4.2. Удаленный рабочий стол Chrome**…………………………………..12

**1.4.3. LogMeIn**………………………………………………………………...13

**1.4.4. Microsoft Remote Desktop**…………………………………………….14

**1.5. Анализ требований к программному обеспечению**………………….16

**1.6. Требования к разработке программного обеспечения**……………...17

**1.6.1. Основания для разработки**…………………………………………..17

**1.6.2. Назначение разработки**………………………………………....……18

**1.6.3. Требования к программе или программному изделию**………….18

**1.6.4. Требования к программной документации**………………………..19

**1.6.5. Порядок контроля и приемки**……………………………………….19

**1.6.6. Стадии и этапы разработки**………………………………………….19

**1.7. Обзор и выбор языков программирования**…………………………...20

**1.8. Обзор и выбор сетевых протоколов**……………………………………23

**1.9. Выбор сред разработки**…………………………………………………..26

**2. ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**………...29

**2.1. Общее проектирование**………………………………………………….29

**2.2. Схема и принцип работы приложения**………………………………...31

**2.3. Проектирование пользовательского интерфейса**……………………33

**2.4. Проектирование механизма передачи данных**……………………….36

ВВЕДЕНИЕ

То, что человечество перешло в век информационных технологий, уже давно ни для кого не является секретом. В мире существует огромное количество различных устройств, предназначенных для упрощения жизни человека и избавления его от огромного объема работы, которую необходимо было выполнять вручную еще сто, или даже пятьдесят, лет назад.

Ежедневно миллионы инженеров и разработчиков трудятся над созданием лучших устройств, предназначенных для услужения человеку. Каждый день контроллеры, планшеты, аудиосистемы, компоненты компьютеров, смартфоны и различные программные продукты тестируются, проверяются на всевозможные дефекты, анонсируются и, в конце концов, выпускаются.

В наши дни самыми используемыми, неоспоримо, являются мобильные приспособления, такие как смартфоны, планшеты и портативные КПК, и компьютеры. Не существует такого предприятия, которое работало бы без обращения за помощью к этим устройствам.

Однако ни одно техническое приспособление не застраховано от ошибок. Любая проблема с вычислительным устройством, возникшая в процессе работы предприятия, может привести к весьма серьезным последствиям, поэтому крайне необходим механизм, предназначенный для профилактики, обнаружения и исправления возникающих ошибок в кратчайшие сроки.

Таким механизмом могут выступать средства удаленного доступа, предоставляющие возможность быстрой реакции на все возникающие вопросы без необходимости траты времени на путь квалифицированного специалиста до проблемного устройства.

Существует огромный выбор программных продуктов, предоставляющих удаленный доступ с компьютера к компьютеру или с компьютера к мобильному устройству, но приложений, обеспечивающих пользователя возможностью получения удаленного доступа с мобильного устройства к компьютеру, не так уж и много. Все представленные на рынке программы обладают либо недостаточным для комфортной работы функционалом, либо ценником с неприличного вида цифрой, что делает вопрос о разработке общедоступного полнофункционального приложения для управления компьютером с мобильного устройства весьма актуальным.

1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ, АНАЛИЗ И ВЫБОР МЕТОДОВ РЕШЕНИЯ ЗАДАЧИ

**1.1. Постановка задачи**

Целью дипломного проекта является разработка клиент-серверного приложения для удаленного доступа к компьютеру с мобильного устройства на платформе Android.

Разрабатываемое программное обеспечение должно обладать следующим функционалом:

1. Адаптивный дизайн – правильное отображение на устройствах с различным размером экрана и при любой ориентации экрана.
2. Форма авторизации подключения. Данный пункт подразумевает запрос серверным (на компьютере) приложением логина и пароля для подтверждения того, что на данный момент подключиться пытается именно тот пользователь, доступ которому выдать и планировалось.
3. Формы взаимодействия пользователей с удаленным устройством как на управляемом компьютере, так и на мобильном девайсе.
4. Демонстрация экрана удаленного компьютера, к которому производится подключение с мобильного устройства.
5. Управление курсором мыши и нажатиями на кнопки клавиатуры.
6. Пересылка файлов. Возможность пересылки файлов между устройствами во время сеанса удаленного доступа зачастую позволяет намного быстрее приблизиться к решению поставленной задачи.
7. Обратная связь. Может быть представлена чатом или звонками.
8. Отправка звука - возможность проигрывания звуковых сигналов с удаленного компьютера.
9. Удобный и простой в освоении пользовательский интерфейс.

Разработка программного обеспечения проходит в несколько этапов:

1. Определение цели и актуальности разработки нового приложения;
2. Изучение существующих аналогов проектируемого приложения, изучение их достоинств и недостатков для их учета в проектируемой системе;
3. Формирование списка функций, выполняемых приложением;
4. Составление технического задания, разработка моделей предметной области, разработка структуры команд для обеспечения связи между устройствами, реализация функционала.

**1.2. Обзор предметной области**

С развитием информационных технологий в современном обществе все чаще возникает вопрос о наличии возможности управления удаленным компьютером через интернет с другого устройства. Системы удаленного администрирования широко используются во всех крупных компаниях для мониторинга, профилактики и исправления, возникающих в процессе работы проблем. Использование средств удаленного доступа привносит в работу любого предприятия неоспоримые плюсы: мобильность, повышенная скорость реакции на большую часть проблем, связанных с администрированием сетей, ускорение и упрощение получения доступа к конечному компьютеру и др.

Удаленный доступ представляет собой функцию, позволяющую пользователю подключаться к компьютеру через Интернет с помощью другого устройства. Опция открывает возможность пользователю использовать свой компьютер удаленно, и обеспечивает следующие дополнительные возможности:

1. Доступ к файлам. Можно производить отправку файлов, специфические размеры или характеристики которых не позволяют отправить их по электронной почте.
2. Передача файлов. Это возможность копирования файлов и папок на текущее устройство с удаленной машины и наоборот.
3. Гостевой доступ. Представляет собой безопасную опцию, позволяющую получать доступ для решения какой-то конкретной задачи на удаленном компьютере, предоставляя доступ к рабочему столу и возможность контролировать клавиатуру и мышь.
4. Сопровождение программ, а также организация работы на предприятии.[1]

Однако, при обширном количестве программ, предоставляющих возможность удаленного управления компьютером с другого компьютера, качественных программных продуктов, позволяющих выполнять те же действия с использованием смартфона на рынке достаточно мало.

Получение удаленного доступа к компьютеру с мобильного устройства предоставляет администратору ряд неоспоримых преимуществ, таких как мобильность – возможность доступа к удаленному компьютеру из любой точки, не обязательно с рабочего места; повышенная скорость реакции на возникающие проблемы.

Но не следует забывать и о необходимости реализации дополнительных возможностей, делающих работу таких программ более комфортной для пользователей: создание обратной связи (чат, звонки), простого и понятного интерфейса, написание руководств пользователю и т. д.

**1.3. Обоснование актуальности разрабатываемого клиент-серверного приложения**

Клиент-серверное приложение для удаленного доступа к компьютеру с Android-устройства актуально и полезно по ряду причин:

1. *Управление удаленным устройством на расстоянии*. Использование приложений для удаленного доступа позволяет пользователю получать данные и производить операции с удаленным компьютером без необходимости нахождения с устройством в одном помещении.
2. *Повышенная мобильность*. Приложение для устройств на платформе Android позволяет пользователю получить доступ к удаленному компьютеру абсолютно из любой точки при условии наличия связи и мобильного устройства.
3. *Снижение затрат и увеличение прибыли.* Предприятия, использующие приложения для удаленного доступа к компьютеру, получают неоспоримые преимущества: повышение скорости реакции и исправления возникающих ошибок в работе информационных систем; нет необходимости выезда специалистов в случае работы с обширной клиентской базой.
4. *Возможность использования общедоступного программного обеспечения с полным функционалом.* Рынок приложений для обеспечения удаленного доступа к компьютеру полон программных продуктов, имеющих только платные версии, либо неполные, ограниченные бесплатные версии для некоммерческого использования. Таким образом, бесплатное приложение с полным функционалом будет явно выделятся на фоне своих конкуррентов.

**1.4. Обзор существующих аналогов разрабатываемого приложения**

Существует не так много ПО для обеспечения доступа к удаленному компьютеру с мобильного устройства на платформе Android. Большая часть из этих программных продуктов является платными коммерческими системами, либо демонстрационными версиями для некоммерческого использования.

Для пользователей основными преимуществами при работе с такими приложениями являются:

1. Удобный, простой и интуитивно понятный пользовательский интерфейс.
2. Возможность подключения по локальной сети или через Интернет, в зависимости от поставленных задач.
3. Возможность безопасного подключения без риска предоставить доступ к конечному компьютеру посторонним лицам.
4. Обеспечение обратной связи в форме чата или звонков.
5. Низкая стоимость приобретения.

Все существующие в этой области приложения в большей или меньшей степени обладают выше указанными особенностями. Не стоит забывать о том, что разработчики такого ПО постоянно обновляют и совершенствуют его функционал для предоставления пользователю максимально комфортной версии. Рассмотрим наиболее популярные из таких приложений.

**1.4.1. TeamViewer**

TeamViewer является самым популярным приложением для удаленного доступа по сети Интернет. Является бесплатным для некоммерческого использования, однако, при необходимости использовать приложение в коммерческих целях, необходимо приобрести платную версию.

Программное обеспечение TeamViewer для коммерческого использования представлено в трех версиях: [2]

1. Business – стоимость лицензии составляет 29900р. Базовая версия программы. В ее функционал входит возможность подключения компьютера к компьютеру и мобильного устройства к компьютеру. Поддерживает только управление устройством.
2. Premium – стоимость лицензии составляет 59200р. Имеет все возможности, предоставляемые лицензией Business, а также управление не только устройствами, но и пользователями и подключение компьютера к мобильному устройству.
3. Corporate – стоимость лицензии составляет 99900р. Помимо всех возможностей, предоставляемых версией Premium, поддерживает составление отчетов о пользовательских подключениях к устройству, возможность подключения до трех пользователей одновременно. Также фирма, приобретающая эту лицензию, получает двенадцатимесячный договор на первоочередное обслуживание.

В стоимость каждой лицензии включены:

* Техническая поддержка (call-центр)
* Обновление ПО

Интерфейс программы TeamViewer на Рисунке 1.1

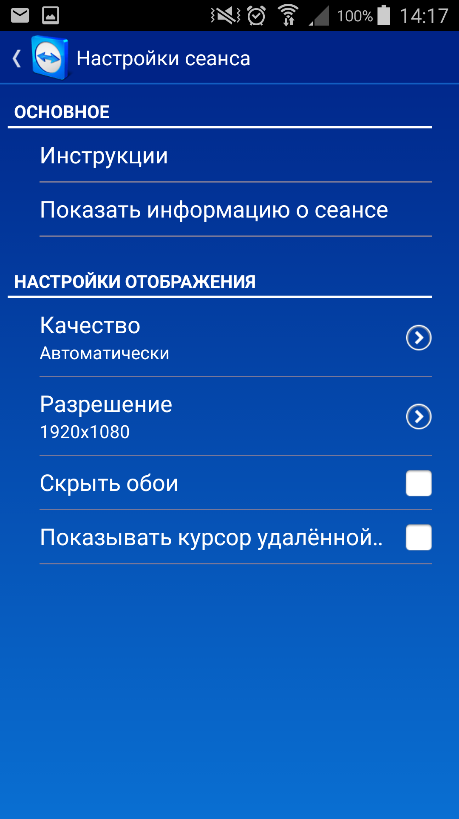
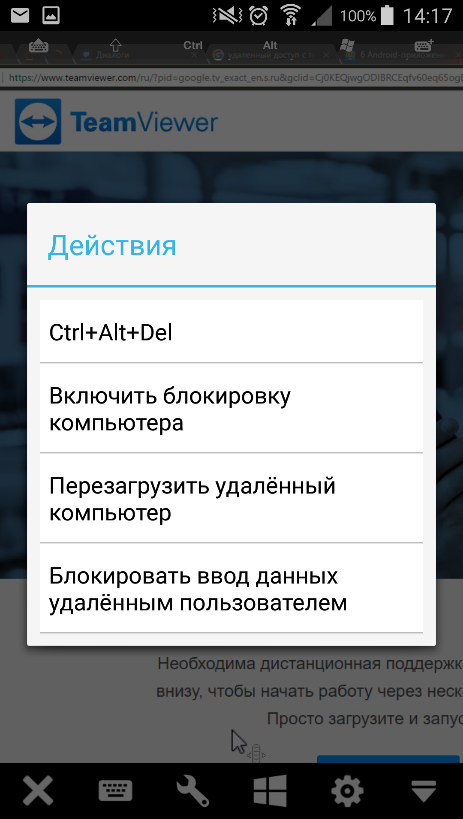
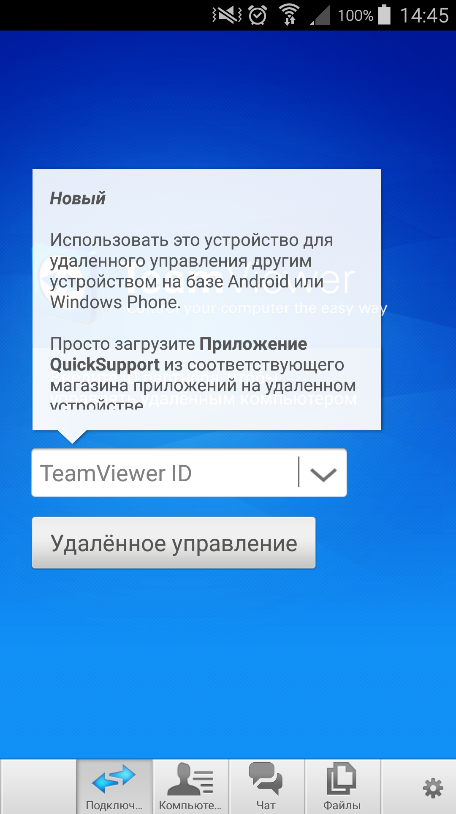


Рисунок 1.1 - TeamViewer. Интерфейс приложения

Одним из недостатков системы TeamViewer является необходимость установки VNC-сервера от производителей. Из-за такого способа подключения приложение может существенно загружать малопроизводительные устройства.

**1.4.2. Удаленный рабочий стол Chrome**

Удаленный рабочий стол Chrome (Рисунок 1.2) – программное обеспечение для предоставления доступа к удаленному рабочему столу на компьютере. Данное ПО является полностью бесплатным, но не поддерживает такие важные функции, как передача файлов или звука. Так же не имеет средств коммуникации с пользователем удаленного устройства. По словам разработчиков, приложение работает значительно быстрее всех своих аналогов, но, исходя из отзывов пользователей, качество картинки временами оставляет желать лучшего.

Приложение распознает удаленные устройства только в том случае, если авторизация в учетных записях обоих конечных устройств произведена с помощью одного и того же электронного адреса.

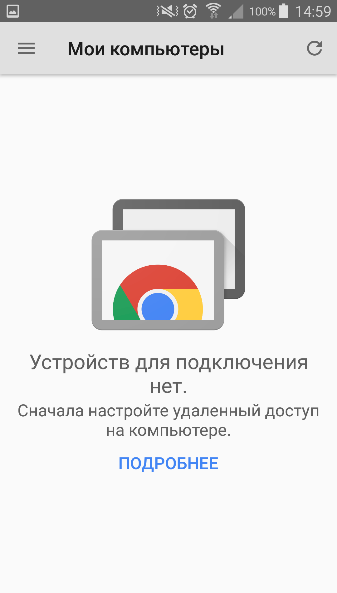
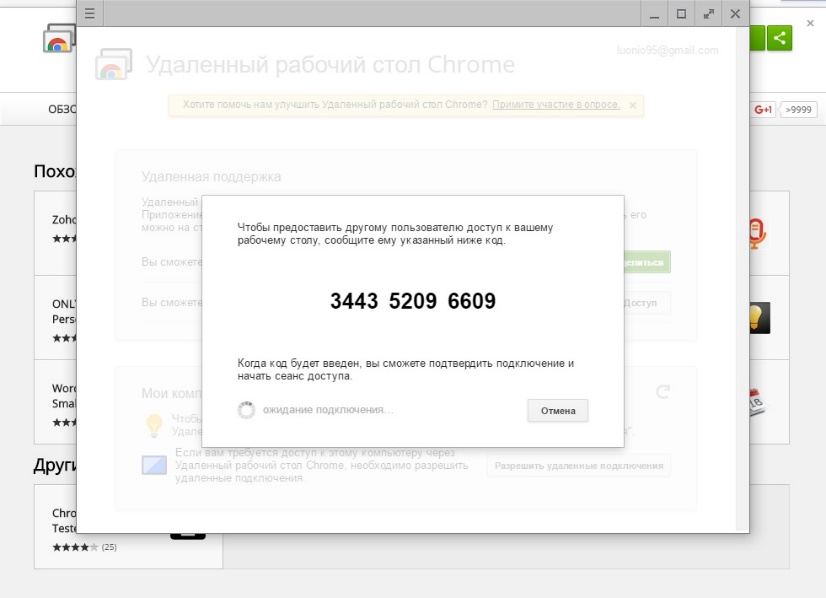


Рисунок 1.2 – Удаленный рабочий стол Chrome. Окна подключения

**1.4.3. LogMeIn**

LogMeIn (Рисунок 1.3) является мощным приложением с широким набором функций, позволяющих не только удаленно управлять курсором мыши и нажатиями клавиш компьютера, но и получать доступ к файловой системе. Данное программное обеспечение не является бесплатным, но обладает пробной версией на 14 дней.

Программное обеспечение LogMeIn имеет достаточно много платных версий, основные из которых описаны ниже: [3]

1. Версия для частных лиц стоимостью 249.99€ в год. Поддерживает удаленный доступ к двум компьютерам и предоставляет одну премиум-лицензию на LastPass (диспетчер паролей).
2. Версия для продвинутых пользователей стоимостью 599.99€ в год. Поддерживает удаленный доступ к пяти компьютерам и предоставляет три премиум лицензии на LastPass.
3. Версия для малых предприятий стоимостью 1099.99€ в год. Поддерживает удаленный доступ к десяти компьютерам и предоставляет пять премиум лицензий на LastPass.

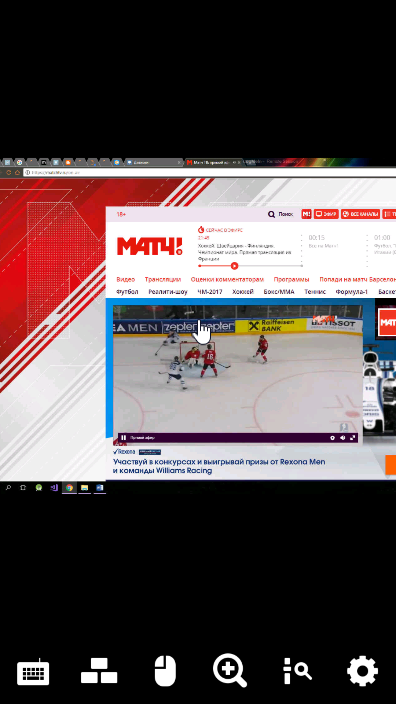
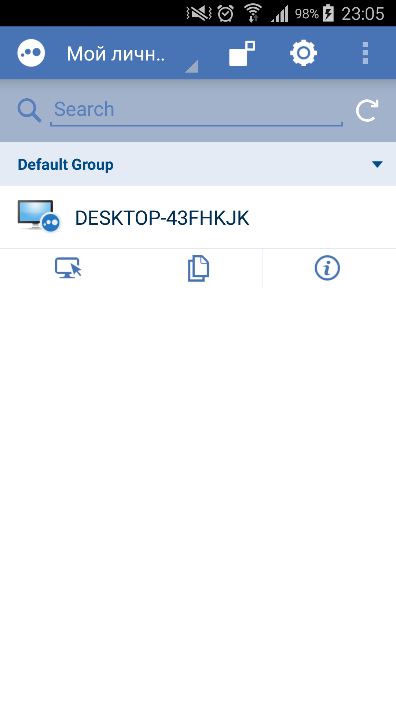
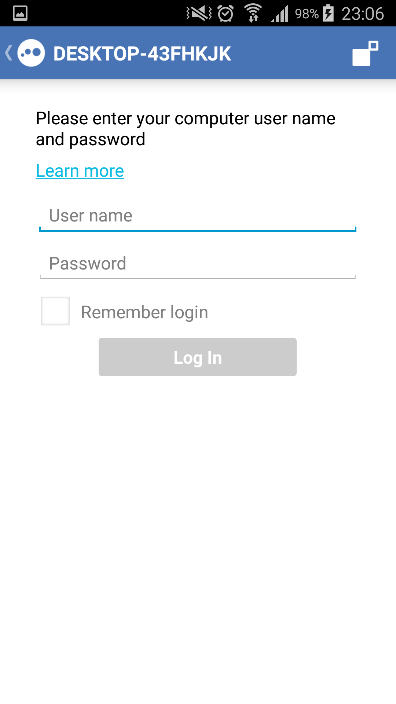


Рисунок 1.3 – LogMeIn. Интерфейс Android-приложения

Хоть программное обеспечение LogMeIn и подкупает своей обильностью разнообразных функций, однако оно имеет и существенные минусы: при таком обилии предоставляемых возможностей приложение не всегда работает плавно и быстро. Зачастую картинка зависает или меняется с частотой, примерно равной 0.5 – 1с, что естественно, создает дискомфорт при работе с удаленным компьютером.

**1.4.4. Microsoft Remote Desktop**

Microsoft Remote Desktop использует стандартный протокол Windows для обеспечения удаленного доступа, однако версии Windows, более старые, чем Windows Vista, этот протокол не поддерживают. Для использования этого программного обеспечения не нужно устанавливать на компьютер никаких сторонних программ, поэтому в некоторых случаях данное решение может стать предпочтительным.

Целью приложения Microsoft Remote Desktop является предоставление стандартного решения для удаленного доступа. Кроме демонстрации экрана и возможности управления курсором мыши и клавиатурой удаленного компьютера данное программное обеспечение не предоставляет никаких дополнительных функций, таких как передача файлов или организация текстовой или голосовой связи между пользователями.

Приложение под Android имеет приятный взгляду, достаточно простой интерфейс (Рисунок 1.4), но не поддерживает русский язык.

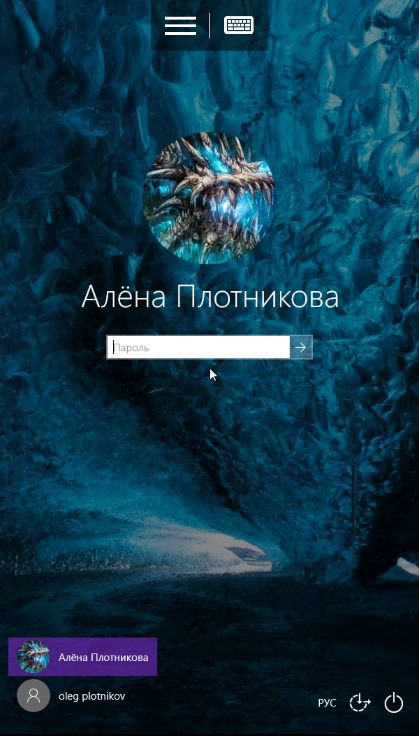
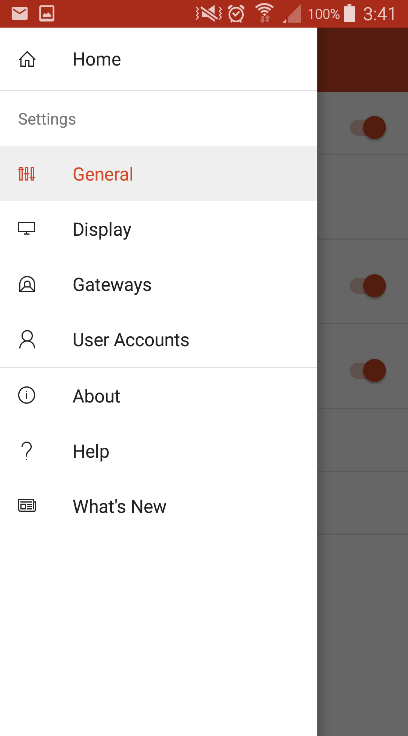
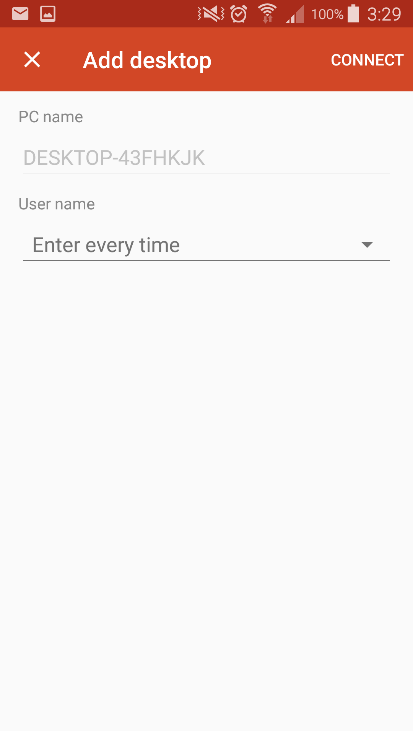


Рисунок 1.4 – Microsoft Remote Desktop. Интерфейс Android-приложения

Из существенных недостатков можно выделить тот факт, что при установке соединения программа выводит подключаемого на экран входа в учетную запись пользователя на удаленном компьютере. Таким образом, для решения поставленных задач будет необходимо делиться с администратором конфиденциальными данными (логином и паролем для входа в учетную запись), либо специально создавать гостевую учетную запись.

Но на этом данный недостаток еще не является исчерпанным. Программа не позволяет одновременно работать с удаленным компьютером и его пользователю, и администратору, подключающемуся при помощи Android-приложения. Как только в приложении на мобильном устройстве подтверждается подлинность введенного пароля, пользователь удаленного компьютера переводится на экран входа для ожидания завершения работы администратора.

Явным недостатком является и то, что для корректной работы данного программного обеспечения в сети Интернет пользователю необходимо вручную производить сетевые операции, связанные с настройкой портов, ip-адресов и пр.

Программное обеспечение Microsoft Remote Desktop может подойти для разового выполнения какой-либо несложной задачи, но из-за столь явных недостатков совершенно не подходит для коммерческого использования.

**1.5. Анализ требований к программному обеспечению**

Исходя из рассмотренных аналогов приложений можно провести анализ требований к дипломному проекту.

Любое программное обеспечение для обеспечения удаленного доступа к компьютеру должно обладать следующим минимальным функционалом:

1. Демонстрация экрана. Возможность пересылки точного снимка экрана в конкретный момент времени;
2. Управление курсором мыши и нажатиями клавиш на клавиатуре. Управление всеми передвижениями курсора мыши не обязательно должно осуществляться на конечном компьютере. Главное – реализация нажатий на кнопки мыши и перетаскиваний;
3. Наличие обратной связи, такой как чат или голосовая связь, для возможности обговорить суть возникшей проблемы «не отходя от кассы»;
4. Простота и удобство пользовательского интерфейса;
5. Обеспечение безопасности при подключении. Ни один посторонний пользователь не должен иметь доступ к удаленному компьютеру на момент подключения.

В таблице 1.1 приведён анализ рассмотренных продуктов, основные возможности, выявлены их преимущества и недостатки.

Таблица 1.1 – Сравнение аналогов

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ПО/  Особенность | TeamViewer | Chrome Remote Desktop | LogMeIn | Microsoft Remote Desktop | Разрабатываемое ПО |
| Наличие бесплатной/демо-версии | да | Полная версия | да (14 дней) | Полная версия | Полная версия |
| Простота и удобство пользовательского интерфейса | да | да | нет | нет | да |
| Демонстрация экрана | да | да | да | да | да |
| Безопасность при подключении | да | да | да | да | да |
| Пересылка файлов | да | нет | да | нет | да |
| Обратная связь | да (только чат) | нет | нет | нет | да |
| Русскоязычная локализация | да | да | только для ПК | нет | да |

**1.6. Требования к разработке программного обеспечения**

**1.6.1. Основания для разработки**

Дипломный проект на тему «Мобильное приложение на платформе Android для получения удаленного доступа к компьютеру».

**1.6.2. Назначение разработки**

Функциональное и эксплуатационное назначение программы – обеспечение удаленного доступа к компьютеру.

Разрабатываемый программный продукт предназначен для обеспечения мониторинга, профилактики и устранения ошибок, возникающих на удаленном компьютере, а также настройки и управления параметрами без необходимости получения прямого доступа к компьютеру.

**1.6.3. Требования к программе или программному изделию**

**Требования к функциональным характеристикам**

Программа должна предоставлять возможность удаленного подключения к любому компьютеру с реальным IP-адресом. Входными данными являются идентификатор (имя) пользователя, IP-адрес устройства, к которому будет производиться подключение, и пароль, генерируемый программой на удаленном компьютере.

**Требования к надежности**

Права на удаленный доступ предоставляются только после ввода правильного пароля, генерируемого серверным приложением на удаленном компьютере. До ввода пароля пользователь не может совершать каких-либо действий с удаленным устройством. Такие ограничения обеспечивают невозможность подключения к компьютеру посторонних лиц, которым не планировалось предоставлять доступ, что гарантирует сохранность данных и невмешательство посторонних в рабочий процесс.

**Условия эксплуатации**

Эксплуатация программного обеспечения не требует от пользователя особых знаний предметной области. Для проведения сеанса удаленного доступа необходим один администратор, который будет производить управление удаленным компьютером, и, не обязательно, пользователь или оператор удаленного компьютера, поддерживающий связь с администратором средствами программного продукта.

**Требования к составу и параметрам технических средств**

Для корректной работы программного обеспечения необходимо подключение к локальной сети или доступ к сети Интернет.

**Требования к информационной и программной совместимости**

Приложение разрабатывается с использованием платформ .NET и Android. На компьютере, к которому будет производиться подключение, должна быть установлена серверная часть приложения и .NET Framework версии 4.5 или выше. На мобильном устройстве должна быть установлена операционная система Android версии 4.1 или выше и клиентская часть приложения.

**Специальные требования**

Специальных требований не предусмотрено.

**1.6.4. Требования к программной документации**

В состав программной документации входят следующие документы:

- Техническое задание (ГОСТ 19201-78);

- Подробная инструкция, с помощью которой пользователь сможет без особого труда работать сданным приложением.

**1.6.5. Порядок контроля и приемки**

Общие требования к приемке работ – должны выполняться функции, согласованные при составлении задачи.

**1.6.6. Стадии и этапы разработки**

Стадии разработки приведены в таблице 1.2.

Таблица 1.2 – Стадии разработки ПО

|  |  |  |  |
| --- | --- | --- | --- |
| **Стадия НИР** | **Содержание работ** | **Трудоемкость** | |
| **дни** | **%** |
| 1.Техническое задание | Подбор и изучение литературы, анализ составления вопроса, согласование с руководителем и утверждение технического задания и плана работ. Постановка задачи. | 3 | 7,69 |

Таблица 1.2 – Стадии разработки ПО. Продолжение

|  |  |  |  |
| --- | --- | --- | --- |
| 2.Эскизный проект | Теоретическая разработка темы. Предварительная разработка структуры входных и выходных данных. Разработка общего описания алгоритма решения задачи. | 4 | 10,25 |
| 3.Технический проект | Проектирование. Определение основных блоков, классов, объектов и связей между ними. | 5 | 12,82 |
| 4.Рабочий проект | Написание и отладка программ. Тестирование и сборка системы. | 25 | 64,10 |
| 5. Внедрение | Подготовка инструкций пользователям, написание, оформление и защита отчета (дипломного проекта). Регистрация. | 2 | 5,12 |
| Итого: |  | 39 | 100 |

**1.7. Обзор и выбор языков программирования**

В качестве основных языков программирования для реализации поставленной задачи рассматривались следующие языки:

- C# (Xamarin for Android)

- Java (Android)

В качестве основного языка для разработки приложения на ПК был выбран C#, а для Android-приложения – Java.

Критерии выбора языка программирования:

1. Возможность разработки многооконных приложений;
2. Высокая скорость и простота в изучении;
3. Наличие обширных возможностей для работы с сетью;
4. Поддержка объектно-ориентированного программирования;
5. Наличие большого сообщества;

**Язык программирования C#**

На сегодняшний день язык программирования C# является одним из самых мощных, быстро развивающихся и востребованных языков IT-отрасли. В настоящий момент на нем пишутся самые различные приложения: от небольших десктопных программок до крупных веб-порталов и веб-сервисов, обслуживающих ежедневно миллионы пользователей. [4]

Язык C# является объектно-ориентированным языком с C-подобным синтаксисом. Среди его основных преимуществ можно выделить:

1. Широкий спектр возможностей для написания программ под любые нужды;
2. Большое сообщество (MSDN);
3. Объектно-ориентированное программирование;
4. Простой и понятный синтаксис;
5. Большое количество готовых библиотек.

Среди недостатков данного языка можно выделить только то, что разработанные на нем приложения не являются кроссплатформенными. Хотя в настоящее время эта проблема уже была решена приобретением организацией Microsoft компании, предоставляющей платную (на тот момент) библиотеку для разработки кроссплатформенных приложений на C# - Xamarin. [5]

Однако и у самой библиотеки Xamarin имеется ряд недостатков.

Во-первых, компиляция приложений, написанных с использованием этой библиотеки, проходит значительно дольше, чем она же у приложений, написанных на нативном для Android языке – Java.

Во-вторых, как у плагина для Visual Studio, так и у своей среды разработки – Xamarin Studio – не поддерживаются появляющиеся подсказки с объектами при работе с XML-файлами.

Присутствие подобных недостатков создает ощущение, что библиотека Xamarin еще не до конца готова к «полевым условиям», поэтому на языке C# будет написано серверное приложение для ПК, а для реализации клиентского Android-приложения будет использован язык Java.

**Язык программирования Java**

На сегодняшний день язык Java является одним из самых распространенных и популярных языков программирования. На данный момент язык Java уже превратился в целую платформу, которая объединяет различные технологии для решения целого ряда задач. [6]

Язык Java является объектно-ориентированным. К его основным преимуществам относят:

1. Возможность решения различных задач: от создания десктопных приложений до написания крупных веб-порталов и сервисов;
2. Кроссплатформенность. Код этого языка транслируется в специальный байт-код, который затем выполняется на любом устройстве виртуальной машиной JVM.
3. Объектно-ориентированное программирование;
4. Простой и понятный синтаксис;
5. Большое количество готовых библиотек.

Среди недостатков языка Java стоит выделить большое потребление оперативной памяти по сравнению с остальными языками программирования и низкое быстродействие.

В настоящее время большая часть существующих Android-приложений написана именно на языке Java. В этом нет ничего удивительного, ведь и сама операционная система Android OS была написана на этом же языке.

Как уже было сказано, в Java for Android код компилируется значительно быстрее, чем в его аналоге – Xamarin для C#, да и сама платформа изначально предрасположена к использованию для написания приложений не только на компьютеры, но и огромный спектр различных устройств. Ввиду его неоспоримых преимуществ для написания клиентского Android-приложения будет использоваться язык Java.

**1.8. Обзор и выбор сетевых протоколов**

Сетевой протокол – главная часть любого клиент-серверного приложения. Он представляет собой набор правил и действий, позволяющий осуществлять соединение и обмен данными между двумя и более включенными в сеть устройствами.

Любой метод реализации удаленного доступа, прежде всего, зависит от выбранного протокола передачи данных, так как он влияет на то, как данные будут интерпретироваться, на надежность и порядок передачи, а также на то, как именно они будут передаваться.

Как варианты решения данной задачи рассматривались следующие протоколы:

1. Передача данных по протоколу TCP;
2. Передача данных по протоколу UDP;
3. Использование технологии RFB;
4. Использование технологии RDP.

**Протокол TCP**

TCP – это сложный протокол, требующий больших затрат времени, обусловленных его механизмом установления соединения, но он берет на себя заботу о гарантированной доставке пакетов, избавляя программиста от необходимости включать эту возможность в прикладной протокол.

TCP имеет встроенную возможность надежной доставки. Если сообщение доставлено некорректно, то будет выведено сообщение об ошибке.

Преимущества протокола:

1. Надежность;
2. Гарантированность доставки пакета;
3. Данные доходят до конечной точки в том же порядке, в котором они были отправлены.

Недостатки:

1. Необходимость установки соединения, следовательно, потеря в скорости работы;
2. Данные передаются медленно, так как работают механизмы проверки доставки.

**Протокол UDP**

UDP является очень быстрым протоколом, так как в нем определен только минимальный механизм передачи данных. Он не требует открывать соединение, так что данные могут быть отправлены сразу же, как только будут подготовлены. Однако нет гарантии того, что данные точно будут доставлены, а не потеряются по пути или не исказятся.

К преимуществам протокола относятся:

1. Отсутствие необходимости установки соединения, а значит значительный перевес в скорости передачи данных по сравнению с протоколом TCP;
2. Дополнительная быстрота передачи данных достигается засчет отсутствия проверок доставки отправленных данных.

Недостатки протокола:

1. Нет гарантии, что данные дойдут;
2. Сообщения приходят в случайном порядке.

**Протокол RFB**

RFB – клиент-серверный сетевой протокол для удаленного доступа к графическому рабочему столу компьютера. Этот протокол работает с пикселями экрана, что позволяет передавать снимки экранов любых форматов.

Достоинства протокола:

1. В отличие от рассмотренных ранее протоколов передачи данных, этот протокол ориентирован на организацию удаленного доступа к компьютеру, следовательно, уже имеет средства обеспечения качественного соединения;
2. Попиксельная работа со снимком экрана гарантирует качество доставляемого изображения при демонстрации.

Недостатки:

1. Является недостаточно универсальным. Не подойдет для организации обратной связи и подобных функций;
2. Более низкая скорость работы по сравнению с протоколом RDP, обусловленная использованием кадровых буферов и попиксельной обработки изображения.

**Протокол RDP**

Remote Desktop является прикладным протоколом, базирующимся на TCP. После установки соединения на транспортном уровне инициализируется RDP- сессия, в рамках которой согласуются различные параметры передачи данных. Для поддержания сеанса используется, так называемое, терминальное соединение (сервер не выполняет никаких вычислений).

Достоинства протокола:

1. Еще один протокол, ориентированный на организацию удаленного доступа к компьютеру со своими готовыми решениями;
2. Вместо изображения передаются графические примитивы для отрисовки на экране удаленного устройства, что значительно упрощает демонстрацию экрана;
3. Низкая нагрузка на сервер.

Недостатки:

1. Неуниверсальность, как и в случае с протоколом RFB;
2. Из-за отсутствия возможности производить вычисления на стороне сервера выполнять более сложные операции, не предусмотренные протоколом, не получится.

Исходя из полученных результатов был сделан выбор в пользу протокола передачи данных UDP. Этот протокол позволяет передавать данные быстро, без установки соединения. Нет никаких ограничений на операции, выполняемые с помощью этого протокола, а избавиться от потерь можно с помощью реализации своей технологии проверки достоверности данных.

**1.9. Выбор сред разработки**

При выборе основной среды разработки на любом языке программирования немаловажными факторами являются такие свойства среды:

1. Удобство отладки;
2. Скорость работы;
3. Автодополнение кода;
4. Приятный интерфейс (подсветка синтаксиса и прочее).

**Microsoft Visual Studio**

**Microsoft Visual Studio -** линейка продуктов компании Майкрософт, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств.[7]

Достоинства:

1. Кроссплатформенность;
2. Поддержка фреймворков;
3. Статический анализ кода, подсветка синтаксиса и ошибок;
4. Навигация по проекту и исходному коду: отображение файловой структуры проекта, быстрый переход между файлами, классами, методами и использованиями методов;
5. Большой выбор средств для реализации оконного приложения Windows;
6. Обширная библиотека для работы с сетью;
7. Удобные методы подключения к проекту сторонних библиотек.

На данный момент Microsoft Visual Studio является одной из лучших сред разработок приложений под .Net, поэтому она и будет использована для написания серверного приложения для Windows.

**Eclipse**

Eclipse – это свободная среда для разработки модульных кроссплатформенных приложений.

Достоинства:

1. Кроссплатформенность;
2. Высокая скорость работы;
3. Свободное ПО;
4. Анализ и исправление кода;
5. Рефакторинг.

Явным недостатком этой среды разработки при решении задач под Android является то, что компания Google отказалась от ее поддержки в пользу собственной многофункциональной среды Android Studio.

**Android Studio**

Android Studio**—** это интегрированная среда разработки (IDE) для работы с платформой Android.[8]

Достоинства:

1. Расширенный редактор макетов;
2. Рефакторинг кода;
3. Статический анализатор кода;
4. Встроенная утилита для подписывания приложений;
5. Средства навигации по проекту и исходному коду;
6. Свободное ПО;
7. Постоянные обновления, не позволяющие среде разработки ни на шаг отставать от развития самой платформы Android.

Выбор среды разработки под Android пал на Android Studio в силу ее неоспоримых преимуществ.

**Выводы**. В данном разделе были определены задачи клиент-серверного приложения, цель проектирования, сутью которой является создание уникальной системы удаленного управления компьютером, предназначенной для мониторинга, профилактики и исправления возникающих ошибок, а также настройки и управления параметрами устройства. Также в этом разделе были рассмотрены аналоги разрабатываемого приложения, достоинства и недостатки которых будут учитываться при разработке. Был произведен обзор технологий разработки программного обеспечения и сделан выбор средств разработки.

2. ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

**2.1. Общее проектирование**

Для решения поставленной задачи необходимо реализовать клиент-серверное приложение, обеспечивающее надежную связь между мобильным приложением и компьютером после проведения процедуры авторизации пользователя.

Необходимо выделить главные аспекты, которым оно должно удовлетворять:

1. Безопасность подключения. Удаленный доступ к компьютеру предоставляется только авторизованным пользователям. Посторонние лица не должны иметь доступа к хранимым на устройстве данным.
2. Удобство использования. Интерфейс приложения и для компьютера, и для мобильного устройства, должен быть прост и понятен.
3. Функционал. Приложение должно обладать максимальным функционалом, необходимым для работы.

**Безопасность подключения**

Основным сетевым протоколом в разрабатываемом клиент-серверном приложении был выбран протокол UDP.

UDP – это сетевой протокол передачи данных без установки соединения между сервером и клиентом. Он обеспечивает быстрое, но ненадежное подключение. Таким образом, необходимо реализовать механизм авторизации подключаемого к компьютеру устройства.

Такой механизм должен включать в себя:

1. Получение сервером данных об удаленном устройстве, которое пытается к нему подключиться. К этим данным относятся: имя пользователя, наименование устройства и удаленный IP-адрес.
2. Систему авторизации пользователя удаленного устройства: при запуске серверного приложения генерируется случайный пароль, состоящий из букв латинского алфавита верхнего и нижнего регистра и цифр; пользователь мобильного устройства получает удаленный доступ к компьютеру только после ввода правильного пароля, сгенерированного серверным приложением.

**Удобство использования**

Интерфейс приложения не должен обладать сложными непонятными пользователю элементами. Весь функционал должен быть легкодоступен, и все поля для заполнения пользователем должны быть определены и подписаны.

Серверное приложение, имеющее многооконную структуру, строится средствами языка программирования C# на платформе Microsoft .NET Framework с использованием API Windows Forms.

Windows Forms позволяет разрабатывать интеллектуальные клиенты - приложения с полнофункциональным графическим интерфейсом, простые в развертывании и обновлении, способные работать при наличии или отсутствии подключения к Интернету и использующие более безопасный доступ к ресурсам на локальном компьютере по сравнению с традиционными приложениями Windows. [MSDN]

С помощью модели программирования Windows Forms достаточно просто разработать красочный, удобный и адаптивный интерфейс серверного приложения, с которым будет просто работать с любого компьютера.

Приложение-клиент, также, состоит из нескольких модулей, выполняющих свои функции. У пользователя не должно возникать проблем при необходимости получения доступа к какому-либо модулю, поэтому интерфейс приложения должен включать в себя простые и эффективные средства перехода между ними.

Разработка приложения на платформе Android средствами языка программирования Java позволяет разбивать проектируемое приложение на независимые друг от друга компоненты, каждый из которых может стать точкой входа для программы; выделять уникальные ресурсы для различных конфигураций устройств; создавать уникальные макеты в XML-файлах для устройств с разными размерами экранов; разрабатывать простой, приятный глазу и интуитивно понятный интерфейс.

**Функционал**

Функционал программного продукта можно разделить на две группы: функционал серверного приложения и функционал клиентского приложения.

Весь функционал разрабатываемого приложения доступен пользователям в полном объеме. К нему относятся: установка подключения между клиентом и сервером; удаленное управление компьютером, на котором установлено серверное приложение; осуществление текстовой или голосовой связи между пользователями клиентского и серверного приложения; настройка личных данных пользователя и генерация нового пароля для подключения; передача файлов.

**2.2. Схема и принцип работы приложения**

1. Авторизация подключения

1.1. Серверное приложение генерирует случайный восьмизначный пароль, состоящий из латинских букв верхнего и нижнего регистра и цифр, и переходит в состояние ожидания подключения.

1.2. Пользователь клиентского приложения на мобильном устройстве вводит данные для подключения: имя пользователя и IP-адрес устройства, к которому будет производиться подключение; и нажимает кнопку «Подключиться».

1.3. Если был введен корректный IP-адрес, данные о подключаемом устройстве (имя пользователя, наименование устройства) отправляются серверному приложению.

1.4. Серверное приложение получает запрос на подключение, проверяет списки сохраненных устройств. Если подключаемого устройства в них нет, то приложение отправляет ему запрос пароля. Если устройство состоит в списке сохраненных устройств, то подключение считается установленным и дальнейшие пункты пропускаются.

1.5. Приложение-клиент получает запрос пароля и отображает диалог ввода пароля для пользователя. Пользователь вводит пароль, и приложение отправляет его обратно серверу.

1.6. Сервер получает пароль, введенный пользователем приложение-клиента, и проверяет его. Если пароль верный, подключение устанавливается. Если пароль неверный, клиенту отправляется сообщение-отказ в подключении.

1.7. После установки соединения пользователям приложения-клиента и приложения-сервера становятся доступны новые модули программы, позволяющие более детально управлять установленным соединением.

2. Инициализация данных об экране

2.1. После авторизации подключения серверное приложение отправляет клиенту команду, содержащую данные об экране.

2.2. Клиент получает данные, инициализирует структуры для работы с серверным экраном и отправляет серверу запрос на получение изображения.

2.3. Сервер получает запрос на изображение и отправляет клиенту снимок экрана с компьютера.

3. Обмен данными. В каждый момент времени серверное приложение фиксирует все изменения, происходящие с изображением управляемого компьютера и отправляет их клиентскому приложению для отображения, а приложение-клиент фиксирует все действия, совершаемые пользователем (нажатия, перетаскивания, удержания, нажатие клавиш на клавиатуре и осуществление любых других команд) и отправляет серверу для воспроизведения тех же действий на управляемом компьютере.

4. Чат и голосовая связь

4.1. Структуры для работы с чатом инициализируются сразу после авторизации подключения.

4.2. При запуске окна чата или инициализации звонка в серверном приложении не происходит никаких изменений, а на время открытия модуля чата или звонка в приложении-клиенте процесс обмена данными между клиентом и сервером приостанавливается, пересылаются только команды, ответственные за осуществление голосовой или текстовой связи.

5. Соединение останавливается пользователем одного из приложений. При этом второму приложению отправляется специальная команда, указывающая, что необходимо прекратить работу всех модулей и перейти к пункту авторизации подключения, либо прекратить работу программы.

Общая схема работы программного обеспечения представлена на рисунке 2.1.



Рисунок 2.1 – Обобщенная схема работы ПО

**2.3. Проектирование пользовательского интерфейса**

Пользовательский интерфейс – важная часть любого приложения, ведь именно он является основным средством взаимодействия пользователя с программой. Интерфейс является неотъемлемой, важной и трудоемкой частью разработки приложения.

В рамках данной квалификационной работы необходимо спроектировать интерфейс клиент-серверного приложения, а именно двух его частей:

1. Интерфейс приложения для устройства на платформе Android;

2. Интерфейс приложения для компьютера.

Основные требования к разрабатываемому приложению:

1. Визуальные эффекты и анимация должны быть простыми и ненавязчивыми;

2. Дизайн приложения должен состоять из неконтрастных, не бросающихся в глаза цветов;

3. Цветовые оформления приложений для мобильного устройства и компьютера должны полностью совпадать или иметь минимальные различия;

4. Все поля для ввода данных пользователем должны быть точно обозначены и подписаны;

5. Минимизация ввода одних и тех же данных от пользователя;

6. Использование только коротких простых фраз;

7. Легкодоступность модулей приложения;

Приняв во внимание перечисленные пункты необходимо подготовить макет приложения для демонстрации. С его помощью оценивается функционал и удобство приложения.

С использованием средств языка программирования C# и модели программирования Windows Forms был разработан уникальный дизайн для заголовков и информационных частей окон серверного приложения, диалоговых окон и элементов взаимодействия с пользователем (кнопок, текстовых полей, полей для ввода и т.д.).

Средствами языка разметки XML была написана основа дизайна для мобильного приложения на платформе Android, состоящая из определения основных сочетаний цветов, настройки поведения элементов интерфейса.

Интерфейс серверного приложения представлен несколькими модулями:

1. Окно настроек подключения, которое открывается при запуске программы и остается видимым до тех пор, пока не установится соединение между клиентом и сервером.

2. Главная панель, которая дает полный доступ ко всему функционалу приложения. Может находиться в двух состояниях: свернутом и развернутом. В свернутом состоянии отображаются только иконки всех доступных функций приложения. В развернутом состоянии к иконкам добавляются подписи. Главная панель сворачивается и разворачивается с плавной анимацией.

3. Диалоговые окна.

4. Окно чата.

5. Окно звонка.

На рисунке 2.2. представлен дизайн окна серверного приложения.

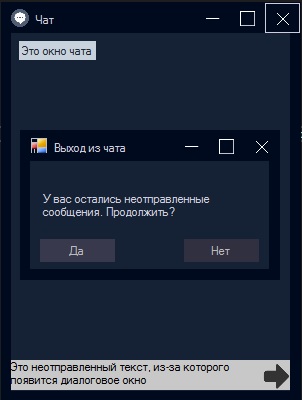


Рисунок 2.2 – Окно чата приложения-сервера

Интерфейс приложения для мобильного устройства также представлен несколькими отдельными модулями:

1. Окно подключения, которое открывается при запуске приложения. Здесь пользователь должен ввести свое имя и IP-адрес удаленного компьютера.

2. Окно управления удаленным компьютером с панелью управления всеми функциями приложения.

3. Окно чата.

4. Окно звонка.

На рисунке 2.3. представлен дизайн окон подключения и управления удаленным компьютером.

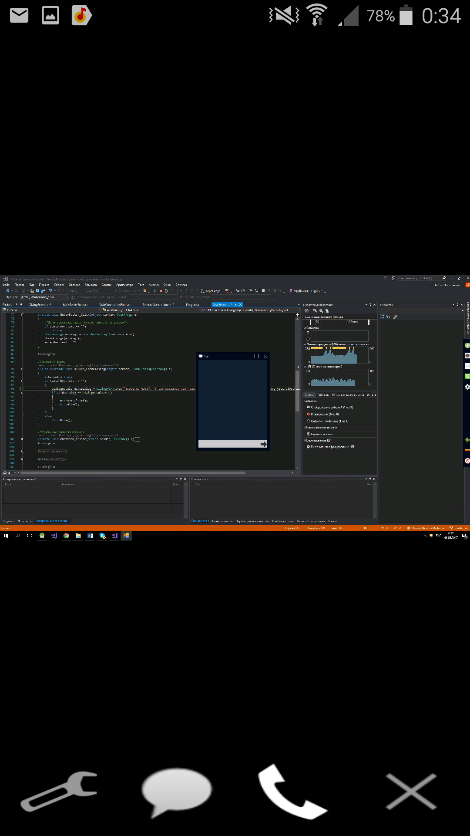
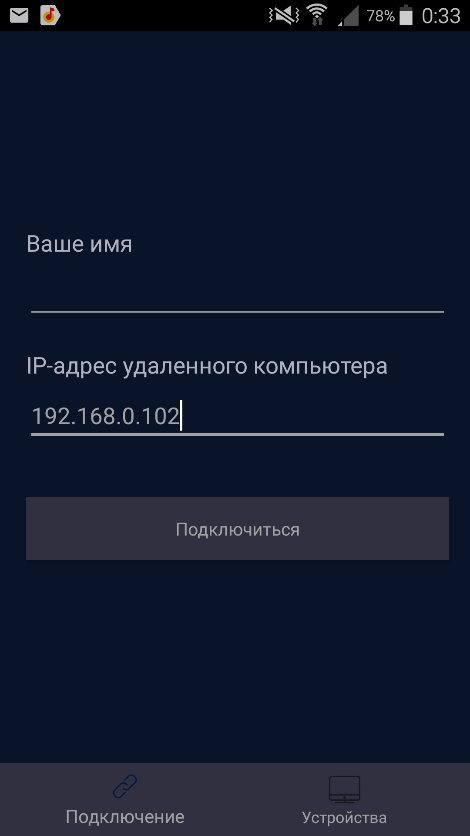


Рисунок 2.3 – Окна приложения-клиента

**2.4. Проектирование механизма передачи данных**

При реализации любого клиент-серверного приложения, прежде всего, необходимо продумать способ взаимодействия двух устройств, связываемых программным продуктом. На этапе обзора сетевых протоколов для передачи данных был выбран датаграммный протокол UDP, который предоставляет возможность быстрого обмена данными без установки соединения.

Данные между клиентом и сервером передаются в виде пакетов, представленных массивом байтов. Структура такого пакета раскрыта в таблице 2.1.

Таблица 2.1 – Общая структура пакета данных

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Команда | \ |  | , | … | , |  |
| 0 – 4 | 5 | 6 – |  |  | – |

**Описание структуры команд**

Разрабатываемое клиент-серверное приложение должно иметь четкую структуру команд для взаимодействия двух управляемых устройств. Каждая команда подает устройству, которому она предназначена, сигнал на выполнение определенного набора действий и взаимодействия с данными, которые приходят в пакете вместе с этой командой.

Пакет считается пустым, если он состоит только из кода команды.

*Команда NONE*. Данная команда, в большинстве случаев, не используется. Она может быть отправлена только в случае непредвиденной ошибки для того, чтобы дать устройству-адресату понять, что на данный момент никаких действий предпринимать не нужно.

Команда имеет код 0x00. Пакет при отправке этой команды состоит из пяти байтов – кода команды.

Команда одинаково интерпретируется и клиентом, и сервером.

*Команда HELLO*. Команда HELLO используется в случае подключения через роутер для инициализации портов для передачи данных. Она предназначена для избавления от аварийных ситуаций, возникающих из-за того, что WIFI-роутер не пропускает исходящие сообщения через порт, с которого устройству не приходило ни одного входящего сообщения.

Команда имеет код 0x01, а пакет при ее отправке также состоит из 5 байтов.

Команда HELLO одинаково интерпретируется клиентом и сервером.

*Команда INIT*. Эта команда используется для инициализации подключения. Ее может отправить только приложение-клиент.

Клиент отправляет команду INIT после нажатия пользователем кнопки «Подключиться». В пакет заносятся имя пользователя и наименование устройства, с которого будет производиться подключение.

Команда имеет код 0x02. Длина пакета суммируется из следующих значений:

– пяти байтов – длины кода команды в байтовом представлении;

– длины имени пользователя в байтовом представлении плюс один байт – разделитель;

– длины наименования устройства в байтовом представлении.

Сервер, приняв команду INIT, считывает значения имени пользователя и наименования удаленного устройства, и начинает процесс авторизации подключения.

*Команда PASSWORD*. Команда PASSWORD используется и клиентом, и сервером для проведения авторизации подключения удаленного устройства.

Сервер отправляет клиенту пустой пакет с запросом пароля.

Пакет PASSWORD, отправляемый клиентом, должен хранить в себе восьмизначный пароль, значение которого сервер сравнит со своим, сгенерированным при запуске программы, значением.

Команда имеет код 0x03. Длина пакета, отправляемого клиентом – 13 байтов. Она складывается из следующих значений:

– пять байтов – длина кода команды в байтовом представлении;

– восемь байтов – длина пароля в байтовом представлении.

*Команда CONNECT* предназначена для успешного завершения процесса авторизации подключения.

Сервер отправляет клиенту в CONNECT-пакете имя пользователя серверного приложения и имя компьютера, к которому ведется подключение. Клиент, в свою очередь, получив пакет от сервера, отправляет ему пустой пакет с командой CONNECT для завершения авторизации подключения и начала инициализации экрана.

Команда имеет код 0x04. Длина пакета, отправляемого сервером суммируется из следующих значений:

– пяти байтов – длины кода команды в байтовом представлении;

– длины имени пользователя в байтовом представлении плюс один байт – разделитель;

– длины имени компьютера в байтовом представлении.

*Команда EXIT* отправляется при необходимости остановки подключения. Пакет с данной командой может быть пустым или содержать одно значение – строку, в которой описана причина прекращения соединения.

Команда имеет код 0x06, а длина пакета с этой командой равна пяти байтам, либо суммируется из длины байтового представления кода команды и байтового представления строки, передаваемой как значение.

Команда EXIT одинаково интерпретируется и клиентом, и сервером.

*Команда SCREENINFO* оперирует при инициализации данных об экране управляемого компьютера. После установки соединения клиент отправляет серверу пустой пакет с командой SCREENINFO, а сервер должен ответить ему пакетом со всеми значениями параметров экрана. К этим параметрам относятся: количество частей, на которые будет делиться снимок экрана, количество строк и столбцов в таблице частей снимка экрана, разрешение экрана, размеры одной части снимка экрана.

Код команды – 0x09. Длина пакета, передаваемого клиенту, составляет 33 байта и суммируется из следующих значений:

– пяти байтов – длины кода команды в байтовом представлении;

– трех байтов – количества частей экрана плюс разделитель;

– шести байтов – количества строк и столбцов в таблице частей экрана, плюс двух разделителей;

– десяти байтов – разрешения экрана плюс двух разделителей;

– девяти байтов – размеров отдельной части плюс одного разделителя.

*Команда SCREEN* используется для передачи кода изображения определенной части экрана. Клиент отправляет серверу команду-запрос, состоящую из семи байтов – кода команды и номера части экрана. Сервер должен среагировать на такой запрос и отправить ответ с номером части экрана, ее координатами и байтовым представлением изображения, вырезанного из снимка экрана.

Код команды – 0x08. Пакет, отправляемый клиенту, имеет длину, суммирующуюся из следующих значений:

– пяти байтов – длины кода команды в байтовом представлении;

– трех байтов – номера части экрана плюс разделитель;

– десяти байтов – координат части экрана плюс два разделителя;

– длины байтового представления изображения, вырезанного из снимка экрана по указанным координатам с указанными размерами.

**СПИСОК ЛИТЕРАТУРЫ**

1. Что такое удаленный доступ к компьютеру. [Электронный ресурс]. http://kzncomputer.ru/network/86-chto-takoe-udalenniy-dostup (дата обращения 03.05.2017)
2. Team Viewer | Программное обеспечение для удаленной поддержки, удаленного доступа, удаленного администрирования. [Электронный ресурс]. https://www.teamviewer.com/ (дата обращения 05.05.2017)
3. LogMeIn | Многофункциональное кросс-платформенное программное обеспечение для предоставления удаленного доступа и управления компьютером. [Электронный ресурс]. https://secure.logmein.com/home/ru (дата обращения 6.05.2017)
4. Язык C# и платформа .Net. [Электронный ресурс].

https://metanit.com/sharp/tutorial/1.1.php (дата обращения 10.05.2017)

1. Подробно о Xamarin. [Электронный ресурс].

https://habrahabr.ru/post/188130/ (дата обращения 10.05.2017)

1. Введение в Java. [Электронный ресурс].

https://metanit.com/java/tutorial/1.1.php (дата обращения 10.05.2017)

1. Microsoft Visual Studio. [Электронный ресурс].

http://good-sale.ru/prog\_info/vs

1. Android Studio – Wikipedia. [Электронный ресурс].

https://ru.wikipedia.org/wiki/Android\_Studio

1. Сокеты, типы сокетов в .Net и C# [Электронный ресурс]. https://professorweb.ru/my/csharp/web/level3/3\_1.php (дата обращения 11.05.2017)
2. Программирование сокетов на Java [Электронный ресурс]. http://www.quizful.net/post/java-socket-programming (дата обращения 12.05.2017)