Министерство образования и науки Республики Башкортостан

Государственное бюджетное профессиональное образовательное учреждение

Уфимский колледж статистики, информатики и вычислительной техники

ОТЧЁТ

по производственной практике

ПП.04.01 Производственная практика

по модулю ПМ.04 Сопровождение и обслуживание программного обеспечение компьютерных систем

Специальность

09.02.07 Информационные системы и программирование

Квалификация

Программист

|  |  |
| --- | --- |
| Заведующий Центром  профессиональной карьеры  \_\_\_\_\_\_\_\_\_\_\_\_/ А.Г. Анастасьев  подпись ФИО | Руководитель практики  от предприятия  \_\_\_\_\_\_\_\_\_\_\_\_/ Ю.Ю. Стрельцова  подпись ФИО  М.П. |
| Руководитель практики  от учебного заведения  \_\_\_\_\_\_\_\_\_\_\_\_/ С.А.Каюмов  подпись ФИО | Студент группы 19П-3  \_\_\_\_\_\_\_\_\_\_\_\_/ Р.Э. Мирхайдарова  подпись ФИО  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 год |

УФА - 2022 год

СОДЕРЖАНИЕ

|  |  |
| --- | --- |
|  | лист |
| Введение | 3 |
| 1. Характеристика организационной и функциональной структуры системы управления предприятия с перечнем задач | 5 |
| 2 Сопровождение и обслуживание программного обеспечения предприятия | 7 |
| 2.1 Анализ аппаратного и программного обеспечения | 7 |
| 2.2 Анализ сетевого обеспечения предприятия | 8 |
| 2.3 Анализ различных антивирусных программ | 9 |
| 2.4 Настройка защиты системы стандартными средствами операционной системы | 10 |
| 3 Проектирование программного обеспечения для решения прикладной задачи | 14 |
| 3.1 Постановка задачи. Техническое задание на разработку программного продукта | 14 |
| 3.2 Описание программы | 21 |
| 3.3 Протокол тестирования разработанного программного продукта | 23 |
| 3.4 Руководство пользователя  Заключение | 27  31 |
| Список использованных источников | 32 |
|  |  |

ВВЕДЕНИЕ

Сопровождение программного продукта является одной из стадий жизненного цикла информационных систем. Оно осуществляется при выполнении опытной эксплуатации созданного программного продукта и состоит в реализации необходимых доработок, повышении общего уровня полезности программы. При этом могут быть выявлены недостатки продукта, ошибки в программных кодах, для этого необходимо моделировать разные действия пользователя, аварийные ситуации, в ходе которых выполняется анализ поведения программы.

Сопровождение должно обеспечить поддержку функционирования программного обеспечения в течение всего периода его эксплуатации. В процессе сопровождения должны фиксироваться и отслеживаться запросы на модификацию, оцениваться влияние предполагаемых коррективов, а также должен подвергаться модификации код и иные компоненты программного продукта, должно проводиться необходимое тестирование и, наконец, должна быть выпущена обновленная версия программного обеспечения. Помимо этого, должно проводиться обучение пользователей и обеспечиваться их ежедневная поддержка при работе с имеющимся вариантом программного продукта.

Специалистам технической поддержки производителя программного обеспечения необходимо не просто обладать доступом ко всем ключевым активам проекта, в их обязанности должно входить исправлений ошибок. В отдельных случаях такие коррективы, до выпуска новой версии программного продукта, могут осуществляться с привлечением непосредственных разработчиков продукта. Причем, разработчики продукта должны получать информацию о выявленных ошибках и, в случае нахождения необходимых решений по ликвидации ошибок специалистами технической поддержки, такие решения должны передаваться разработчикам. Разработчики могут или включить такие изменения в новую версию программного продукта, если изменения успешно прошли все необходимые тесты, или могут найти более адекватное решение. В число обязанностей специалистов службы сопровождения, в общем случае, могут входить:

* Осуществление проверки пользовательского сценария, который приводит к сбою;
* Выполнение идентификации причин сбоя, то есть, осуществление локализации ошибки и причин ее возникновения;
* Предложение необходимых исправлений или, при невозможности формирования таковых на данном этапе либо в заданные сроки, предложение обходных путей решения проблемы для достижения требуемых целей;
* Фиксация всех работ и операций, размещение описания проблемы и ее решения в базе знаний службы сопровождения, а также передача всей информации разработчикам.

Цель практики – получение навыков сопровождение и обслуживания программного обеспечения компьютерных систем.

Задачи производственной практики:

* ознакомление с предприятием (отделом), его функциями, структурой;
* ознакомление с имеющимися в отделе автоматизированными системами, аппаратным и программным обеспечением ПК, сетевым обеспечением, антивирусными программами.
* настройка защиты системы стандартными средствами операционной системы;
* разработка программного продукта, согласно задачам, выполняемым в отделе.

# 1 Характеристика организационной и функциональной структуры системы управления предприятия с перечнем задач

Уфанет – телекоммуникационный оператор связи, а также системный интегратор на территории Республики Башкортостан, Республики Татарстан, Оренбургской области, Марий Эл, Нижегородской области, Рязанской области. Центральный офис компании находится в Уфе.

Уфанет предоставляет услуги кабельного телевидения, видеонаблюдения, Интернета по технологии FTTB (оптоволокно до здания) и FTTH (оптоволокно в частный дом), телефонии, а также услуги «Умный домофон», «Телеметрия», комплекс датчиков «Умный дом». Является крупным поставщиком услуг для корпоративного сектора: системная интеграция, цифровые каналы связи, радиоэзернет, создание сетей, ИТ-аутсорсинг, видеонаблюдение, монтажные работы, обслуживание тепловых узлов и др. По данным на конец 2021 года, доступ в сеть компания Уфанет обеспечила для более, чем 1,5 млн. домохозяйств во всех регионах присутствия компании.

Компания «Уфанет» основана в 1996 году командой из 4 человек (Бахтияров Искандар, Галикеев Вадим, Гришин Олег, Камалова Нафиса). С момента образования «Уфанет» быстро росла, и на конец 2021 года в ней работает более 3,8 тыс. человек. Средний возраст сотрудника — 34 года.

Первый филиал компании за пределами Республики Башкортостан открылся в 2006 г. В [Оренбурге](https://ru.wikipedia.org/wiki/%D0%9E%D1%80%D0%B5%D0%BD%D0%B1%D1%83%D1%80%D0%B3).

Команда технической поддержки «Уфанет» несколько раз становилась призёром [Международной программы номинирования лучших из лучших в индустрии контакт-центров и обслуживания клиентов «Хрустальная Гарнитура»](http://callcenterawards.ru/).

Организационная структура организации представлена на рисунке 1.1.

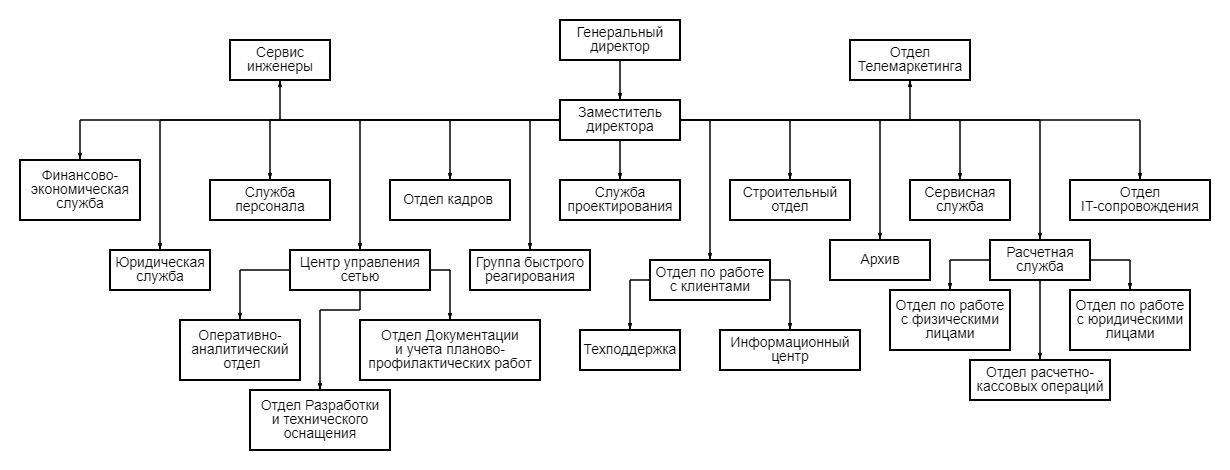


Рисунок 1.1 – Организационная структура организации

Схема расположения рабочих мест представлена на рисунке 1.2.

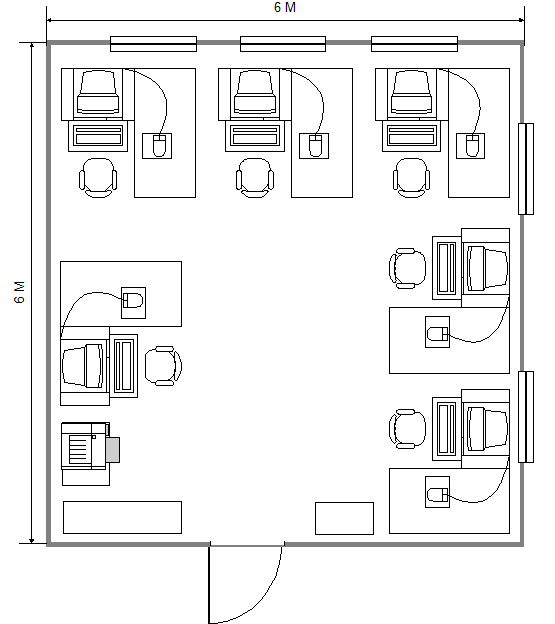


Рисунок 1.2 – Схема расположения рабочих мест

2 Сопровождение и обслуживание программного обеспечения предприятия

2.1 Анализ аппаратного и программного обеспечения

Существующая до начала модернизации ЛВС в компании была организована следующим образом: Рабочие станции одной конфигурации общим количеством 5 штук были объединены в локальную сеть, используя топологию звезда. В качестве среды передачи данных использовалась витая пара, которая обеспечивала скорость передачи данных 100Мбит/сек.

Конфигурация рабочих станций:

* процессор - Intel Pentium IV 3GHz;
* ОЗУ – DIMM DDR PC3200 1024Mb;
* жесткий диск - Samsung 120Gb;
* ОС - Microsoft Windows XP Professional.

Microsoft Outlook – персональный информационный менеджер с функциями почтового клиента, входящий в пакет офисных программ Microsoft Office. Помимо функций почтового клиента для работы с электронной почтой, Microsoft Outlook является полноценным органайзером, предоставляющим функции календаря, планировщика задач, записной книжки и менеджера контактов.

На предприятии используется офисный пакет Microsoft Office 2013. Microsoft Office – офисный пакет приложений, созданных корпорацией Microsoft для операционных систем Microsoft Windows, Apple Mac OS X. В состав этого пакета входит программное обеспечение для работы с различными типами документов: текстами, электронными таблицами, базами данных и др. Microsoft Office является сервером OLE объектов и его функции могут использоваться другими приложениями, а также самими приложениями Microsoft Office.

Microsoft SQL Server – система управления реляционными базами данных (СУРБД), разработанная корпорацией Microsoft. Основной используемый язык запросов - Transact-SQL, создай совместно Microsoft и Sybase.1 ransact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для paooibi с базами данных размером от персональных до крупных баз данных масштаба предприятия; конкурирует с другими СУБД в этом сегменте рынка.

Adobe Acrobat - пакет программ, выпускаемый компанией Adobe Systems для создания и просмотра электронных публикаций в формате PDF.

WinRAR это архиватор файлов в форматы RAR и ZIP для 32 – и 64-разрядных операционных систем Windows.

2.2 Анализ сетевого обеспечения предприятия

В качестве устройства для организации сетевого доступа использовался коммутатор D-Link DES-1026G 24port 19` rack-mount [DES-1026G]. Организация имеет в наличии четыре сервера. Основные функции, которые должны реализовывать серверы:

DataBase - сервер, хранит базу данных о сотрудниках организации и рабочие БД, также федерального и местного бюджета были организованы на платформе Windows 2003 Server SP2.

FileServer - используемый для хранения файлов, был организован на платформе Novell NetWare v.4.0.

BackupServer - используется для резервных копий первых двух серверов.

ProxyServer - является посредником ("proxy" - посредник) между рабочими станциями и интернетом.

Соединение было реализовано через ADSL модем, и обеспечивало скорость около 500Кбит/сек.

Компьютеры подключены к общей локальной сети. Сеть архива изображена на рисунке 2.2.1.

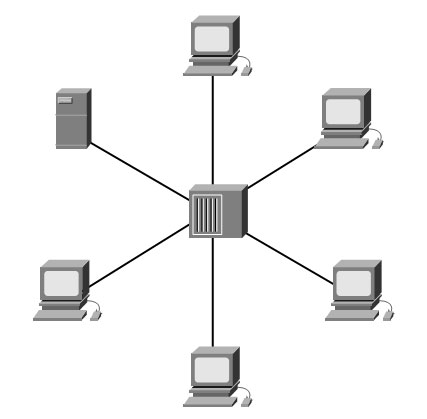


Рисунок 2.2.1 – Способ подключения к сети

2.3 Анализ различных антивирусных программ

На компьютерах предприятия используется антивирус Kaspersky.

Все продукты «Лаборатории Касперского» для Windows отлично справляются с нейтрализацией вредоносного ПО и при этом не сильно влияют на производительность системы. А в программу начального уровня встроена даже защита от Ransomware. По этому показателю его превосходит только Bitdefender, бесплатная версия которого предлагает несколько больше возможностей.

В пакете среднего уровня от «Лаборатории Касперского» есть функции для безопасности браузера, защиты ноутбуков от кражи и веб-камеры. Кроме того, он может предложить VPN (правда, с ограничениями трафика), который автоматически включается при подключении к Wi-Fi. Также в этот пакет включено ПО для Android, iOS для мобильных устройств и macOS для компьютеров. Антивирусный пакет класса премиум от Kaspersky включает возможность резервного копирования, шифрование, программу-шредер, родительский контроль и менеджер паролей без каких-либо ограничений.

2.4 Настройка защиты системы стандартными средствами операционной  
системы

Брандмауэр, он же Firewall, он же межсетевой экран — это технологический барьер, который защищает сеть от несанкционированного или нежелательного доступа. Проще говоря, Firewall — охранник вашего компьютера, как и антивирус.

Для того, чтобы настроить брандмауэр ОС Windows, нужно перейти в панель управления, вызвав окно «Выполнить» с помощью комбинации клавиш Win+R и выполнив команду «control», как представлено на рисунке 2.4.1.

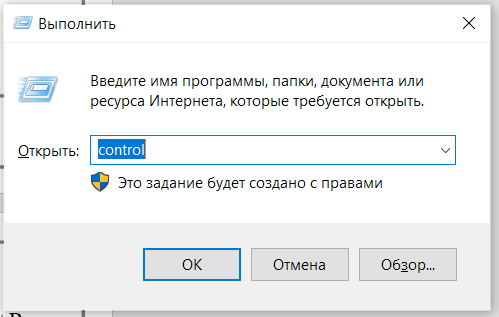


Рисунок 2.4.1 – Вызов окна «Выполнить» и ввод команды

Затем нужно поставить режим просмотра на «Крупные значки» и перейти в «Брандмауэр защитника Windows». Как установить режим просмотра, изображено на рисунке 2.4.2.

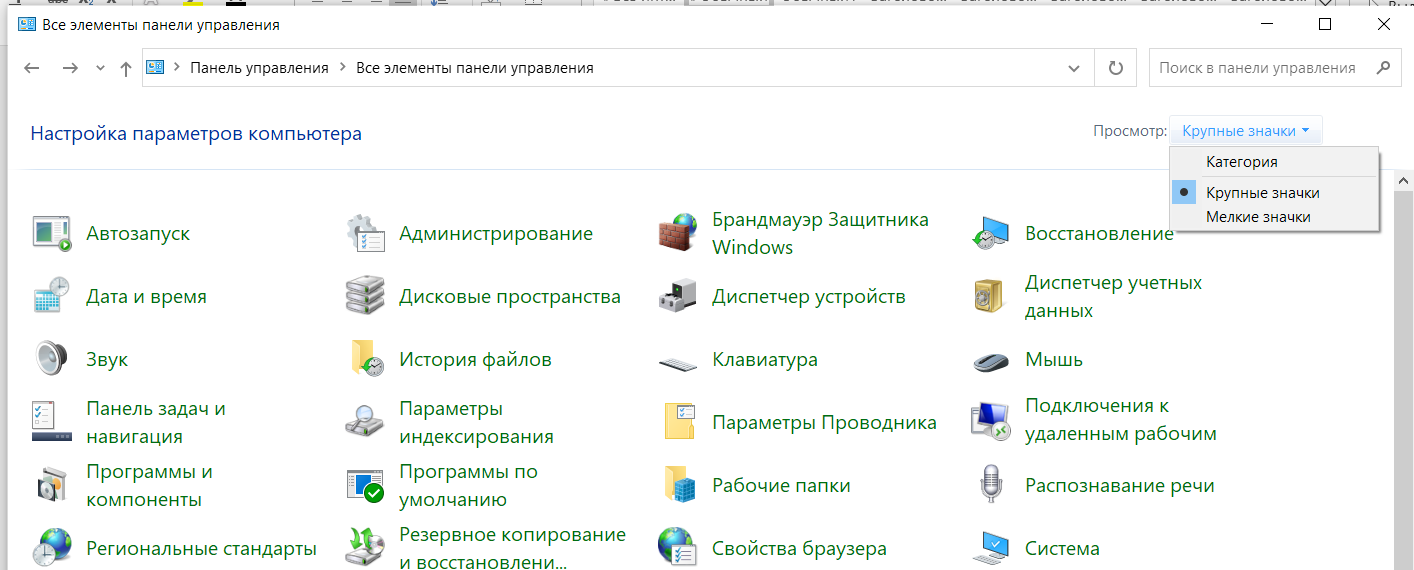


Рисунок 2.4.2 – Установка режима просмотра

Включение и отключение брандмауэра производится в соответствующем отделе. Нужный отдел отмечен на рисунке 2.4.3.

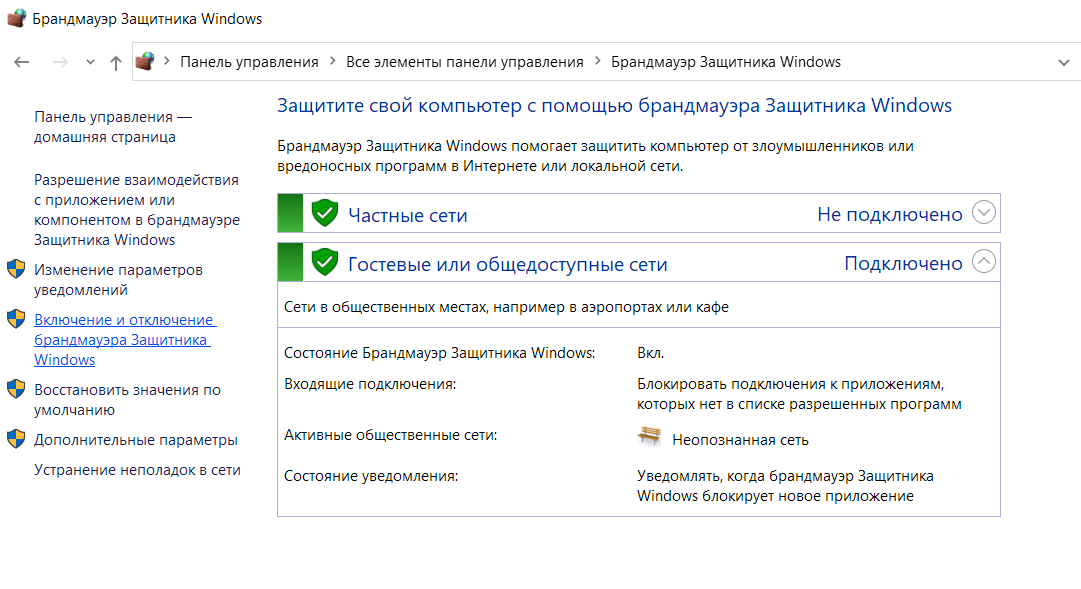


Рисунок 2.4.3 – Отдел для включения и выключения брандмауэра

Далее, необходимо произвести настройку параметров, как на рисунке 2.4.4.

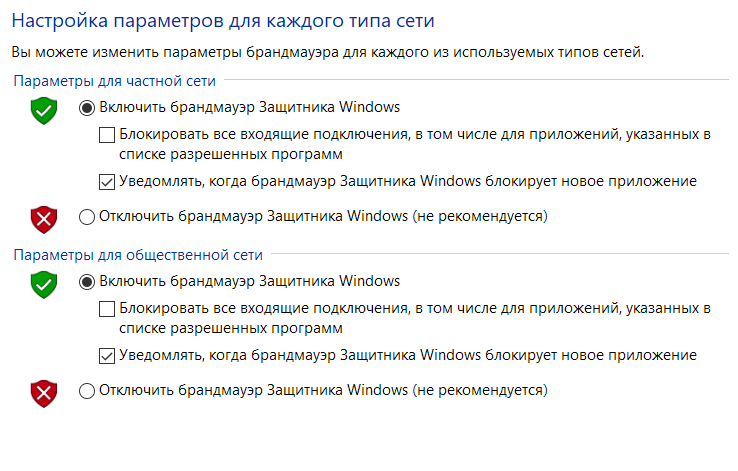


Рисунок 2.4.4 – Настройка для включения брандмауэра

Для того, чтобы настроить брандмауэр средствами ОС Linux Ubuntu, воспользуемся оболочкой под названием UFW или Uncomplicated FireWall.

По умолчанию UFW отключен. Вы можете проверить статус службы UFW с помощью следующей команды, представленной на рисунке 2.4.5.



Рисунок 2.4.5 – Проверка статуса брандмауэра

Если подключиться к Ubuntu из удаленного места, перед включением брандмауэра UFW нужно явно разрешить входящие соединения SSH. В противном случае подключиться к машине больше не получится. Чтобы настроить брандмауэр UFW для разрешения входящих соединений SSH, введите команду, представленную на рисунке 2.4.6.

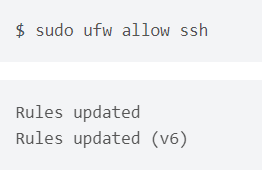


Рисунок 2.4.6 – Разрешение соединений SSH

Теперь, когда брандмауэр настроен на разрешение входящих соединений SSH, можно включить его, набрав команду, представленную на рисунке 2.4.7.



Рисунок 2.4.7 – Включение брандмауэра

В зависимости от приложений, которые работают в системе, также может потребоваться открыть порты. Общий синтаксис открытия порта представлен на рисунке 2.4.8.



Рисунок 2.4.8 – Открытие порта

Как разрешить HTTP-соединения, показано на рисунке 2.4.9.



Рисунок 2.4.9 – Разрешение HTTP

Чтобы разрешить соединения на всех портах с заданного IP-адреса источника, нужно использовать ключевое слово from за которым следует адрес источника. Пример применения представлен на рисунке 2.4.10.



Рисунок 2.4.10 – Разрешение определенного IP-адреса

Синтаксис разрешения подключений к подсети IP-адресов такой же, как и при использовании одного IP-адреса. Единственное отличие состоит в том, что нужно указать маску сети.

Чтобы разрешить подключения к определенному сетевому интерфейсу, нужно использовать ключевое слово in on за которым следует имя сетевого интерфейса, как на рисунке 2.4.11.



Рисунок 2.4.11 – Разрешение определенного сетевого интерфейса

3 Проектирование программного обеспечения для решения прикладной  
задачи

3.1 Постановка задачи. Техническое задание на разработку программного продукта

3.1.1 Техническое задание

3.1.1.1 Введение

3.1.1.1.1 Наименование приложения

Наименование приложения – «Ufanet\_App».

3.1.1.1.2 Краткая характеристика области применения

Система «Ufanet\_App» предназначена для работников предприятия, и используется для работы с клиентской базой.

3.1.1.2 Основания для разработки

Разработка выполняется на основании задания по практике.

3.1.1.3 Назначение разработки

Приложение будет использоваться работниками архива.

3.1.1.3.1 Функциональное назначение

Для работников архива приложение предоставляет возможность просмотра всех клиентов компании, их добавление, изменение и удаление.

3.1.1.3.2 Эксплуатационное назначение

Приложение должно эксплуатироваться на предприятии. Конечными пользователями приложения является администратор.

3.1.1.4 Требования к программе или программному изделию

3.1.1.4.1 Требования к функциональным характеристикам

3.1.1.4.1.1 Требования к составу выполняемых функций

Приложение должно обеспечивать возможность выполнения перечисленных ниже функций:

* функции добавления, изменения и удаления в базе данных информации клиентах;
* просмотр информации о подключенном у клиента тарифе;

3.1.1.4.1.2 Требования к организации входных и выходных данных

Данные о клиентах хранятся в базе данных. Ввод, изменение и удаление данных в базе выполняет администратор.

3.1.1.4.1.3 Требования к временным характеристикам

Требования к временным характеристикам программы не предъявляются.

3.1.1.4.2 Требования к надежности

3.1.1.4.2.1 Требования к обеспечению надежного (устойчивого) функционирования программы

Надежное (устойчивое) функционирование приложения должно быть обеспечено выполнением Заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:

* организацией бесперебойного питания технических средств;
* использованием лицензионного программного обеспечения;
* регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998 г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;
* регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

3.1.1.4.2.2 Время восстановления после отказа

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем (не крахом) операционной системы, не должно превышать 10 минут при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

3.1.1.4.2.3 Отказы из-за некорректных действий оператора

Отказы программы возможны вследствие некорректных действий оператора (пользователя) при взаимодействии с операционной системой. Во избежание возникновения отказов программы по указанной выше причине следует обеспечить работу пользователя без предоставления ему административных привилегий.

3.1.1.4.3 Условия эксплуатации

3.1.1.4.3.1 Климатические условия эксплуатации

Специальные условия не требуются.

3.1.1.4.3.2 Требования к видам обслуживания

Программа не требует проведения каких-либо видов обслуживания.

3.1.1.4.3.3 Требования к численности и квалификации персонала

При установке и настройке системы необходим системный администратор. В процессе эксплуатации с приложением работает администратор.

Системный администратор должен иметь высшее профильное образование и сертификаты компании-производителя операционной системы.

Пользователи приложения должны обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы.

3.1.1.4.4 Требования к составу и параметрам технических средств

В состав технических средств должен входить IBM-совместимый персональный компьютер (ПЭВМ), включающий в себя процессор оперативную память, видеокарту, монитор, мышь.

3.1.1.4.5 Требования к информационной и программной совместимости

3.1.1.4.5.1 Требования к информационным структурам и методам решения

Требования к информационным структурам на входе и выходе, а также к методам решения не предъявляются.

3.1.1.4.5.2 Требования к исходным кодам и языкам программирования

Исходные коды программы должны быть реализованы на языке C#.

3.1.1.4.5.3 Требования к программным средствам, используемым программой

Системные программные средства, используемые приложением, должны быть представлены лицензионной локализованной версией операционной системы.

3.1.1.4.5.4 Требования к защите информации и программ

Требования к защите информации и программ не предъявляются.

3.1.1.4.5.5 Требования к транспортированию и хранению

Требования к транспортированию и хранению не предъявляются.

3.1.1.4.5.6 Специальные требования.

Приложение должно обеспечивать взаимодействие с пользователем (оператором) посредством графического пользовательского интерфейса.

Требования к дизайну приложения:

* использование единого стиля оформления;
* выделение важного на странице;
* основные разделы должны быть доступны с первого окна в зависимости от их уровня доступа.

Должен быть разработан набор уникальных элементов фирменного стиля и визуальной идентификации: знак, логотип, цветовое решение, условные значки для элементов интерфейса.

Наличие окон: главная, изменение и добавление клиентов, информация по тарифу.

3.1.1.5 Требования к программной документации

В состав программной документации должны входить:

* техническое задание;
* руководство пользователя;
* примеры тестирования приложения.

3.1.1.6 Технико-экономические показатели

Ориентировочная экономическая эффективность не рассчитываются.

3.1.1.7 Стадии и этапы разработки

Разработка должна быть проведена в две стадии:

* техническое задание;
* проектирование и разработка.

На стадии «Техническое задание» должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии «Проектирование и разработка» должны быть выполнены перечисленные ниже этапы работ:

* разработка приложения;
* разработка программной документации;
* испытания приложения.

Содержание работ по этапам:

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

* постановка задачи;
* определение и уточнение требований к техническим средствам;
* определение требований к приложению;
* определение стадий, этапов и сроков разработки приложения и документации на нее;
* выбор языков программирования;
* согласование и утверждение технического задания.

На этапе разработки приложения должна быть выполнена работа по программированию (кодированию) и отладке приложения.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями ГОСТ 19.101-77.

На этапе испытаний приложения должны быть выполнены перечисленные ниже виды работ:

* разработка, согласование и утверждение порядка и методики испытаний;
* проведение приемо-сдаточных испытаний;
* корректировка приложения и программной документации по результатам испытаний.

3.1.1.8 Порядок контроля и приемки

Приемосдаточные испытания приложения должны проводиться согласно разработанной исполнителем и согласованной заказчиком «Программы и методики испытаний».

Ход проведения приемо-сдаточных испытаний заказчик и исполнитель документируют в протоколе испытаний.

На основании протокола испытаний исполнитель совместно с заказчиком подписывают акт приемки-сдачи приложения в эксплуатацию.

3.2 Описание программы

Программа имеет модульную структуру. При ее запуске выполняется проект на WPF ufanet\_app.exe. Схема взаимодействия модулей программы представлена на рисунке 3.2.1. Описание модулей и методов представлено в таблице 3.2.1.

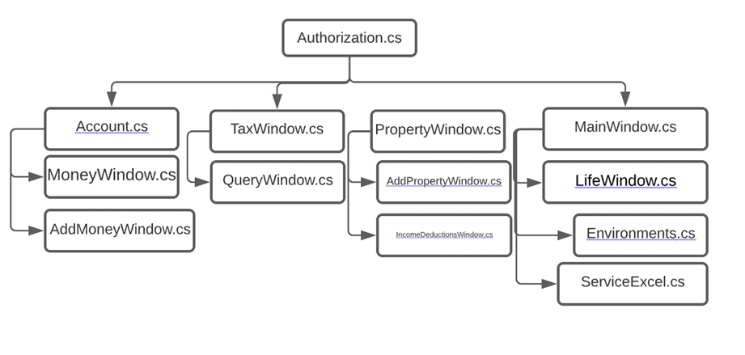


Рисунок 3.2.1 – Схема взаимодействия модулей

Таблица 3.2.1 – Описание модулей и методов

|  |  |
| --- | --- |
| Методы | Назначение |
| 1 | 2 |
| Модуль Authorization.cs – Авторизация | |
| private void ButtonEnter\_Click | Проверка данных для входа, открытие главного меню абонента |
| private void ButtonReg\_Click | Регистрация пользователя, открытие главного меню абонента |
| Модуль MainWindow.cs – Меню программы | |
| private void ButtonAccount\_Click | Открытие окна с адресами |
| private void ButtonMoney\_Click | Открытие окна баланса |
| private void ButtonProperty\_Click | Открытие окна добавить адрес |
| private void ButtonLifeSituations\_Click | Открытие помощь клиентам |
| Модуль MoneyWindow.cs – Окно отображения баланса | |
| private void ButtonAddMoney\_Click | Пополнение баланса |
| private void Window\_Activated | При загрузке окна отображение данных пользователя |
| Модуль PropertyWindow.cs – Подключенные адреса | |
| private void ShowAll | Метод отображения данных |
| private void DeleteProperty | Удаление услуги |
| private void ComboView\_SelectionChanged | Выпадающий список с элементами |
| private void ShowSelect | Выборка |
| private void ButtonAdd\_Click | Добавление адреса |
| Модуль LifeWindow.cs – Информация о договоре | |
| private void Button3NDFL\_Click | Открытие окна с кэшбэком |
| private void ButtonGetDocuments\_Click | Подать обращение |
| private void ButtonD\_Click | Подать заявку на обратный звонок |
| Модуль Account.cs – Информация абонента | |
| private void Button\_Click\_1 | Сохранение данных |
| private void EditINN\_PreviewTextInput | Только цифры в поле |
| private void PreviousPageButton\_Click(object sender, EventArgs e) | Переход на следующую страницу клиентов |
| private void NextPageButton\_Click(object sender, EventArgs e) | Переход на предыдущую страницу клиентов |
| private void FilterComboBox\_SelectedIndexChanged(object sender, EventArgs e) | Применение параметров фильтрации |
| private void SearchTextBox\_TextChanged(object sender, EventArgs e) | Применение параметров поиска |
| private void FilterComboBox\_SelectedIndexChanged(object sender, EventArgs e) | Применение параметров фильтрации |
| private void SearchTextBox\_TextChanged(object sender, EventArgs e) | Применение параметров поиска |
| private void SortComboBox\_SelectedIndexChanged(object sender, EventArgs e) | Применение параметров сортировки |
| private void DeleteButton\_Click(object sender, EventArgs e) | Удаление клиента |
| private void ClientsDataGrid\_SelectionChanged(object sender, EventArgs e) | Выбор клиента |
| private void InsertButton\_Click(object sender, EventArgs e) | Переход на окно добавления и обновления клиентов |
| private void UpdateButton\_Click(object sender, EventArgs e) | Переход на окно добавления и обновления клиентов |
| private void VisitsButton\_Click(object sender, EventArgs e) | Просмотр информации о тарифе клиента |
| InsertUpdateClientForm.cs |  |
| public InsertUpdateClientForm(int clientId = -1) | Инициализация компонентов |
| private void OkButton\_Click(object sender, EventArgs e) | Сохранение изменений |
| TarifForm.cs |  |
| public TarifForm(int clientid) | Инициализация компонентов |

3.3 Протокол тестирования разработанного программного продукта

В ходе тестирования приложения на корректных и некорректных данных не было обнаружено ошибок, которые влияли бы на работу самого приложения и всей системы.

Данное приложение удовлетворяет всем предъявленным требованиям, имеет комфортный интерфейс и интуитивно понятный функционал, исключает появления системных ошибок.

В таблице 3.3.1 представлена общая информация о тестировании. В таблицах 3.3.2 – 3.3.4 представлены протоколы тестирования: тестирование удаления клиента, тестирование добавления клиента с пустыми полями, тестирование добавления клиента.

Таблица 3.3.1 – Общая информация о тестировании

|  |  |
| --- | --- |
| Название проекта | Ufanet\_App |
| Номер версии | 1.0 |
| Имя тестера | Мирхайдарова Рада Эдуардовна |
| Даты тестирования | 02.11.2022 |

Таблица 3.3.2 – Протокол тестирования авторизации на корректных данных

|  |  |
| --- | --- |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| 1 | 2 |
| Test Case # | ufanet\_app \_1 |
| Приоритет тестирования | Высокий |
| Название тестирование | Проверка авторизации в модуле Authorization с корректными пользовательскими данными |
| Резюме испытания | Необходимо добиться корректного поведения программы при вводе корректных данных |
| Шаги тестирования | Ввод корректных данных в текстовые поля;  нажатие кнопки «Вход». |

Продолжение таблицы 3.3.2

|  |  |
| --- | --- |
| 1 | 2 |
| Фактический результат | В результате тестирования программа показала окно |
| Предпосылки | Открытие модуля Authorization(«Авторизация») |
| Постусловия | Система не зависает, находится в состоянии полной работоспособности. |
| Статус (Pass/Fail) | Pass |
| Фактический результат | В результате тестирования программа показала окно |
| Предпосылки | Открытие модуля Authorization(«Авторизация») |

Результат тестирования добавления клиента с пустыми полями представлен на рисунке 3.3.1.

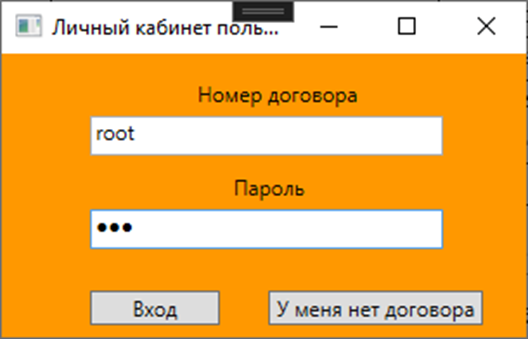


Рисунок 3.3.1 – Результат тестирования добавления клиента с путыми полями

Таблица 3.3.3 – Протокол тестирования авторизации на некорректных данных

|  |  |
| --- | --- |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| 1 | 2 |
| Test Case # | ufanet\_app \_2 |
| Приоритет тестирования | Высокий |

Продолжение таблицы 3.3.3

|  |  |
| --- | --- |
| 1 | 2 |
| Название тестирование | Проверка авторизации в модуле Authorization с некорректными пользовательскими данными |
| Резюме испытания | Необходимо добиться корректного поведения программы при вводе неправильного логина или пароля |
| Шаги тестирования | Ввод неправильного логина и пароля в текстовые поля;  нажатие кнопки «Вход». |
| Название тестирование | Проверка авторизации в модуле Authorization с некорректными пользовательскими данными |
| Резюме испытания | Необходимо добиться корректного поведения программы при вводе неправильного логина или пароля |
| Фактический результат | Сообщение о добавлении клиента, добавление клиента в базу |
| Предпосылки | Открытие окна InsertUpdateClientForm |
| Постусловия | Система находится в состоянии работоспобности |
| Статус (Pass/Fail) | Pass |

Результат авторизации с неправильным логином и паролем представлен на рисунках 3.3.2 – 3.3.3.

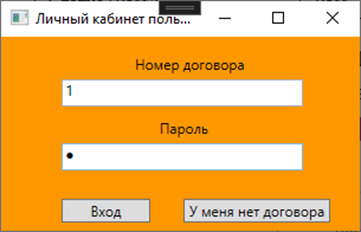


Рисунок 3.3.2 – Авторизация с вводом неправильных данных

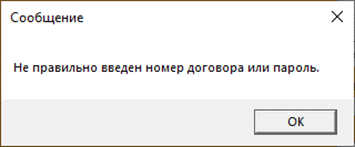


Рисунок 3.3.3 – Сообщение об ошибке

Таблица 3.3.4 – Протокол тестирования удаления клиента

|  |  |
| --- | --- |
| Описание информационных полей для тестирования | |
| 1 | 2 |
| Наименование | Описание |
| Test Case # | ufanet\_app \_4 |
| Приоритет тестирования | Высокий |
| Название тестирование | Проверка ввода незаполненного поля в поле оплаты |
| Резюме испытания | Необходимо добиться сообщения об ошибке |
| Шаги тестирования | Т Текстовые поля пустые;  Нажатие кнопки пополнить. |
| Данные тестирования | - |
| Ожидаемый результат | Программа должна сообщить пользователю о том, что нельзя использовать пустые поля |
| Фактический результат | В результате тестирования программа вывела сообщение «Введите сумму» |
| Предпосылки | Открытие модуля MoneyWindow(«Информация о балансе») |
| Постусловия | Система не зависает, находится в состоянии полной работоспособности. |
| Статус (Pass/Fail) | Pass |

Результат тестирования удаления клиента представлен на рисунках 3.3.4 – 3.3.6.

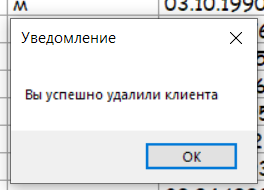


Рисунок 3.3.4 – Результат тестирования удаления клиента

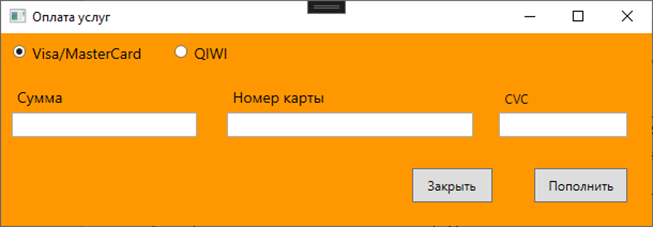


Рисунок 3.3.5 – Оплата услуг

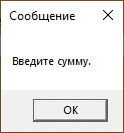


Рисунок 3.3.6 – Сообщение об ошибке

3.4 Руководство пользователя

Назначение системы

Данная программа предназначена для добавления новых адресов, услуг связи, просмотра истории пользователя.

Для работы с настольным приложением администратор должен обладать базовыми навыками работы с ПК.

Надежность системы должна быть обеспечена выполнением следующих пунктов:

* разграничение прав доступа пользователей;

Отказы программы вследствие некорректных действий пользователя при взаимодействии с системой недопустимы.

Минимальные системные требования для компьютера, на котором планируется использование программы:

* объем ОЗУ не менее 1 Гб;
* процессор Intel Pentium(R) 2.16 GHz и выше;
* операционная система Windows 7 и выше;
* не менее 50 Мб свободного места на жестком диске.

Необходимые виды периферийных устройств: оптический манипулятор типа "мышь", клавиатура, монитор.

Системные программные средства, используемые программным обеспечением, должны быть представлены лицензионной локализованной версией операционной системы Windows.

Для запуска программы необходимо запустить приложение ufanet\_app.exe из каталога, в котором установлен данный программный продукт. После этого открывается главное окно со списком пользователей. Окно со списком пользователей представлено на рисунке 3.4.1.

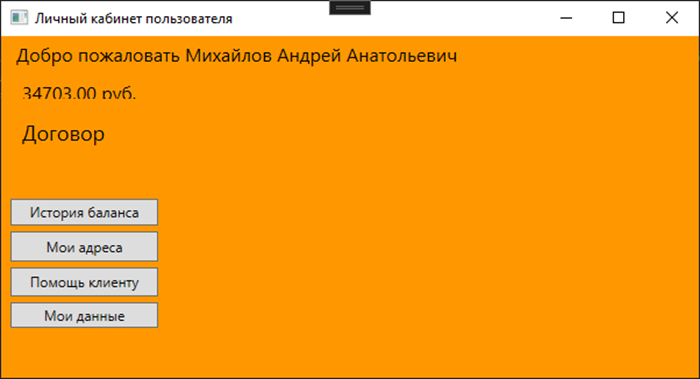


Рисунок 3.4.1 Меню абонента

При нажатии на кнопку «история баланса» абонент переходит в окно с данными о балансе, где может пополнить счет для оплаты услуг. Окно баланса представлено на рисунке 3.4.2

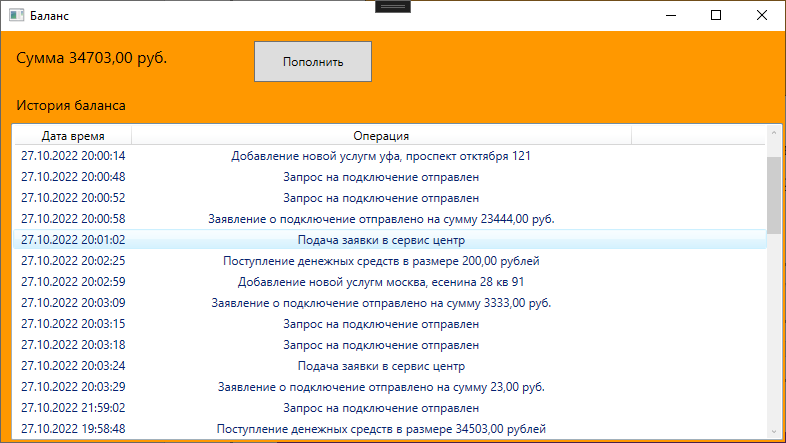


Рисунок 3.4.2 – Баланс

При выборе пололения счета показывается окно для оплаты и ввода реквизитов. Окно представлено на рисунке 3.4.3.

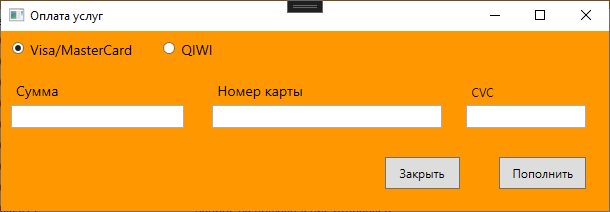


Рисунок 3.4.3– Оплата услуг

При нажатии кнопки мои адреса открывается информация о типе услуг, дате подключения и адреса проживания. Адрес можно добавить в кабинет и отменить услуги. Окно с адресами представлено на рисунке 3.4.4.

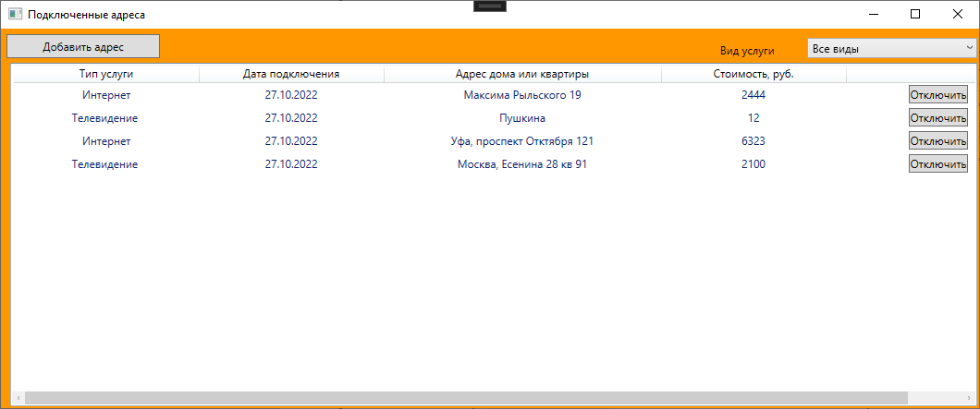


Рисунок 3.4.4– Подключенные адреса

Нажав в предыдущем окне кнопку «Добавить адрес», клиент попадает на форму ввода адреса. Форма представлена на рисунке 3.4.5.

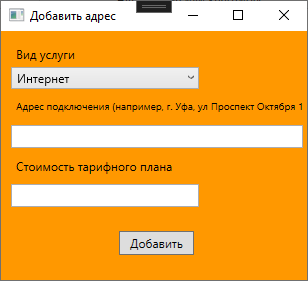


Рисунок 3.4.5– Добавить адрес

При выборе формы помощи клиенту абонент может выбрать нужное. Форма добавления купона представлена на рисунке 3.4.5.

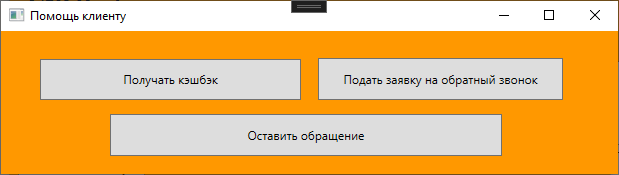


Рисунок 3.4.5– Помощь клиенту

При выборе пункта меню «Получить кэшбэк» появляется форма отправления. Форма представлена на рисунке 3.4.6.

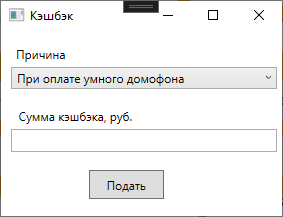


Рисунок 3.4.6 – Кэшбэк

При открытии заявки на обратный звонок абонент должен выбрать категорию услуги. Получение услуг представлено на рисунке 3.4.7.

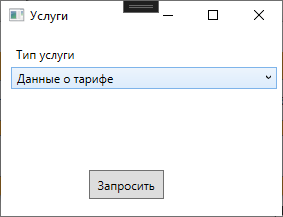


Рисунок 3.4.7 – Получить услугу

При нажатии кнопки «Мои данные» форма редактирования данных пользователя. Мои данные представлены на рисунке 3.4.8.

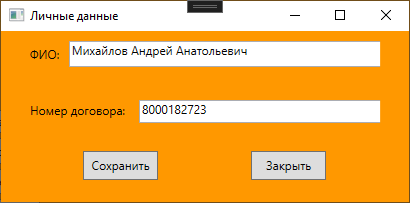


Рисунок 3.4.8– Мои данные

ЗАКЛЮЧЕНИЕ

Производственная практика является одной из основных видов учебной деятельности. Она проводится на предприятиях разных форм собственности, уровней технологии, организации производства, управление которых соответствует современным требованиям подготовки специалиста и полностью отвечает программе практики.

В ходе прохождения практики были получены навыки сопровождения и обслуживания программного обеспечения компьютерных систем. А также решены следующие задачи:

* ознакомление с предприятием (отделом), его функциями, структурой;
* ознакомление с имеющимися в отделе автоматизированными системами, аппаратным и программным обеспечением ПК, сетевым обеспечением, антивирусными программами.
* настройка защиты системы стандартными средствами операционной системы;

разработка программного продукта, согласно задачам, выполняемым в отделе.

Также был приобретен практический опыт в профессиональной сфере, получены навыки практической разработки программного продукта, а также разработка технической документации для него.

В ходе похождения производственной практики были решены все задачи, поставленные цели достигнуты.

В результате создано приложение Ufanet\_App.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 19.701-90 ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения [Текст]. – Введ. 1992–01–01. – М.: Изд-во стандартов, 1992. – 24 с.
2. ГОСТ 2.105 - 95. Общие требования к текстовым документам. Единая система конструктивной документации [Текст]. – Введ. 1996 - 07 - 01. – М.: Стандартинформ, 2007.
3. ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания [Текст]. – Введ. 1990 - 01 - 01. – М.: Стандартинформ, 2007.
4. ГОСТ Р 51583-2014. Защита информации. Порядок создания автоматизированных систем в защищенном исполнении. Общие положения. [Текст]. – Введ. 2014- 09 - 01. – М.: Стандартинформ, 2007.
5. Базовая модель угроз безопасности персональных данных при их обработке в информационных системах персональных данных [Электронный ресурс]: утв. ФСТЭК России 15 февраля 2008 // Информационно-правовая система КонсультантПлюс.
6. Шустов Н.П. Сопровождение и обслуживание программного обеспечения компьютерных систем. Учебник / Шустов Н.П. - М.: Юрайт, 2020. 220с.
7. Агуров П.А. С#. Разработка компонентов в MS VisualStudio 2008/2010.СПб.: БХВ –Петербург, 2018. 480с.
8. Кариев Ч.А. Разработка Windows-приложений на основе Visual C# [Электронный ресурс]: учебное пособие / Ч.А. Кариев. — Электрон. текстовые данные. — Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017. — 768 c.
9. Белов, Е.Б. Организационно-правовое обеспечение информационной безопасности [Текст]: учеб. пособ. для студ. учрежд. СПО / Е.Б. Костров, В.Н. Пржегорлинский. — М.: Издательский центр "Академия", 2017. — 336 с.
10. Кузнецов, С. Д. Основы баз данных / С.Д. Кузнецов. - М.: Бином. Лаборатория знаний, Интернет-университет информационных технологий, 2017. - 488 c.
11. Стружкин, Н. П. Базы данных. Проектирование. Учебник / Н.П. Стружкин, В.В. Годин. - М.: Юрайт, 2016. - 478 c.
12. Якобсон А. Язык UML. Руководство пользователя. Второе издание. ДМК, 2016, 496 с.
13. Албахари Дж. C# 6.0. Справочник. Полное описание языка [Текст] / Дж. Албахари, Б. Албахари — 6-е изд. — Москва: Вильямc, 2016. — 1040 c.
14. Базы данных. Проектирование. Практикум. Учебное пособие для СПО: Стружкин Н.П. – М.: «Юрайт», 2018 г.
15. Биллиг В. А. Основы объектного программирования на C# (C# 3.0, VisualStudio 2008) [Текст]: учебное пособие / В. А. Биллиг. — Москва: Интернет-университет информационных технологий, Бином. Лаборатория знаний, 2016. — 584 c.
16. Либерти, Д. Программирование на C# /Д. Либерти. – СПб.: Питер, 2019. – 688 с.:ил.
17. WindowsForms. Программирование на C# [Электронный ресурс]. — Режим доступа: http://csharpcoding.org/category/windows-forms/,свободный
18. Википедия [Электронный ресурс] // Свободная энциклопедия. - Режим доступа: http://ru.wikipedia.org/wiki/, свободный.
19. METANIT [Электронный ресурс] // Сайт о программировании. - Режим доступа: https://metanit.com/, свободный.
20. Microsoft [Электронный ресурс] // Сайт о программировании. - Режим доступа: https://microsoft.com/, свободный.