**Содержание**

Введение 4

1 Общая часть 6

1.1 Структура предприятия, место и функции информационного отдела 7

1.2 Анализ бизнес-процессов и постановка задачи10

1.3 Анализ средств разработки 11

2 Специальная часть 11

2.1 Проектирование архитектуры решения 11

2.2 Проектирование базы данных 11

2.3 Проектирование программных модулей решения 11

2.4 Реализация технического решения 11

2.4.1 Разработка пользовательского интерфейса решения 11

2.4.2 Разработка программных модулей решения 11

2.5 Тестирование разработанного решения 11

3 Организационно-экономическая часть 11

3.1 Расчет сметы затрат на разработку программного решения 11

3.2 Расчет экономической эффективности внедрения программного решения 11

4 Мероприятия по технике безопасности и охране окружающей среды 11

4.1 Организация безопасности труда на предприятии 11

4.2 Техника безопасности при работе с вычислительной техникой 11

Заключение 11

Список использованных источников 11

# Введение

Влияние сети Интернет на различные аспекты социального взаимодействия на сегодняшний день неоспоримо. Возможности мобильных приложений позволяют интернет-пользователям потреблять и передавать большой объем информации в режиме реального времени при помощи своих смартфонов. Тенденции, формирующиеся в рамках широкого распространения информационных технологий, говорят об актуальных сегодня мобильных приложениях – мессенджерах.

Мессенджеры – это новый способ коммуницирования между людьми, вне зависимости от географических особенностей, посредством обмена мгновенными сообщениями. В условиях развития Интернета как публичного пространства с возможностью доступа к различной персональной информации у пользователей возникает потребность в осознанном потреблении и приватном взаимодействии. Потому и растет актуальность среди приложений, предоставляемых подобные возможности.

Мессенджер — это программа для мгновенного обмена текстовыми сообщениями, аудиозаписями, фотографиями и другими мультимедиа.

Во время прохождения производственной практики в ООО «ПТ-СОФТ» было дано задание, которое заключалось в реализации системы корпоративного коммуницирования, при помощи которой обмен данными о ходе разработки проектов должен ускориться, что позволит выполнять задачи быстрее.

В соответствии с полученным заданием, была определена цель дипломного проекта: «Реализация системы корпоративного коммуницирования для ООО «ПТ-СОФТ»».

Исходя из поставленной цели были определены следующие задачи дипломного проекта:

* 1. Изучить требования заказчика и провести анализ предметной  
     области.
  2. Спроектировать архитектуру решения.
  3. Спроектировать базу данных.
  4. Спроектировать программные модули решения.
  5. Разработать пользовательский интерфейс решения.
  6. Разработать программные модули решения.
  7. Протестировать созданное решение.
  8. Рассчитать стоимость программного решения.
  9. Описать меры техники безопасности.
  10. Оформить техническую документацию в виде пояснительной  
      записки на дипломный проект.

# 1 Общая часть

# 1.1 Структура предприятия, место и функции информационного отдела

Организация ООО «ПТ-СОФТ» занимается разработкой программного обеспечения. Структура предприятия выглядит следующим образом:



Рисунок 1 – Структура предприятия

В список разрабатываемого программного обеспечения включено следующие:

Разработка микропрограммного обеспечения для ARM, DSP, AVR;

Разработка клиент-серверных архитектур с использованием ASP.Net Core, React JS;

Разработка программного обеспечения для системы управления тормозами поездов с использованием ПЛК, HMI, Scada;

Сервисное программное обеспечение для преобразователей частоты, накопителей энергии, зарядных станций.

# 1.2 Анализ бизнес-процессов и постановка задачи

На предприятии ведётся разработка большого количества проектов. Так же каждый проект может быть отсрочен до неопределённого срока, в связи с чем необходимо хранить информацию по каждому проекту, а также все обсуждения, связанные с ним.

При попытке использования сторонних программ мог создаваться визуальный шум, работникам предприятия могли писать пользователи, не относящиеся к предприятию.

Так же был выявлен недостаток функционала, который мог бы быть реализован в своём приложении.

В процессе анализа бизнес-процессов были выявлены следующие недостатки:

лишний визуальный шум в популярных приложениях;

ограниченный бесплатный функционал;

сложность реализации обсуждения нескольких разрабатываемых программ.

В связи с вышеперечисленными причинами была поставлена задача по разработке собственного приложения.

# 1.3 Анализ средств разработки

В настоящее время мессенджеры можно разработать при помощи следующих технологий и средств разработки:

1. Системы управления базами данных (СУБД):

Microsoft Access – реляционная система управления базами данных (СУБД) корпорации Microsoft. Входит в состав пакета Microsoft Office;

Oracle Database — объектно-реляционная клиент-серверная СУБД, одна из первых и самых популярных в мире. Платная, сложная, подходит для больших проектов.

1. Среды разработки программного обеспечения:

Android Studio — интегрированная среда разработки (IDE) для работы с платформой Android. Android Studio, основанная на программном обеспечении IntelliJ IDEA от компании JetBrains, — официальное средство разработки Android приложений. Данная среда разработки доступна для Windows, macOS и GNU/Linux.

App Inventor - Популярный no-code-конструктор от Google с большим набором готовых модулей. Размещая их, пользователь создаёт дизайн будущей программы, а бизнес-логика реализуется с помощью простейших алгоритмов.

Draftbi - Блочный no-code-конструктор с большими возможностями и гибкой кастомизацией встроенных шаблонов. Размещая и настраивая модули внутри окна, пользователь создаёт макет будущего приложения. Это очень похоже на работу в графических редакторах вроде Canva или Figma.

1. Языки программирования:

Java – строго типизированный объектно-ориентированный язык программирования общего назначения, разработанный компанией Sun Microsystems;

C++ – компилируемый, статически типизированный язык программирования общего назначения;

Python — мультипарадигмальный язык программирования. Полностью поддерживаются объектно-ориентированное, структурное, обобщённое, функциональное программирование и метапрограммирование.

1. Технологии доступа к данным:

ODBC – разработанный компанией Microsoft универсальный интерфейс программирования приложений для доступа к базам данных;

BDE – технология доступа к данным SQL Links, имеющая возможность взаимодействовать с ODBC через специальные «интерфейсы-мосты»;

OLE DB – набор COM-интерфейсов, которые позволяют приложениям унифицировано работать с данными разных источников и хранилищ информации.

В ходе реализации мессенджера использовались следующие средства разработки и технологии:

язык программирования C#, потому что удобен в использовании и предоставляет широкий спектр возможностей;

система управления базами данных Microsoft SQL Server, которая позволяет структурировать информацию и манипулировать данными;

технология Xamarin.Forms, так как поддерживает гибкий поток, размещающий элементы управления на основе их содержимого, и позволяет создавать более наглядный, современный пользовательский интерфейс;

Windows Communication Foundation (WCF) платформа для создания приложений. С помощью WCF можно отправлять данные в виде асинхронных сообщений из одной конечной точки службы в другую. Конечная точка службы может входить в постоянно доступную службу, размещаемую в IIS, или представлять службу, размещаемую в приложении.

среда разработки программного обеспечения Visual Studio, которая выгодно отличается эффективностью и широким функционалом.

# 2 Специальная часть

# 2.1 Проектирование архитектуры решения

Системы необходимо проектировать с учетом их дальнейшего развития. Для проектирования системы, устойчивой к таким изменениям, следует предположить, как она будет изменяться на протяжении отведенного ей времени жизни.

Для предоставления результатов проектирования планируемого разрабатываемого решения наиболее точным является проектирование архитектуры программной системы.

Разрабатываемое решение на мобильной платформе построено на клиент-серверной архитектуре, где клиентом является мобильное приложение, а сервером – компонент WCF размещенный на сервере. Для обмена между клиентом и сервером используется протокол HTTP, данные передаются в формате класса Message.

Подробное описание взаимодействия этих компонентов представлено на диаграмме компонентов (рисунок 2).

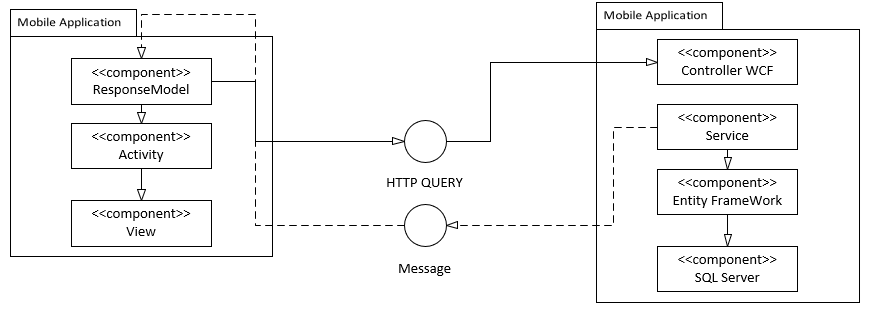


Рисунок 2 – Диаграмма компонентов

На сервере размещена база данных, которая будет содержать информацию о сотрудниках, сообщениях, участников чатов, файлах и т.д.

Разработка серверной части осуществляется на языке программирования C# при использовании платформы WCF и взаимодействия с базой данных посредством фреймворка EntityFramework.

# 2.2 Проектирование базы данных

Проектирование базы данных разрабатываемого решения в целях соблюдения логического следования проводится в несколько этапов:

1. Анализ предметной области.
2. Проектирование концептуальной модели базы данных.
3. Проектирование логической модели базы данных.
4. Проектирование физической модели базы данных .

В данной предметной области необходимо хранить данные о сообщениях, пользователях, ролей пользователей, чатах и группах.

У каждого пользователя есть своя роль, которая определяет его функционал. Так же пользователь имеет ФИО, логин, пароль и путь к каталогу, где хранятся файлы, которые он отправлял.

Пользователь состоит в чатах, которые служат связующим звеном пользователей и сообщений.

Сообщения содержат в себе текст сообщения, дату отправки, чат, в котором оно было отправлено, и файл, если такой имеется.

Так же системой предусмотрена совместная разработка, которая проявляется возможностью обсуждения одного проекта.

Проект включает себя начальника проекта, файлы прикреплённые к этому проекту, а так же участников.

На основании анализа предметной области составляется концептуальная модель базы данных. Концептуальное проектирование является основой всего процесса проектирования баз данных.

В данном случае для описания концептуальной модели используется нотация Питера-Чена, позволяющая отобразить сущности, атрибуты и связи между сущностями (рисунок 3).

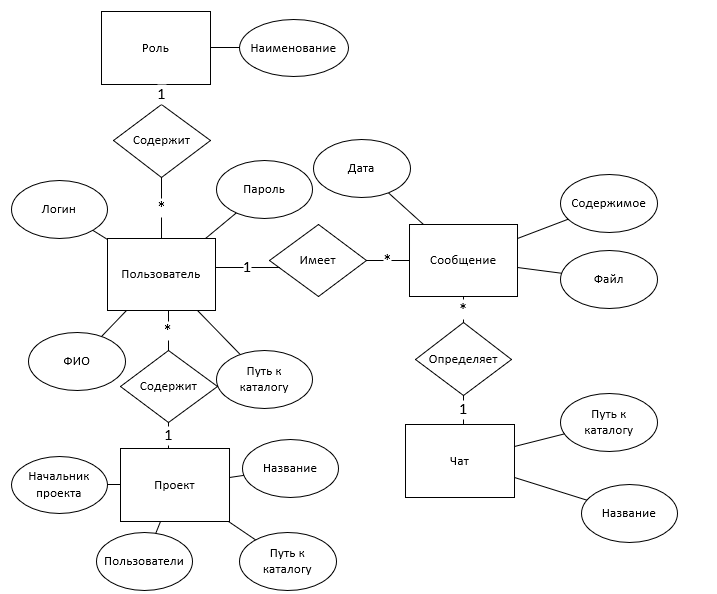


Рисунок 3 – Концептуальная модель базы данных

Проектирование концептуальной модели базы данных и анализ предметной области позволили определить сущности, атрибуты и связи между ними. Сущности и их атрибуты представлены в таблице 1.

Таблица 1 – Сущности и атрибуты базы данных

|  |  |  |  |
| --- | --- | --- | --- |
| **Ключ** | **Наименование** | **Тип значения** | **Ограничение** |
| Сущность «Роли» | | | |
| PK | Идентификатор | int | Not null, identity |

Продолжение Таблицы 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Ключ** | **Наименование** | **Тип значения** | **Ограничение** |
|  | Наименование роли | nvarchar(100) | Not null |
| Сущность «Пользователи» | | | |
| PK | Идентификатор | int | Not null, identity |
|  | Фамилия | nvarchar(100) | Not null |
|  | Имя | nvarchar(100) | Not null |
|  | Логин | nvarchar(100) | Not null |
|  | Пароль | nvarchar(100) | Not null |
| FK | Роль | int | References «Роли» |
| Сущность «Чаты» | | | |
| PK | Идентификатор | int | Not null, identity |
|  | Название | nvarchar(100) |  |
|  | Путь к каталогу | nvarchar(100) |  |
| Сущность «Сообщения» | | | |
| PK | Идентификатор | int | Not null, identity |
|  | Содержимое | Nvarchar(max) |  |
|  | Дата | datetime | Not null |
|  | Файл | Nvarchar(100) |  |
| FK | Пользователь | int | References «Пользователи» |
| FK | Чат | int | Refences «Чаты» |
| Сущность «Проект» | | | |
| PK | Идентификатор | int | Not null, identity |
|  | Название | Nvarchar(100) | Not null |
|  | Путь к каталогу | Nvarchar(100) |  |
| FK | Начальник | int | Refences Пользователи |
| Сущность «Пользователи в проектах» | | | |
| FK | Проект | int | References «Проекты» |
| FK | Чаты | int | References «Чаты» |

Основанием для проектирования логической модели базы данных является концептуальная модель. В отличие от концептуальной модели базы данных, логическая модель является начальным прототипом будущей физической модели, включающая в себя сущности, атрибуты и связи, которые отражают бизнес-правила и бизнес-информацию проектируемой базы данных.

Для проектирования логической модели базы данных используется ER-модель в IDEF1X (рисунок 4):

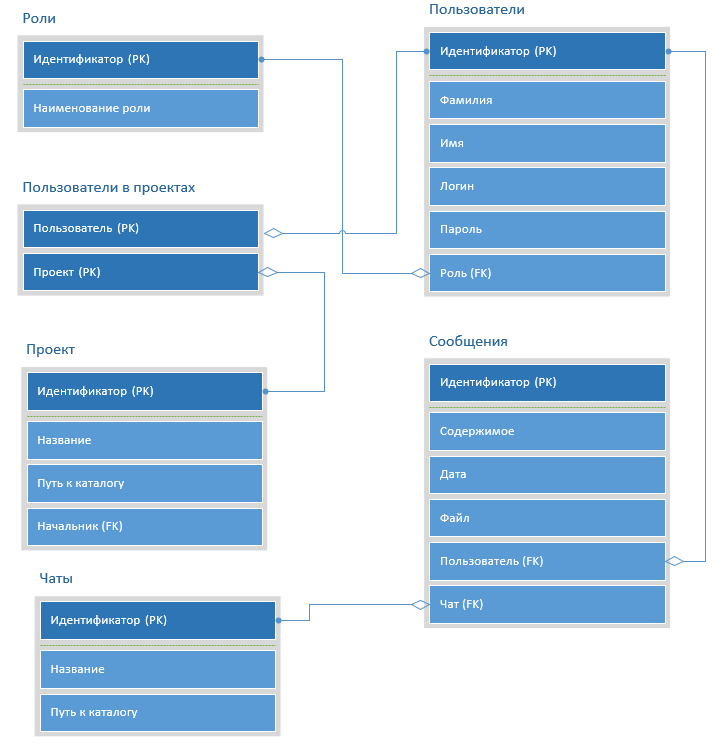


Рисунок 4 – Логическая модель БД

По завершению этапа анализа предметной области, проектирования логической и концептуальной модели, следует разработка физической модели базы данных (рисунок 5):

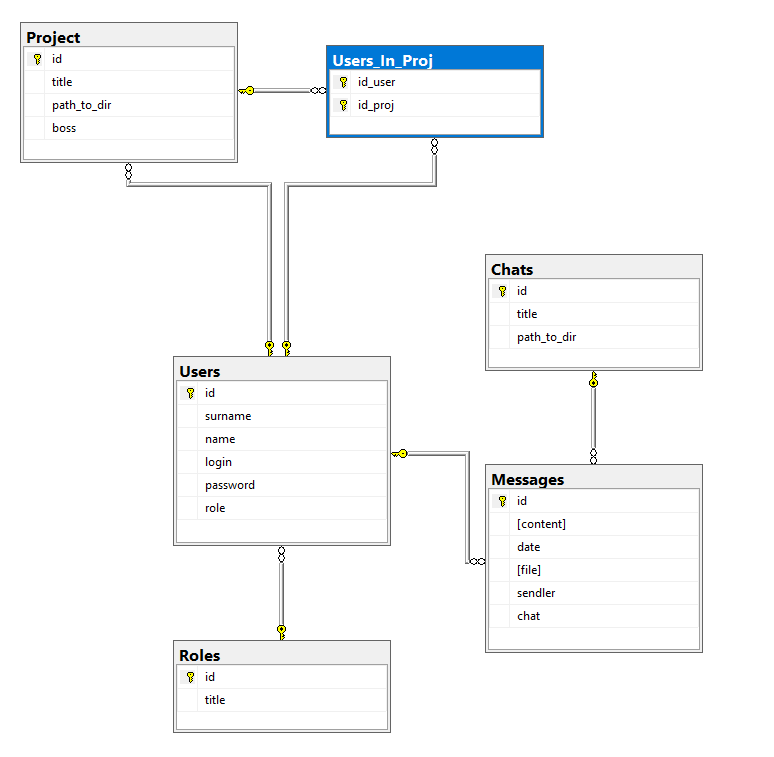


Рисунок 5 – Физическая модель базы данных

# 2.3 Проектирование программных модулей решения

Основной задачей мессенджера является отправка сообщений между пользователями. Схема отправки сообщений описана на рисунке 6:

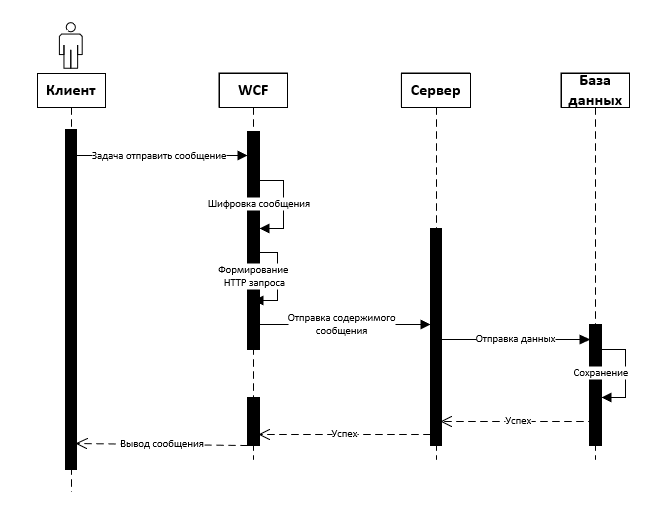


Рисунок 6 – Схема отправки сообщения

Клиент – мобильное приложение, с которого осуществляется отправка сообщения.

WCF – Windows Communication Foundation, платформа для работы архитектуры «клиент-сервер».

Сервер – компьютер, обрабатывающий запросы клиентов и работающий с базой данных напрямую.

База данных – система хранения данных.

Платформа WCF используется в каждой схеме работы клиента с сервером. В том случае, где платформа будет выполнять те же функции что и на рисунке 6 она будет пропущена.

Для определения какой именно пользователь отправляет сообщение реализован процесс авторизации на рисунке 7:

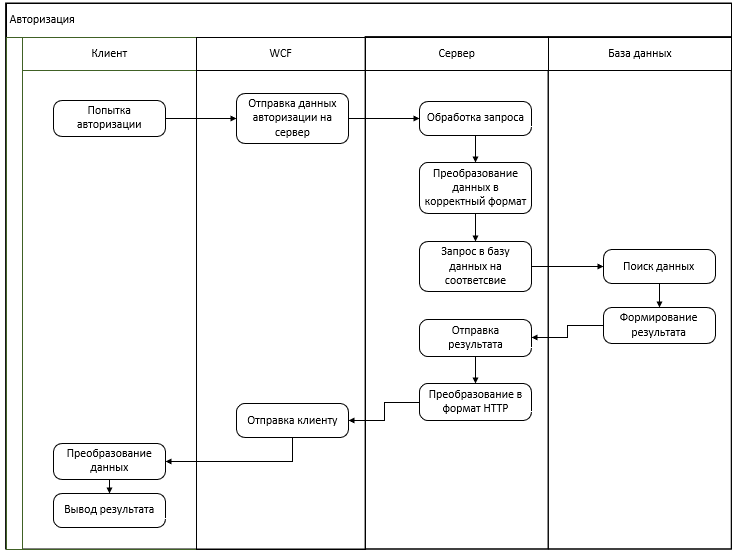


Рисунок 7 – Процесс авторизации

# 2.4 Реализация технического решения

# 2.4.1 Разработка пользовательского интерфейса решения

Разработка интерфейса напрямую связана с требованиями заказчика. Цветовая палитра и внешний вид были сформированы в соответствии с требованиями.

Навигация по интерфейсу происходит древовидным образом и выглядит следующим образом на рисунке 8:

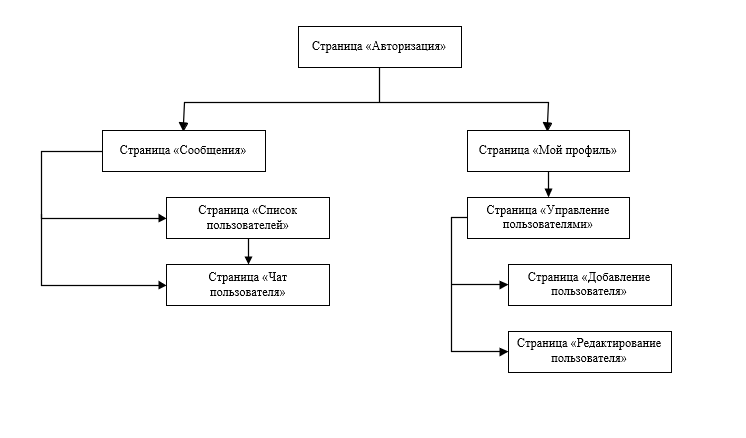


Рисунок 8 – Диаграмма дерева переходов

Приложение имеет схожий дизайн рабочих окон и при первом запуске откроется страница авторизации на рисунке 9.

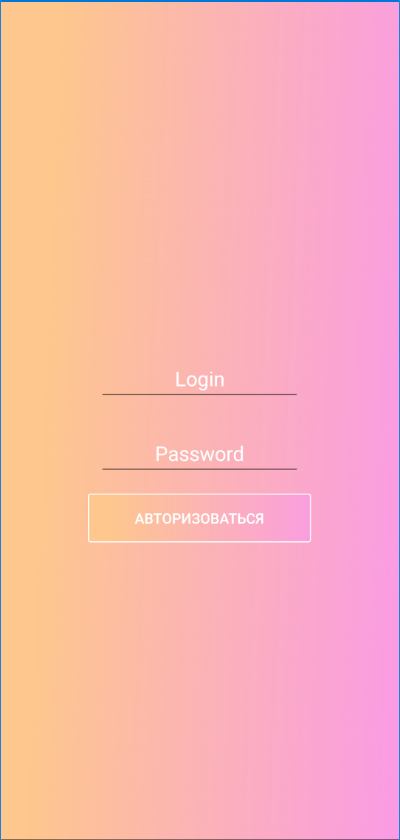


Рисунок 9 – Страница авторизации

Для навигации используется меню, которое располагается в нижней части мобильного приложения.

Окно сообщений содержит информацию о сообщениях на рисунке 10.

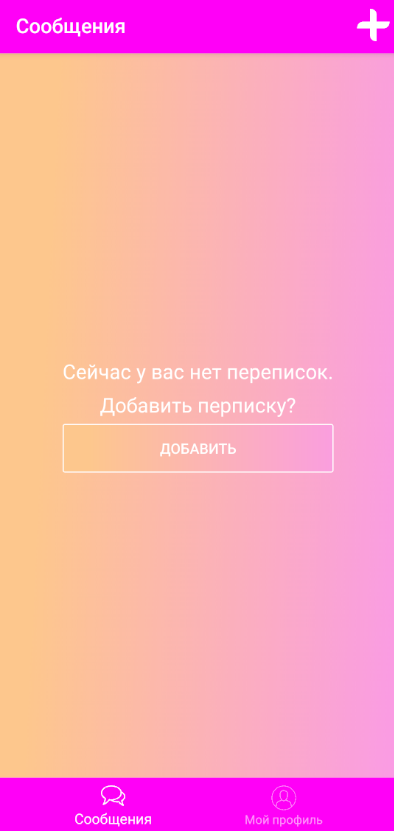


Рисунок 10 – Страница сообщений

Со страницы сообщений можно осуществить переход на страницу диалога на рисунке 11.

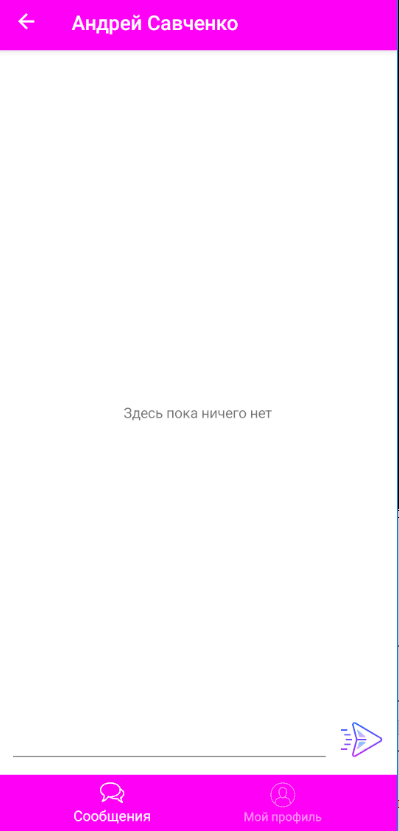


Рисунок 11 – Страница диалога

При добавлении диалога всплывает форма поиска пользователей на рисунке 12.



Рисунок 12 – Страница пользователей

При выборе пользователя откроется страница диалога (рисунок 11).

В приложении присутствует форма профиля (рисунок 13).

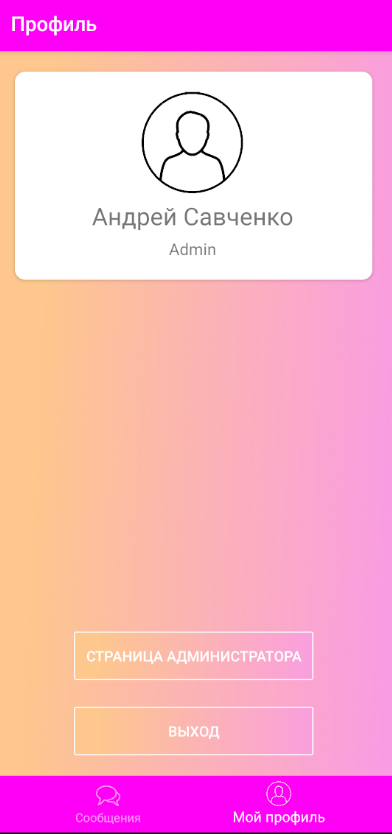


Рисунок 13 – Страцина профиля

На данной странице отображается ФИО пользователя и его логин. В случае если в системе зарегистрирован администратор добавляется кнопка «Страница администратора».

При нажатии на данную кнопку откроется страница поиска пользователя, с возможностью добавления нового пользователя (рисунок 14).

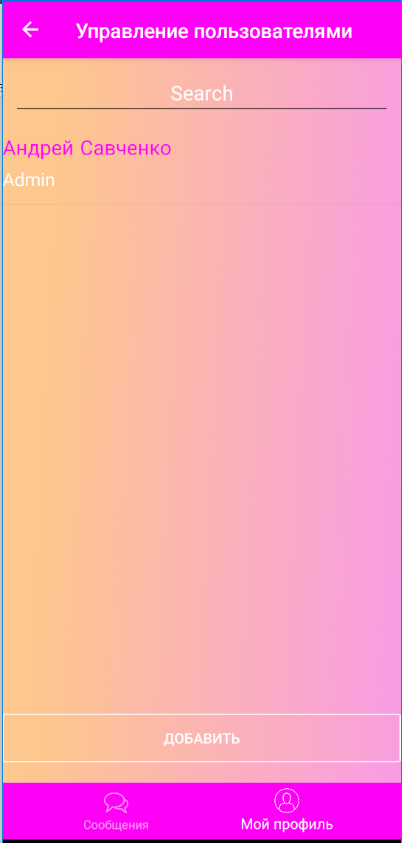


Рисунок 14 – Управление пользователями

При нажатии на кнопку добавить откроется окно добавления пользователя. При выборе пользователя откроется то же окно, только с данными о текущем пользователе (рисунок 15).

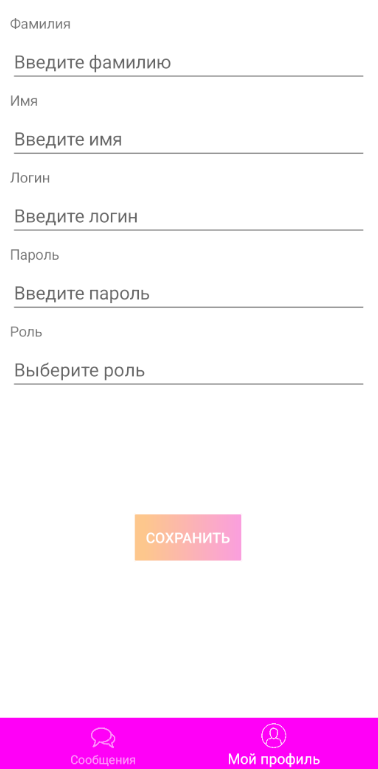


Рисунок 15 – Страница добавления пользователя

# 2.4.2 Разработка программных модулей решения

Для обеспечения безопасности передачи данных в сети данные шифруются следующим образом (рисунок 16).

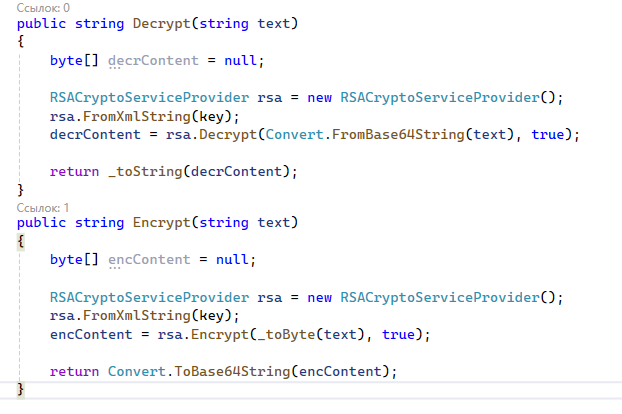


Рисунок 16 – Шифрование данных

Затем при авторизации данные из базы данных дешифруются в приложении и сравниваются с введёнными (рисунок 17).

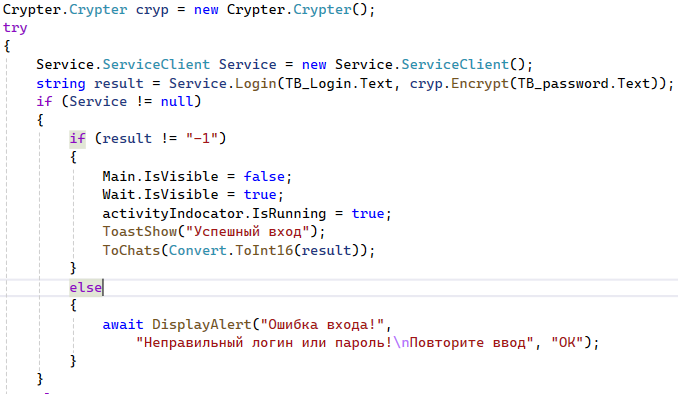


Рисунок 17 – Авторизация

После процесса авторизации идёт загрузка данных о диалогах пользователя (рисунок 18).

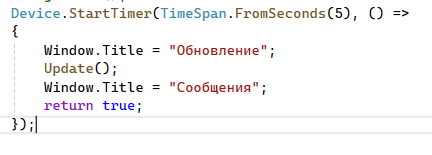


Рисунок 18 – Запуск процесса обновления

Рисунок 19 содержит код вывода сообщений.

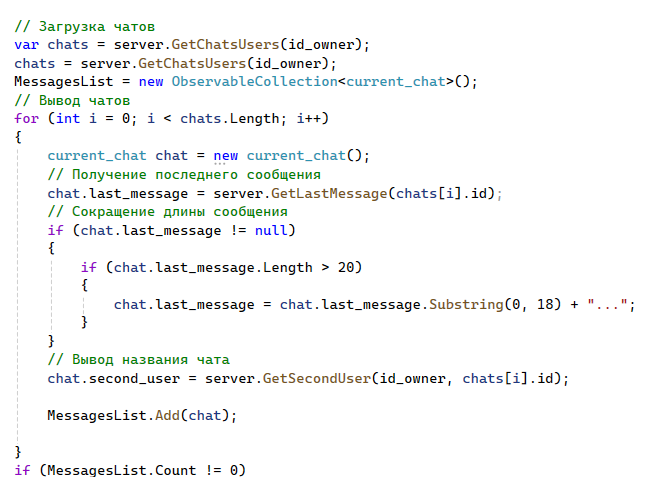


Рисунок 19 – Вывод сообщений

При выборе пользователя происходит открытие диалога и загрузка всех сообщений (рисунок 20).



Рисунок 20 – Загрузка сообщений

Для плавной загрузки реализованы метода «Hide» и «Visable», которые позволяют скрыть загрузку (рисунок 21).

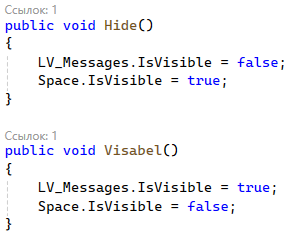


Рисунок 21 – Методы плавной загрузки

При нажатии кнопки отправления реализован следующий код (рисунок 22).

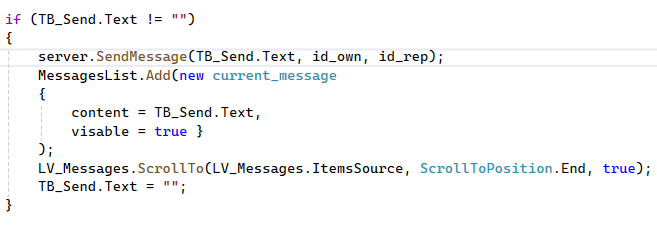


Рисунок 22 – Отправка сообщения

Так же реализовано добавление нового диалога, при выборе пользователя в соответствующем окне (рисунок 23).

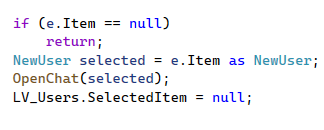


Рисунок 23 – Добавление нового диалога

# 2.5 Тестирование разработанного решения

Обработка исключительных ситуация позволяет предотвратить действия пользователя, способные повлечь за собой нарушение целостности программы и ее составляющих.

Каждая ситуация обрабатывается программой и выводит соответствующее сообщение пользователю на экран с указанием ошибки.

Пример обработки ошибок:

Авторизация:

При попытке авторизации на сервер отправляется пароль в зашифрованном виде. В случае успешной авторизации вернётся id авторизированного пользователя. В случае ошибки сработает обработка исключения. Пример кода можно увидеть на рисунке 17. В случае если не были найдены введённые данные пользователь получит следящие сообщение (рисунок 24).

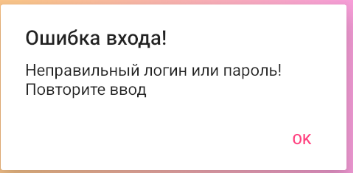


Рисунок 24 – Вывод сообщения о неудачной авторизации

Процесс ожидания ответа сервера:

При отправке запроса на сервер на каждом участке кода прописана обработка исключения. В случае ошибки со стороны сервера программа сообщит об этом. Например, при долгом ответе (рисунок 25).

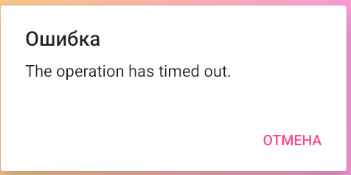


Рисунок 25 – Вывод сообщения об ошибке связи с сервером

Код обработки в такой ситуации выглядит следующим образом (рисунок 26).

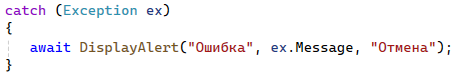


Рисунок 26 – Код обработки ошибки связи с сервером

При попытке отправить сообщение, без какого-либо текста система выдаст следующие оповещение (рисунок 27).

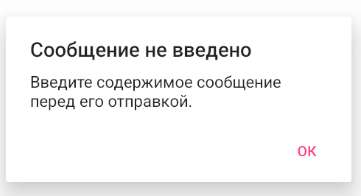


Рисунок 27 – Вывод сообщения о пустом содержимом

При попытке сохранения изменения о пользователе, когда заполнены не все поля система так же сообщит об ошибке (рисунок 28).

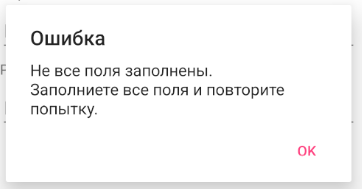


Рисунок 28 – Вывод сообщения о не верном заполнении

# 3 Организационно-экономическая часть

# 3.1 Расчет сметы затрат на разработку программы

# 3.1.1 Составление проекта выполнения работ

Для расчета сметы затрат составлен проект выполнения работ по созданию программы. Он представляет собой перечень мероприятий, которые необходимо выполнить, чтобы разработать и внедрить программу мессенджер для мобильных устройств «DevIm». Работы перечислены в требуемой последовательности с установленной продолжительностью каждого этапа, данные в таблице 2.

Таблица 2 – Проект выполнения работ по созданию программы.

|  |  |
| --- | --- |
| **Наименование этапов** | **Продолжительность, дни** |
| Получение задания на разработку программы | 1 |
| Сбор и изучение требований заказчика, анализ предметной области | 3 |
| Проектирование концептуальной модели | 1 |
| Проектирование графического интерфейса программного приложения | 2 |
| Разработка функциональных возможностей программы | 8 |
| Тестирование программы | 2 |
| Отладка программного продукта и встреча с заказчиком | 3 |
| Эксплуатация, сдача проекта заказчику | 1 |

Общие затраты времени на разработку программы определены как сумма продолжительности работ и составляют 21 день.

Рассчитываем затраты на эксплуатационные материалы, исходные данные представлены в таблице 3, в ней же и результаты расчетов.

Таблица 3 – Затраты на эксплуатационные материалы.

|  |  |  |  |
| --- | --- | --- | --- |
| **Наименование материала** | **Количество** | **Цена, руб.** | **Сумма, руб.** |
| Флешка | 1 | 159 | 159 |
| Интернет (дней) | 21 | 15 | 315 |
| Тетрадь для макетов | 1 | 21 | 21 |
| Гелевая ручка | 2 | 15 | 30 |
| Итого |  |  | См = 525 |

См – сумма затрат на эксплуатационные материалы.

Расчёт стоимости электроэнергии.

ЭВМ будет использоваться 21 дней по 8 часов в день, то есть 168 часов.

Исходные данные:

* потребляемая мощность – 0.3 кВт/ч;
* время работы на ЭВМ – 168 ч;
* тариф по электроэнергии – 3.36 руб. /кВт

Рассчитываем стоимость электроэнергии (Сэл.) по формуле:

Сэл. = P × tраб. × Ц, руб.,

где P – потребляемая мощность, кВт/ч;

tраб. – время работы на ЭВМ, ч;

Ц – цена за 1 кВт/ч, руб.

Сэл. = 0.3 × 168 × 2,73 = 137,59 руб.

Рассчитываем сумму материальных затрат (См.з.):

См.з. = См. + Сэл., руб.,

См.з. = 137,59 + 525 = 662,59 руб.

Расчет заработной платы программиста.

Исходные данные:

* время работы над программой, 21 дн.;
* должностной оклад за месяц, 20500 руб.;
* количество рабочих дней за месяц, 22 дн.;
* процент премии, 35 %;
* районный коэффициент, 1,15.

Рассчитываем заработную плату по тарифу:

где Зпл.по тар.- заработная плата по тарифу, руб;

T – время работы над программой, дн.;

ДО – должностной оклад за месяц, руб.;

tм-ц – количество дней за месяц, дн.

Рассчитываем сумму премии:

где ∑прем. – сумма премии, руб.;

%Пр. – процент премии.

Рассчитываем сумму зарплаты за отработанное время:

. руб.

Рассчитываем сумму зарплаты с районным коэффициентом:

,

где с районным коэффициентом, руб.,

РК – районный коэффициент, 1,15

Для расчета очередного отпуска определяем дневную зарплату (Дн.з.пл.):

Исходные данные:

* + дневная зарплата, руб.;
  + средняя продолжительность отпуска, 20 дн.;
  + годовой фонд рабочего времени, 247 дн.;
  + время работы над программой, 21 дн.

Рассчитываем сумму оплаты за отпуск:

где ∑оп.от. – сумма оплаты отпуска, руб.;

tот. – продолжительность отпусков, дн.;

Фр.в. – годовой фонд рабочего времени, дн.;

Т – время работы над программой, дн.;

Дн.з.пл. – дневная заработная плата, руб.;

Рассчитываем сумму затрат на оплату труда (Зот.):

, руб.

Рассчитываем амортизационные отчисления за год, исходные данные и результаты расчётов в таблице 4.

Таблица 4 - Амортизационные отчисления за год

|  |  |  |  |
| --- | --- | --- | --- |
| **Наименование основных производственных фондов** | **Стоимость ОПФ, руб.** | **Норма амортизации, %** | **Сумма амортизации, руб.** |
| Системный блок | 31592 | 17 | 5370,64 |
| Монитор | 15279 | 17 | 2597,43 |
| Итого |  |  | Ar = 7968,07 |

Ar - годовая сумма амортизационных отчислений.

Годовая сумма амортизационных отчислений (Ar) рассчитывается по формуле:

где Cn – стоимость первоначальная, руб;

Ha – норма амортизации, %

Рассчитываем годовую сумма амортизационных отчислений системного блока, Aг с.б.:

Рассчитываем годовую сумму амортизационных отчислений монитора, Aг м.:

Рассчитываем сумму амортизационных отчислений за время разработки программы:

где Ar – годовая сумма амортизационных отчислений, руб.;

n – количество дней на разработку программы;

365 – количество дней в году.

Общий процент отчислений в фонды остается прежним по сравнению с предыдущим 2021 годом – 30%, из них:

* 22% в пенсионный фонд;
* 2,9% в фонд социального страхования;
* 5,1% в фонд медицинского страхования.

Накладные расходы — это налоги, сборы, отчисления в специальные фонды, платежи по кредитам, затраты на командировки, оплата услуг связи и другие.

Рассчитываем себестоимость разрабатываемой программы по формуле:

руб.,

где С/С – себестоимость, руб.;

См.з. – материальные затраты, руб.;

З.о.т. – затраты га оплату труда, руб.;

А – амортизационные отчисления, руб.;

От. – отчисления на социальные нужды, руб.;

НР – накладные расходы, руб.

Рассчитываем отчисления на социальные нужды.

Исходные данные:

* затраты на оплату труда -руб.;
* установленный размер отчислений - 30 %.

где %отчисл. – процент отчислений, %

От. – отчисления на социальные нужды, руб.

Рассчитываем накладные расходы (80% от основной заработной платы)

Рассчитываем себестоимость.

Исходные данные:

См.з – материальные затраты, 662,59 руб.;

З.о.т. – затраты на оплату труда, руб.;

А – амортизационные отчисления, руб.;

От. – отчисления на социальные нужды, руб.;

НР – прочие расходы (накладные расходы), руб.

Определяем структуру себестоимости разрабатываемой программы.

Структура – процентное содержание элементов затрат к общей сумме себестоимости.

Таблица 5 – Структура себестоимости

|  |  |  |
| --- | --- | --- |
| **Наименование статей затрат** | **Сумма, руб.** | **Структура, %** |
| Материальные затраты |  | 0,99 |
| Затраты на оплату труда |  | 45,51 |
| Отчисления на социальные нужды |  | 16,39 |
| Амортизационные отчисления |  | 0,69 |
| Накладные расходы |  | 36,42 |
| Итого: |  | 100 |

Вывод: самую большую долю затрат на разработку программы составляет затраты на оплату труда, а самую малую долю – амортизационные отчисления, так как основная работа происходит на компьютере, которому требуется только электроэнергия для функционирования и выполнения своих функций.

# 3.2 Расчет экономической эффективности внедрения программы

Экономическая эффективность – это соотношение результатов с затратами.

Если разрабатываемая программа пользуется спросом, то возможна ее реализация. В этом случае эффективность определяется возможной прибылью от реализации разработанной программы.

Для этого необходимо установить цену на разработанную программу.

При установлении цены используют полную сбытовую себестоимость товарной продукции, она включает производственную себестоимость и внепроизводственные расходы.

Внепроизводственные (коммерческие) расходы включают затраты, связанные с реализацией продукции (расфасовка, упаковка, отгрузка, реклама, маркетинговые исследования, комиссионные; хранение, транспортировка), а также различного рода отчисления и платежи.

Полная себестоимость рассчитывается по формуле:

где С/Ск.– полная себестоимость единицы продукции, руб;

С/С – себестоимость единицы продукции производственная, руб;

Вн.р. – внепроизводственные расходы (7-10% от производственной себестоимости), руб.

Исходные данные:

* внепроизводственные расходы – руб. (7 % от производственной себестоимости);
* производственная стоимость – руб.

Оптовую цену предприятия можно определить расчетным путем, используя уровень рентабельности продукции.

Уровень рентабельности продукции устанавливают в зависимости от величины прибыли (20-40% от коммерческой (полной) себестоимости единицы продукции).

Рассчитываем цену разрабатываемой программы по формуле:

где Ц – цена разрабатываемой программы, руб;

С/Ск. – себестоимость коммерческая (полная) разрабатываемой программы, руб.;

Pnp. – рентабельность продукции, %.

Исходные данные:

* рентабельность продукции – 21 %;
* полная себестоимость продукции – руб.

Расчет прибыли от реализации программы.

Прибыль от реализации продукции (работ, услуг) определяется как разница между выручкой от реализации продукции (работ, услуг) в действующих ценах (без налога на добавленную стоимость, акцизного налога) и затратами на ее производство и реализацию, включаемыми в себестоимость продукции (работ, услуг).

Рассчитываем прибыль от реализации разрабатываемой программы:

где Пр. – прибыль от реализации программы, руб.

На основе проведенного анализа экономической эффективности, внедрение данного ПО является экономически выгодным и позволит сэкономить средства компании.

# 4 Мероприятия по технике безопасности и охране окружающей среды

# 4.1 Организация безопасности труда на предприятии

На предприятиях используются различные меры по обеспечению безопасности труда, включая:

1. Программы по обучению и профессиональной подготовке сотрудников. Важно обучать персонал правильным методам работы с компьютерами и другими техническими устройствами, а также по управлению информацией и данными.

2. Использование эргономической мебели и оборудования. Это включает в себя поддержку правильной осанки и удобную рабочую среду для сотрудников, уменьшая риск возникновения мускулоскелетных проблем.

3. Регулярные перерывы и физические упражнения. Для предотвращения усталости и стресса у сотрудников важно обеспечить им возможность регулярно отдыхать и проводить физические упражнения.

4. Информационная безопасность. Защита компьютерных систем от вирусов, хакерских атак и утечек данных является одной из важных мер безопасности в IT предприятиях.

5. Регулярные проверки оборудования и инфраструктуры. Проведение проверок и обслуживание оборудования помогает предотвратить возможные аварийные ситуации и обеспечить безопасность персонала.

Кроме того на предприятия имеются специальные процедуры и планы для управления чрезвычайными ситуациями, например пожар или эвакуация здания..

# 4.2 Техника безопасности при работе с вычислительной техникой

Основные правила безопасности:

* 1. При проведении работ с вычислительной техникой необходимо обеспечивать ее заземление для предотвращения статического электричества.
  2. Использование стабилизаторов напряжения и защитных устройств является обязательным для предотвращения повреждения оборудования от скачков напряжения.
  3. Подключение и отключение устройств должно осуществляться исключительно путем удержания корпуса, а не шнура питания.

Хранение и обслуживание:

* 1. Все устройства должны храниться в сухом и прохладном месте, защищенном от влаги, пыли и механических воздействий.
  2. Регулярная очистка и обслуживание оборудования от пыли должны проводиться согласно рекомендациям производителя.

Информационная безопасность:

* 1. Подключение к сети и интернету должно осуществляться только через надежные и защищенные сети Wi-Fi с установленными паролями.
  2. Резервное копирование важных данных является обязательным для предотвращения потерь информации при сбое оборудования.

Обучение и контроль:

* 1. Персонал должен пройти обучение по правилам работы с вычислительной техникой и безопасности информации.
  2. Руководители обязаны контролировать соблюдение настоящего Устава и принимать меры к устранению нарушений.

# Заключение

В ходе выполнения дипломного проекта была достигнута поставленная цель: «Реализация системы корпоративного коммуницирования для ООО «ПТ-СОФТ»».

При работе над дипломным проектом были выполнены следующие задачи:

* 1. Изучить требования заказчика и провести анализ предметной  
     области.
  2. Проектировать план создания программного решения.
  3. Разработать программные модули решения.
  4. Разработать интерфейс приложения.
  5. Разработать программные модули решения.
  6. Протестировать созданную систему.
  7. Рассчитать стоимость программного решения.
  8. Запустить сервер в локальной сети.
  9. Оформить техническую документацию в виде пояснительной  
     записки на дипломный проект.

Итогом дипломного проекта стала система корпоративного коммуницирования «Devim».

# Список использованных источников

1. Бабаш, А.В., Информационная безопасность. [Текст]: Лабораторный практикум / Бабаш, А.В.– М.: КноРус, 2019 г. – 136 c.: ил.
2. Гаврилов, М.В. М.В. Информатика и информационные технологии: [Текст]: Учебник / Гаврилов М.В., Климов В.А.– Люберцы: Юрайт 2019 г. – 383 c
3. Кулаков, К.А. Основы тестирования программного обеспечения: учебное электронное пособие для обучающихся Института математики и информационных технологий [Текст] / К.А. Кулаков, В.М. Димитров. – Петрозаводск.: Издательство ПетрГУ, 2019 г. – 56 с.: ил.
4. Фленов, М.Е. Библия C#: для программистов. [Текст] / М.Е. Фленов. 2-е изд., перераб и доп. – СПБ.: БХВ-Петербург, 2019 г. – 560 с.: ил.
5. ChatApp.Mobile/ Брахим Гербу // GitHub – 2021 - АльфаТехстудии – Xamarin-Forms – URL: https://github.com/AlphaTechstudios/Xamarin-Forms/tree/master/ChatApp.Mobile (дата обращения 03.04.24).
6. LineBreakMode Перечисление / Microsoft // Microsoft – 2023 – Learn - .Net – Обозреватель API – Xamarin.Forms – URL: https://learn.microsoft.com/ru-ru/dotnet/api/xamarin.forms.linebreakmode?view=xamarin-forms (дата обращения 03.05.23).
7. ListView.ScrollTo Метод / Microsoft // Microsoft – 2023 – Learn –.NET – Обозреватель API – Xamarin.Forms – ListView – Методы – URl: https://learn.microsoft.com/ru-ru/dotnet/api/xamarin.forms.listview.scrollto?view=xamarin-forms (дата обращения 04.05.23).
8. NavigationPage Класс / Microsoft // Microsoft – 2023 – Learn - .Net – Обозреватель API – Xamarin.Forms – URL: https://learn.microsoft.com/ru-ru/dotnet/api/xamarin.forms.navigationpage?view=xamarin-forms (дата обращения 05.05.24).
9. WPF – Система авторизации и регистрации [электронный ресурс] you-hands.ru URL: https://you-hands.ru/2018/08/31/wpf-sistema-avtorizacii-i-registracii/?ysclid (Дата обращения 03.03.2023).
10. Xamarin Forms, динамическое добавление нового элемента в представление списка/ Али Аль-Синан // Stack Overflow – 2019 - URL: https://stackoverflow.com/questions/52245998/xamarin-forms-add-new-item-to-listview-dynamically (дата обращения 09.04.24)
11. Xamarin ListView Получить выбранный элемент / Игны // Stack Overflow – 2019 – URL: https://stackoverflow.com/questions/54281705/xamarin-listview-get-selected-item (дата посещения 10.04.24)
12. Xamarin.Forms Выбора / Microsoft // Microsoft – 2023 – Learn – Xamarin – URL: https://learn.microsoft.com/ru-ru/xamarin/xamarin-forms/user-interface/picker/ (дата обращения 12.04.24).
13. Xamarin.Forms Запись / Microsoft // Microsoft – 2023 – Learn – Xamarin – URL: https://learn.microsoft.com/ru-ru/xamarin/xamarin-forms/user-interface/text/entry (дата обращения 13.04.24).
14. Xamarin.Forms Метки / Microsoft // Microsoft – 2023 – Learn – Xamarin – URL: https://learn.microsoft.com/ru-ru/xamarin/xamarin-forms/user-interface/text/label (дата обращения 14.04.24).
15. Заставка / Microsoft // Microsoft – 2023 – Learn – Xamarin – URL: https://learn.microsoft.com/ru-ru/xamarin/android/user-interface/splash-screen (дата обращения 15.04.24).
16. Изображения в Xamarin.Forms / Microsoft // Microsoft – 2023 – Learn – Xamarin – URL: https://learn.microsoft.com/ru-ru/xamarin/xamarin-forms/user-interface/images?tabs=windows (дата обращения 16.04.24).
17. Инструкции по установке сертификата сервера в службах IIS / Microsoft // Microsoft – 2023 – Learn – .Net – .Net Framework – Windows Communication Foundation – Образцы Windows Communication Foundation (WCF) – URL: https://learn.microsoft.com/ru-ru/dotnet/framework/wcf/samples/iis-server-certificate-installation-instructions (дата обращения 16.04.24).
18. Как стилизовать элементы ListView/ Джэфф ЛаФэй, Томас Левек // Stack overflow – 2019 – URL: https://stackoverflow.com/questions/4474436/how-do-i-style-listview-items (дата обращения: 17.04.24).
19. Кисти Xamarin.Forms: линейные градиенты / Microsoft // Microsoft – 2023 – Learn – Xamarin – URL: https://learn.microsoft.com/en-us/xamarin/xamarin-forms/user-interface/brushes/lineargradient (дата обращения 18.04.24).
20. Кнопка градиента в Xamarin Forms / Майнак Чоудхури // Stack Overflow – 2019 - URL: https://stackoverflow.com/questions/48834030/gradient-button-in-xamarin-forms (дата посещения 19.04.24)
21. Навигация в Xamarin.Forms / Microsoft // Microsoft – 2023 – Learn – Xamarin – URL: https://learn.microsoft.com/ru-ru/xamarin/xamarin-forms/app-fundamentals/navigation/ (дата обращения 20.04.24).
22. Практическая работа "Авторизация и регистрация пользователя в WPF-приложении к базе данных MS SQL Server" [электронный ресурс]: infourok.ru URL: https://infourok.ru/prakticheskaya-rabota-avtorizaciya-i-registraciya. (дата обращения 21.04.24).
23. Службы размещения / Microsoft // Microsoft – 2023 –– Learn –.NET – .NET Framework – Windows Communication Foundation – URL: https://learn.microsoft.com/ru-ru/dotnet/framework/wcf/hosting-services (дата обращения 2.05.23).
24. Microsoft.com : образовательная система: сайт – США – 2016- URL: https://learn.microsoft.com
25. Stackoverflow.com: система вопросов и ответов о программировании: сайт – США – 2008 – URL: https:// stackoverflow.com