**0МИНИСТЕРСТВО ЭНЕРГЕТИКИ, ПРОМЫШЛЕННОСТИ И СВЯЗИ СТАВРОПОЛЬСКОГО КРАЯ**

Государственное бюджетное профессиональное образовательное учреждение «Ставропольский колледж связи имени Героя Советского Союза В.А. Петрова» (ГБПОУ СКС)

Цикловая комиссия вычислительной техники,

экономики и менеджмента

**КУРСОВАЯ РАБОТА**

**по ПМ.02 «Осуществление интеграции программных модулей»**

**МДК 02.01 «Технология разработки программного обеспечения»**

Тема: Разработка информационной системы заявки на обслуживание ремонта компьютерной техники

Выполнил

00Золотарев Алексей Владимирович

(Ф. И. О.,

3 курс группа ИП-202

курс, № группа)

|  |  |
| --- | --- |
|  | Руководитель |
|  | Лукьянова Светлана Павловна |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Ставрополь, 2024

**«Ставропольский колледж связи**

**имени Героя Советского Союза В.А. Петрова»**

УТВЕРЖДАЮ

Председатель цикловой комиссии

\_\_\_\_\_\_\_\_\_\_\_\_\_\_/И.В. Ерёмина

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г.

**ЗАДАНИЕ**

**для курсовой работы (проекта)**

Студенту *Фамилия Имя Отчество, курс, группа*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель Лукьянова Светлана Павловна\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Тема \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Разделы, подразделы и их содержание** | **Сроки выполнения** |
| 1 | Введение | 00.00.2024 |
| 2 | Теоретические основы разработки программного обеспечения | 00.00.2024 |
| 3 | Техническое задание на разработку | 00.00.2024 |
| 4 | Функциональная модель информационной системы | 00.00.2024 |
| 5 | Объектно-ориентированное проектирование системы | 00.00.2024 |
| 6 | Создание информационной системы | 00.00.2024 |
| 7 | Тестирование программного продукта | 00.00.2024 |
| 8 | Техническая документация | 00.00.2024 |
| 9 | Заключение | 00.00.2024 |
| 10 | Список использованных источников | 00.00.2024 |
| 11 | Приложения | 00.00.2024 |

Преподаватель-руководитель курсовой работы (проекта)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**/С.П. Лукьянова/

**СОДЕРЖАНИЕ**

[Введение 3](#_Toc156296148)

[1 Теоретические основы разработки программного обеспечения 4](#_Toc156296149)

[2 Техническое задание на разработку 5](#_Toc156296150)

[3 Функциональная модель информационной системы 6](#_Toc156296151)

[4 Объектно-ориентированное проектирование системы 8](#_Toc156296152)

[4.1 Модели вариантов использования системы 8](#_Toc156296153)

[4.2 Диаграмма классов 8](#_Toc156296154)

[4.3 Диаграммы деятельности 8](#_Toc156296155)

[4.4 Диаграмма последовательности 8](#_Toc156296156)

[4.5 Диаграмма кооперации 9](#_Toc156296157)

[4.6 Диаграмма состояний 9](#_Toc156296158)

[5 Создание информационной системы 10](#_Toc156296159)

[5.1 Разработка интерфейса программного продукта 10](#_Toc156296160)

[5.2 Разработка программного кода системы 10](#_Toc156296161)

[5.3 Разработка базы данных 10](#_Toc156296162)

[5.4 Справочная система 10](#_Toc156296163)

[6 Тестирование программного продукта 11](#_Toc156296164)

[7 Техническая документация 12](#_Toc156296165)

[ЗАКЛЮЧЕНИЕ 13](#_Toc156296166)

[Приложение А 15](#_Toc156296167)

[Приложение Б 16](#_Toc156296168)

[ПРИЛОЖЕНИЕ В 17](#_Toc156296169)

# **Введение**

Информационная система “Заявки” предназначена для составления и управления заявками на обслуживание и ремонт компьютерной техники.

В результате анализа задания и исследования предметной области для разработки был выбран язык C# на платформе Windows Presentation Foundation (.NET).

Основными требования к проекту были выведены:

* возможность регистрации клиентов;
* функции добавления новых, удаления и/или изменения существующих заявок;
* хранение заявок в базе данных SQLServer/MySQL;
* наличие графического пользовательского интерфейса;
* разделение прав между оператором ИС и клиентом.

Информационная система, будет реализованная с использованием цифровых технологий хранения и обработки данных, и современных языков программирования (C#/SQL).

Для создания информационной системы, соответствующей требованиям пользователя и обеспечивающая должную конкурентоспособность, необходимо провести анализ различных аспектов отрасли.

Важно, чтобы информационная система удовлетворяла потребностям пользователей, интерфейс был интуитивно понятен, а программа могла максимально автоматизировать бизнес-процессы для наиболее эффективной работы.

Цель курсового проекта: используя современные средства проектирования и разработки реализовать все этапы жизненного цикла программных продуктов.

# **Теоретические основы разработки программного обеспечения**

Информационная система “Заявки” предназначена для хранения и обработки заявок на обслуживание и ремонт компьютерной техники. Программа должна иметь интуитивно понятный пользователю интерфейс, возможность эффективно обрабатывать большие объемы данных и автоматизировать процессы хранения и обработки информации.

Система включает в себя:

* Серверное приложение для управления базой данных (SQLServer/MySQL) с необходимой для этого инфраструктурой;
* Клиентское/терминальное приложение с графическим пользовательским интерфейсом, возможностью регистрации и авторизации пользователя, протоколами взаимодействия с серверной частью приложения.

Пользователю должна быть предоставлена возможность регистрации в системе используя такие данные как: email, логин и пароль. А также возможность авторизоваться в системе используя эти же данные.

Из принципов безопасности права и возможности пользователей будут ограничены возможностью создать заявку на ремонт. Удаление и изменение заявок будет производится со стороны оператора ИС.

При разработке информационной системы необходимо учитывать требования надежности и эффективности.

# **Техническое задание на разработку**

Техническое задание представляет собой документ, который включает в себя требования к разработке, функциональности, содержанию и другим аспектам проекта, определяет цели и задачи, а также конкретные требования.

В техническом задании описываются системные, функциональные и эксплуатационные требования к системе, её структура и структура используемых баз данных.

Также техническое задание включает в себя расписание этапов разработки, испытаний и приёмки, их состав, порядок исполнения и примерное время исполнения.

Функциональные требования описывают необходимые функции для разрабатываемой системы, их назначение, а также наличие или отсутствие, графического пользовательского интерфейса, для взаимодействия с системой/базой данных.

Техническое задание на разработку программного продукта «Информационная система «Заявки» приведено в приложении А.

# **Функциональная модель информационной системы**

Диаграммы позволяют графически представить функционал информационной системы, отобразить возможности программы и взаимодействие модулей.

Контекстная диаграмма прежде всего позволяет быстро, кратко и ёмко описать назначение и границы системы.

Ramus – это программное обеспечение, относящееся к CASE-средствам, является инструментом для анализа и моделирования IDEF0 и DFD диаграмм.

Контекстная диаграмма ИС «Заявки» показана на рисунке 1. Функциональная диаграмма первого уровня приведена на рисунке 2. На рисунке 3 показана функциональная диаграмма второго уровня для функции «Авторизация/Регистрация».

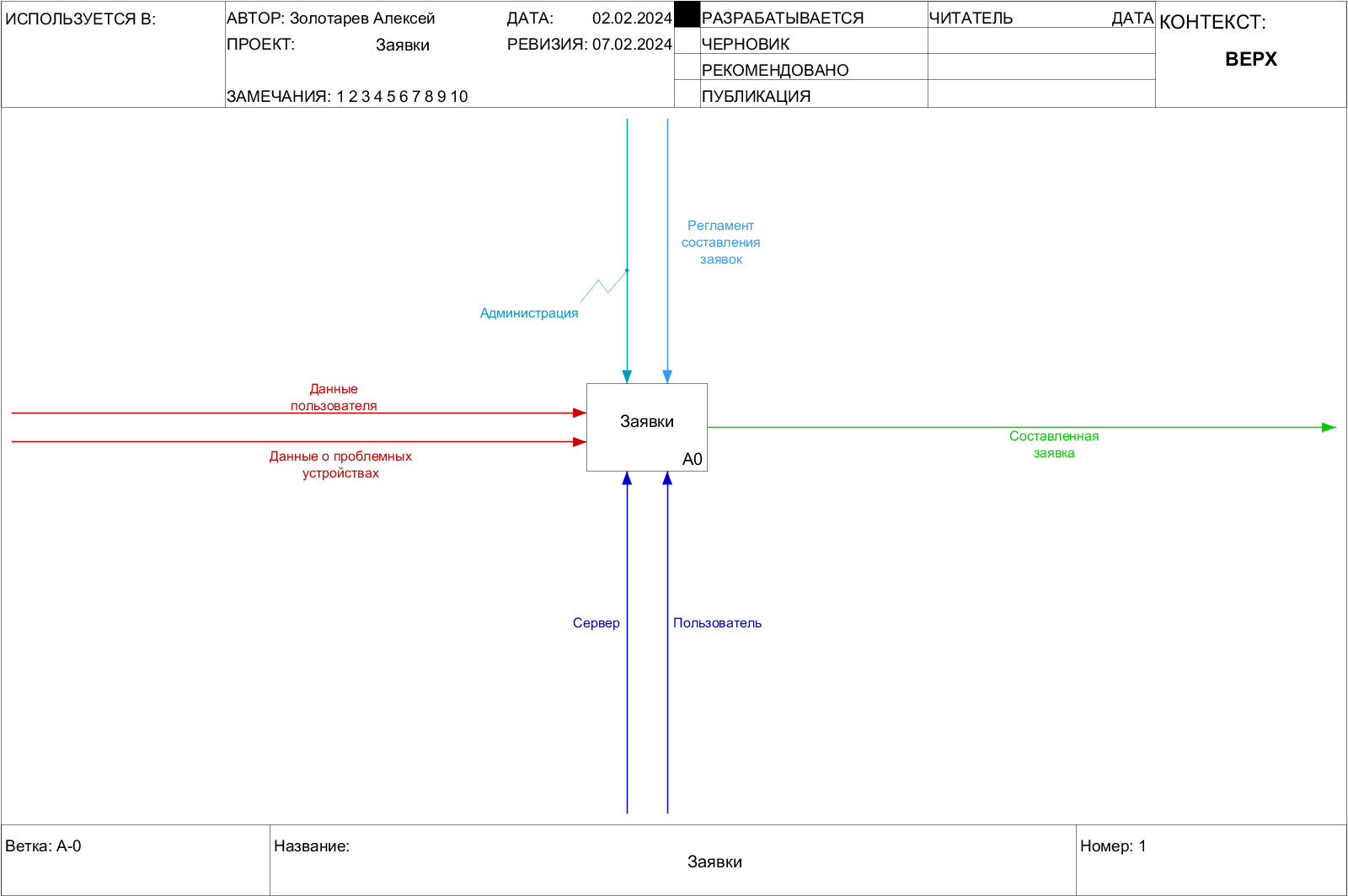


Рисунок 1 – Контекстная диаграмма информационной системы

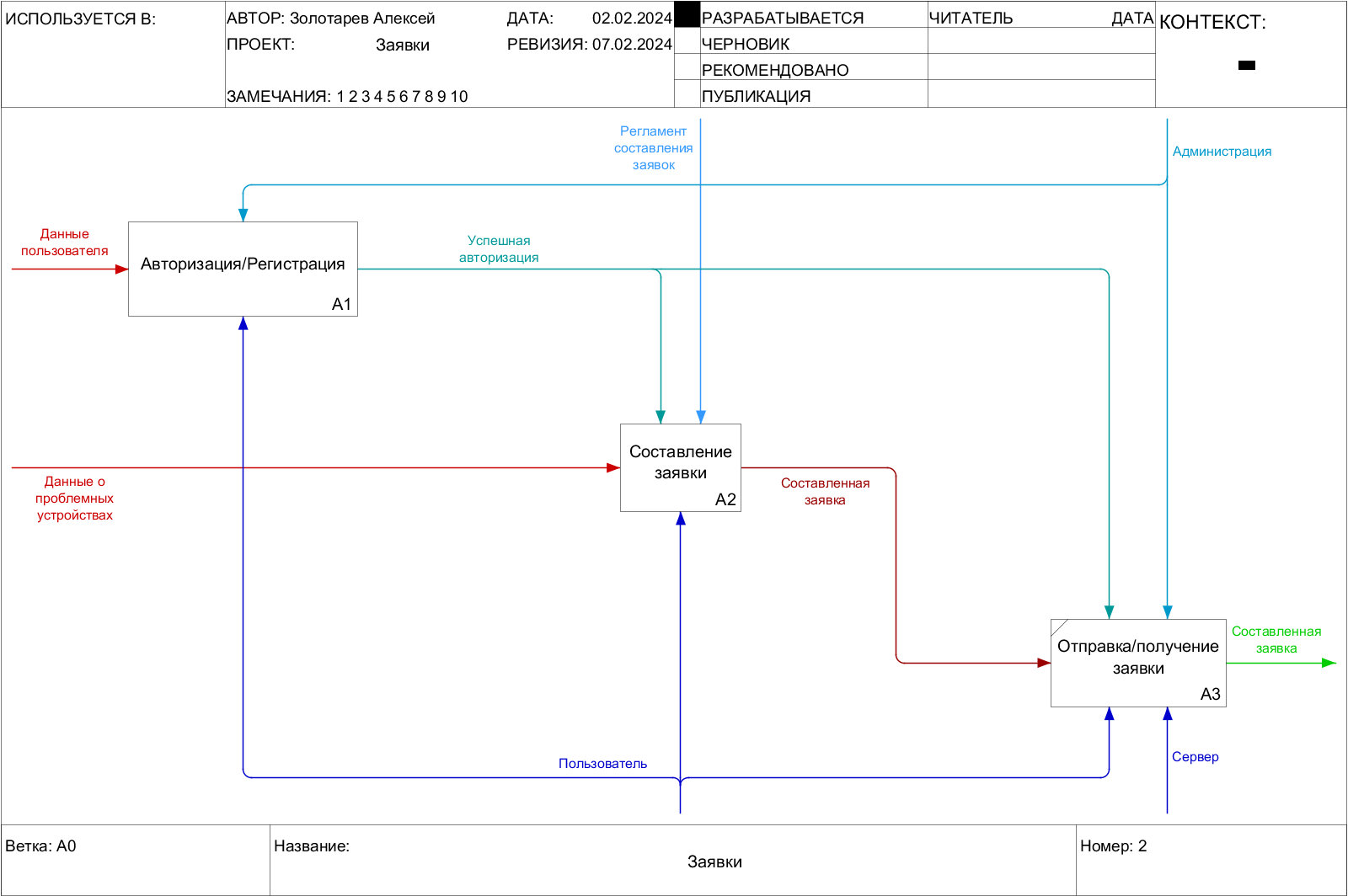


Рисунок 2 – Функциональная диаграмма первого уровня

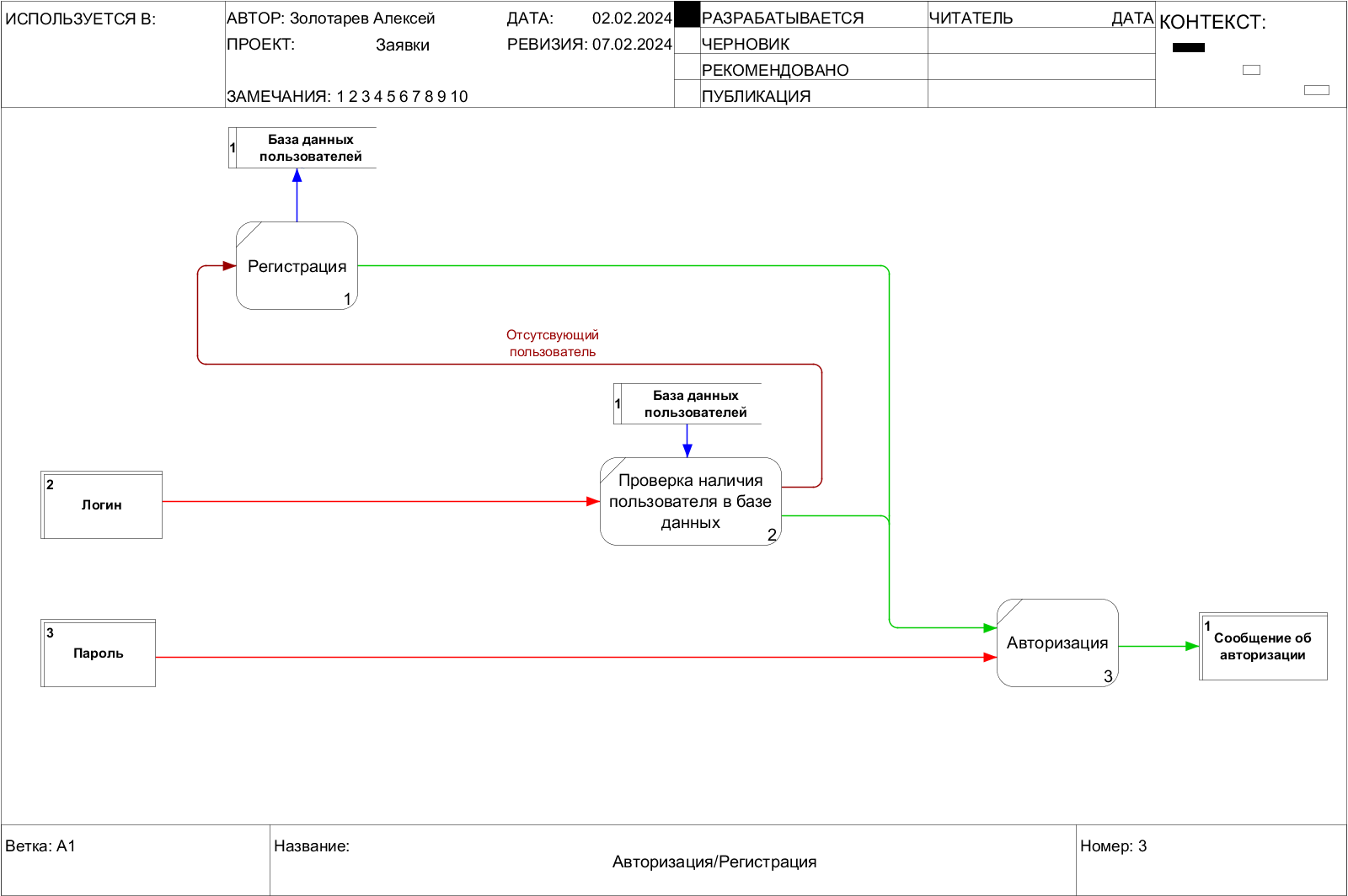


Рисунок 3 – DFD диаграмма для информационной системы

# **Объектно-ориентированное проектирование системы**

## 4.1 Модели вариантов использования системы

В ходе анализа для проектируемой информационной системы было выделено 2 действующих лица. Для каждого из них были выделены прецеденты.

Полученная диаграмма вариантов использования ИС «Заявки» показана на рисунке 4.



Рисунок 4 – Диаграмма вариантов использования информационной системы

Каждый пользователь имеет возможность авторизироваться/зарегистрироваться в системе и составить заявку. Администрация имеет дополнительные возможности удаления и изменения заявок.

## 4.2 Диаграмма классов

Основной диаграммой проекта является диаграмма классов. Статистическая модель организации данных. Основная диаграмма классов представлена на рисунке 5.

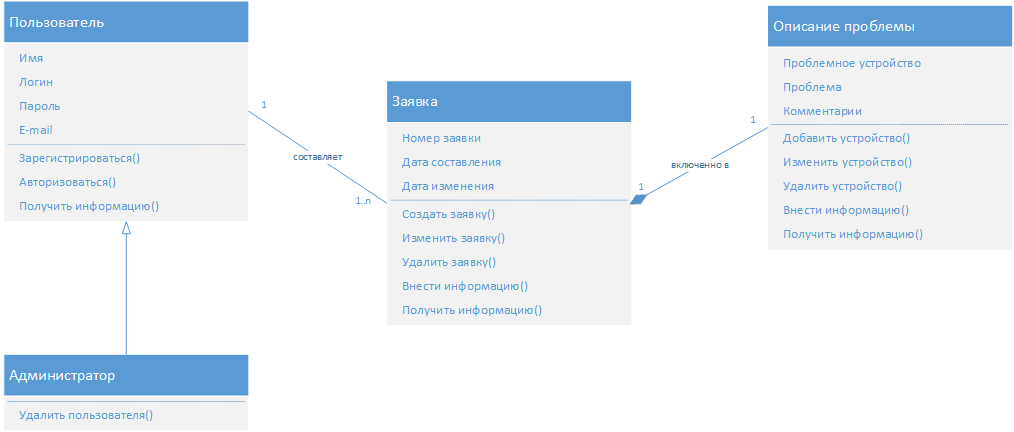


Рисунок 5 – Диаграмма классов

На диаграмме классов представлена структура организации данных приложения/базы данных, и взаимодействия между ними.

Каждый пользователь может зарегистрироваться и авторизироваться, но только пользователи с правами администратора могут удалять других пользователей.

Пользователи самостоятельно составляют заявку, один пользователь может составить несколько заявок от своего имени.

В каждой заявке имеется описание проблемы, включающее в себя поля для ввода информации об одном или нескольких устройствах и их неисправностях.

Используя систему, можно получить информацию по каждому из представленных классов.

## 4.3 Диаграммы деятельности

Диаграмма деятельности, составленная для информационной системы, представлена на рисунке 6.

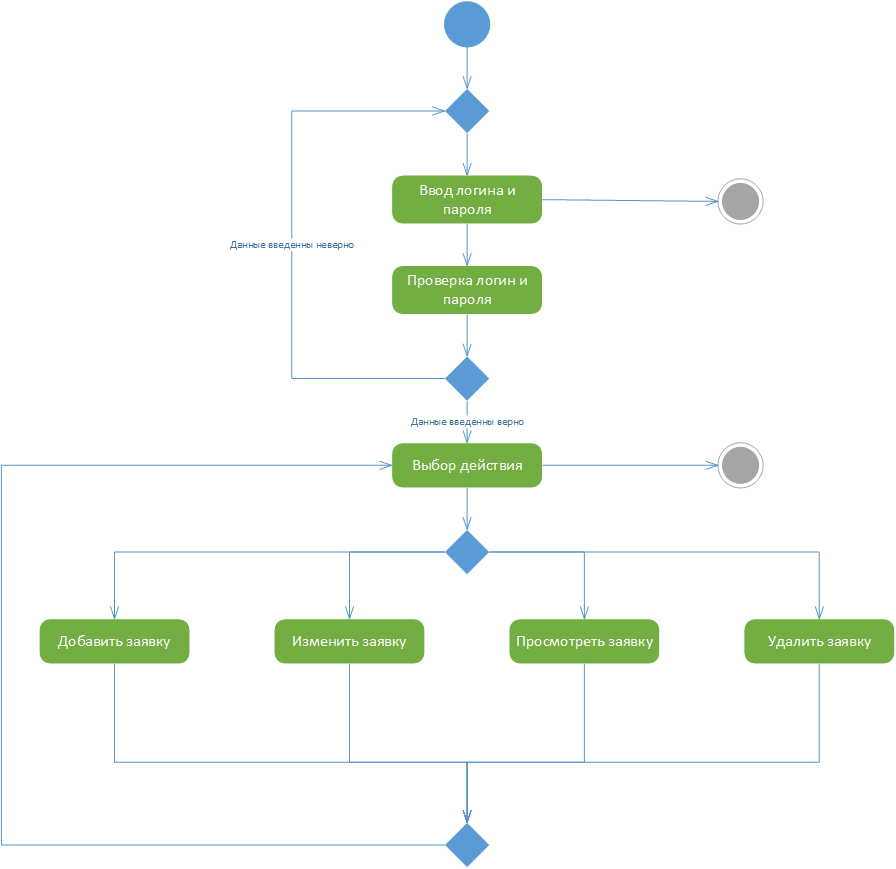


Рисунок 6 – Диаграмма деятельности

На диаграмме деятельности показаны возможные варианты, последовательность и результаты действий, выполняемых при работе с приложением.

Выход из приложения может осуществляться либо на этапе ввода логина и пароля, либо во время непосредственной работы с приложением.

## 4.4 Диаграмма последовательности

Диаграмма последовательности, составленная для варианта использования «Создание заявки», представлена на рисунке 7.

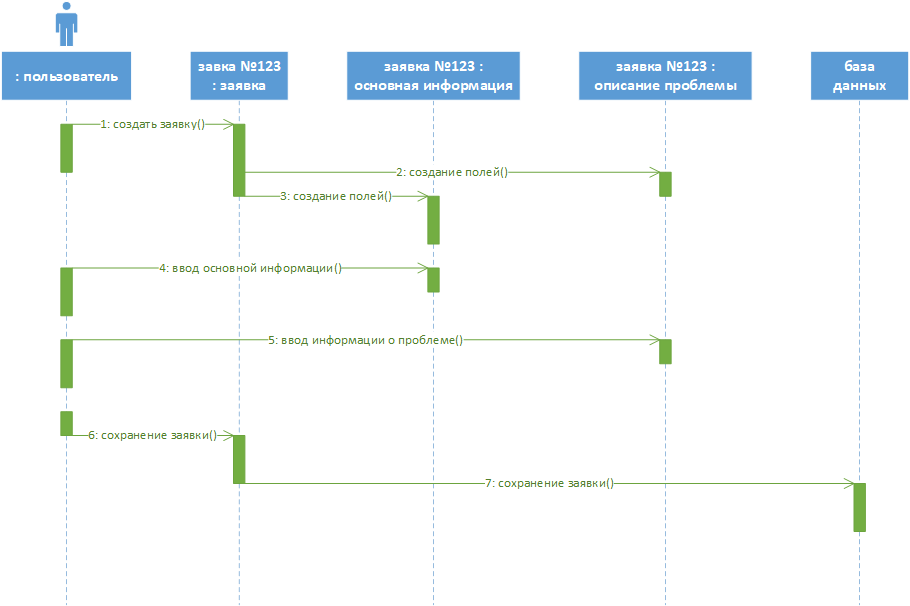


Рисунок 7 – Диаграмма последовательности

На диаграмме последовательности отражена последовательность действий, выполняемая пользователем для создания и сохранения заявки.

## 4.5 Диаграмма кооперации

Диаграмма кооперации, составленная для варианта использования «Создание заявки», представлена на рисунке 8.

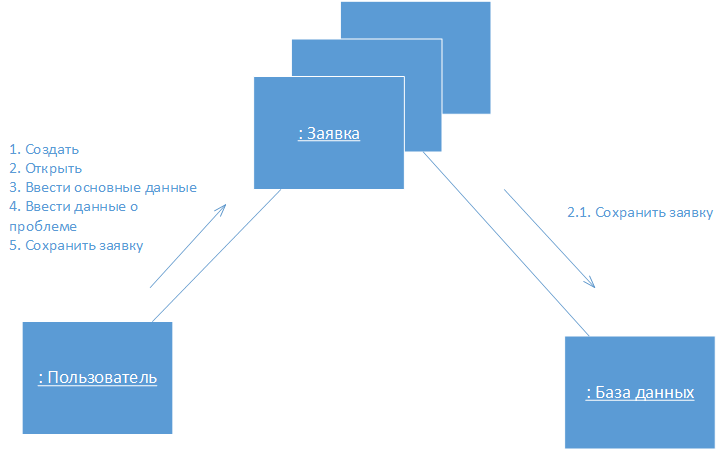


Рисунок 8 – Диаграмма кооперации

На диаграмме кооперации показаны действия, выполняемые системой во время создания и сохранения заявки.

Диаграмма кооперации графически представляет все структурные отношения между объектами, участвующими во взаимодействии.

## 4.6 Диаграмма состояний

Диаграмма состояний, составленная для варианта использования «Создание/закрытие заявки», представлена на рисунке 9.

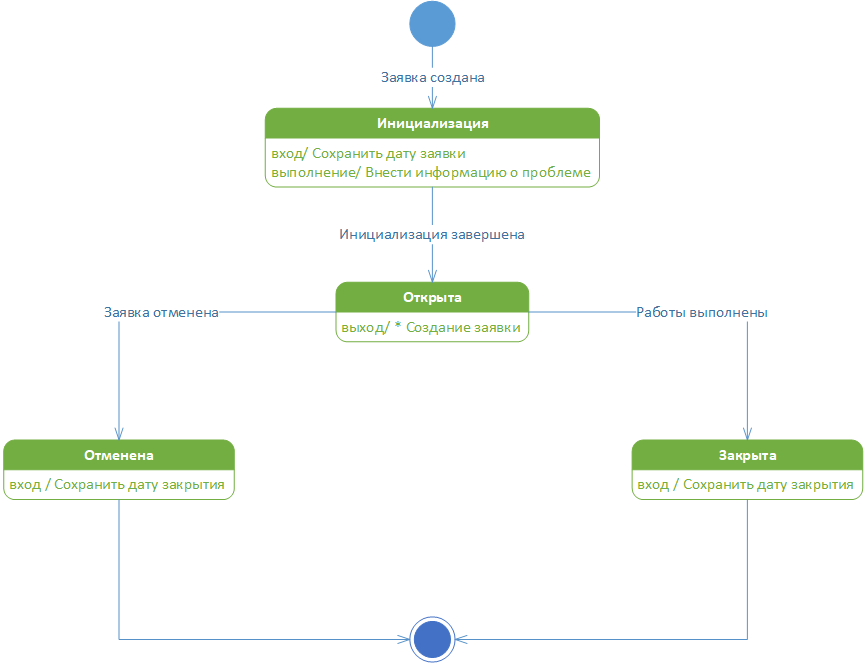


Рисунок 9 – Диаграмма состояний

На диаграмме состояния отражены состояния системе во время создания заявки и возможные варианты ведущие к её закрытию.

# **Создание информационной системы**

## 5.1 Разработка интерфейса программного продукта

В ходе разработке было принято решение использовать в системе графический интерфейс.

Графический интерфейс позволяет облегчить работу с пользователя с системой за счёт:

* Снижения требований к знаниям пользователя;
* Наглядности используемых элементов;
* Расширения возможностей взаимодействия с пользователем.

Графический интерфейс хоть и является сложным в создании и настройке намного превосходит возможности консольного интерфейса.

При этом стоит учитывать, что для выполнения поставленной задачи использование других видов интерфейсов невозможно.

На рисунке 10 представлен интерфейс разработанного программного продукта.

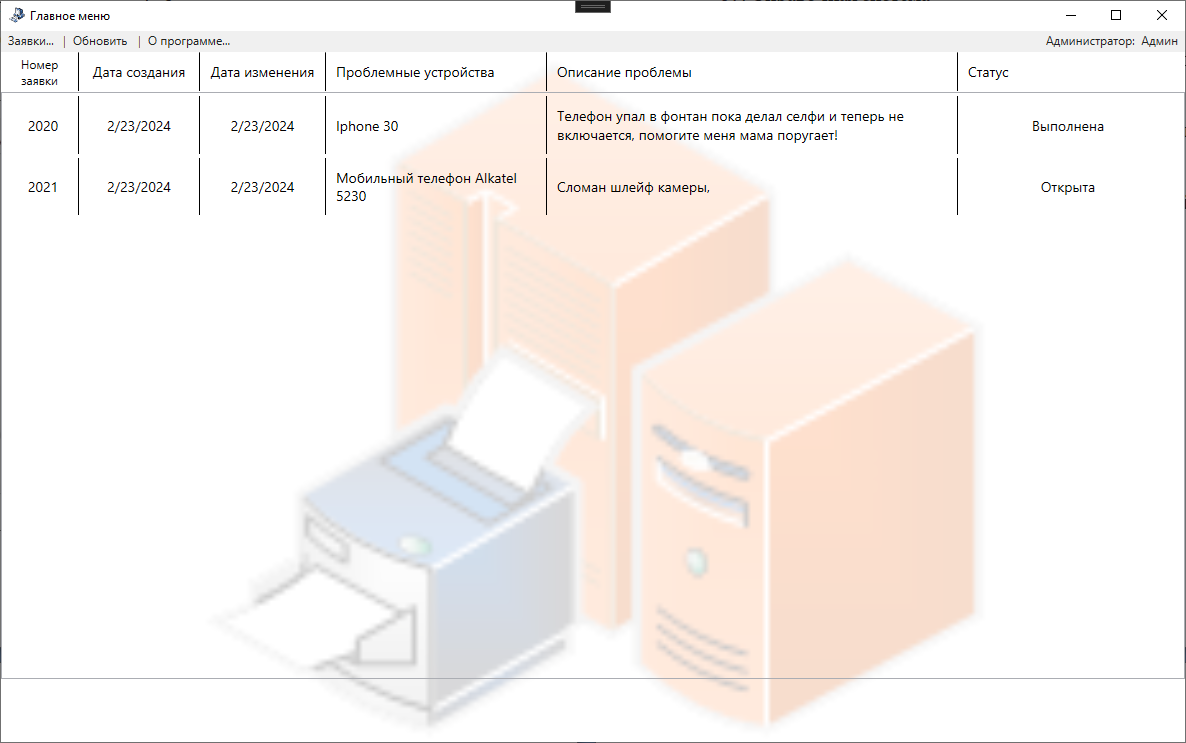


Рисунок 10 – Интерфейс приложения «Заявки»

## 5.2 Разработка программного кода системы

Разработка приложения производилась на платформе .Net Microsoft на языке программирования С#.

C# — объектно-ориентированный язык программирования общего назначения. C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java, это делает его намного более доступным из-за отсутствия надобности в изучении нового синтаксиса.

Являясь высокоуровневым языком, C# даёт намного более широкие возможности в разработке приложений с графическим интерфейсом и работе с базами данных.

Таким образом, C# является наилучшим выбором для данного проекта. Также для разработки был выбран программная платформа Windows Presentation Foundation по причине личных предпочтений и технических знаний.

Исходный код приложения представлен в приложении Б.

## 5.3 Разработка базы данных

База данных была реализована на основе СУБД SQLServer. Структура базы данных велась с использование Transact-SQL и среды разработки Microsoft Visual Studio.

База данных состоит из 4 таблиц:

* Таблица пользователей – USERS;
* Таблица заявок – REQUESTS;
* Таблица ролей пользователей – ROLES;
* Таблица статусов заявок – STATES.

Таблица USERS связанна с таблицей REQUESTS связью «один-ко-многим», таблица USERS связанна с таблицей ROLES связью «один-ко-многим», таблица REQUESTS связанна с таблицей STATES связью «один-ко-многим».

На рисунке 11 представлена ERD-диаграмма базы данных.

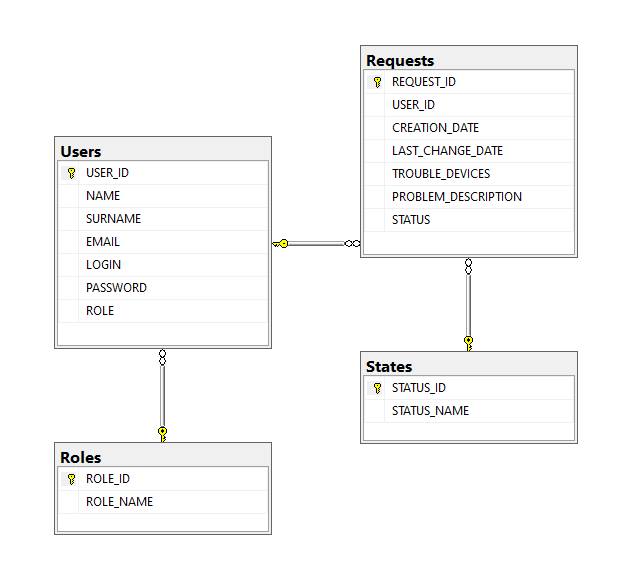


Рисунок 11 – ERD-диаграмма базы данных

## 5.4 Справочная система

Справочная система для приложения была разработана при помощи специализированной программы CHM Builder. Файл справки, представленный на рисунке 11, был прикреплен к разработанному приложению.

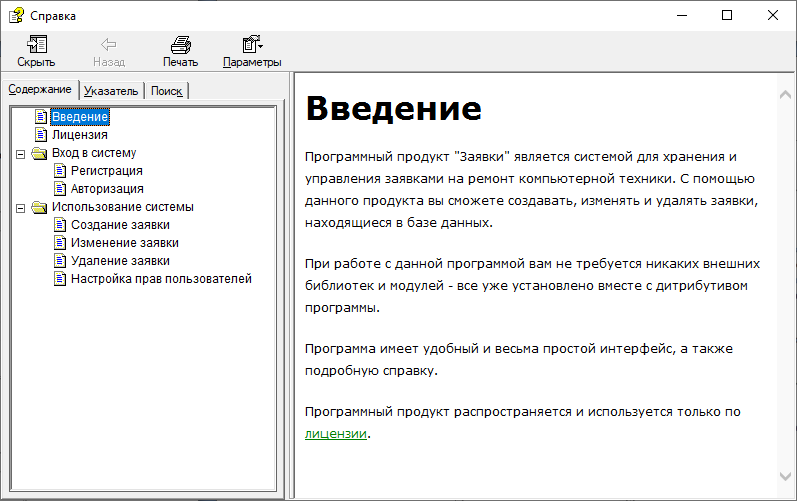


Рисунок 11 – Стартовое окно файла справки

# **Тестирование программного продукта**

В данном разделе описывается процесс ручного модульного функционального тестирования белого ящика, для проверки была выбран модуль авторизации/регистрации.

**Регистрация**

Для успешной регистрации необходимо чтобы в таблицу пользователей была добавлена новая запись с соответствующими данными.

На рисунке 12 представлено окно регистрации приложения.

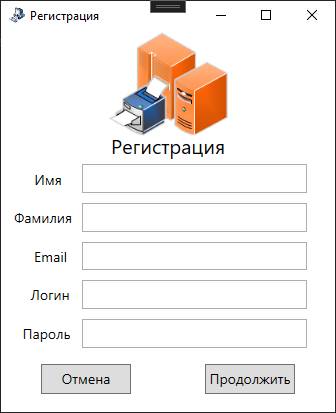


Рисунок 12 – Окно регистрации приложения

Первым делом проверяется «пустая» регистрация - попытка регистрации без ввода информации. В результате пользователю должна быть выдана ошибка, представленная на рисунке 13.

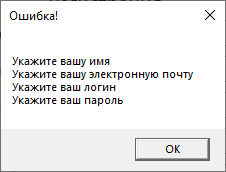


Рисунок 13 – Ошибка при «пустой» регистрации

Следующим шагом проверяем успешную регистрацию, заполнив все необходимые поля, как на рисунке 14.

Вводимые данные:

* Имя – Максим
* Email – [maxim@mail.ru](mailto:maxim@mail.ru)
* Логин – max123
* Пароль – qwerty123

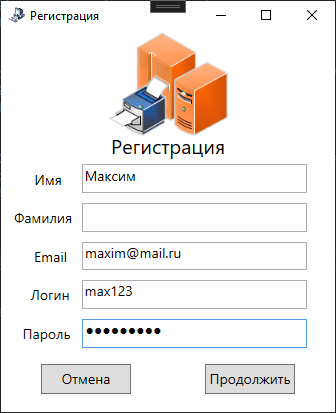


Рисунок 14 – Заполненные поля окна регистрации

После нажатия кнопки «Продолжить» пользователю необходимо пройти проверку CAPTCHA, представленную на рисунке 15.

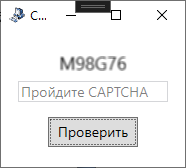


Рисунок 15 – Проверка CAPTCHA

При успешном прохождении проверки будет показано сообщение об успешной регистрации, представленное на рисунке 16.

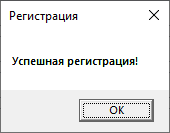


Рисунок 16 – Проверка CAPTCHA

Таким образом, введённые данные были успешно добавлены в таблицу пользователей, представленную на рисунке 17.

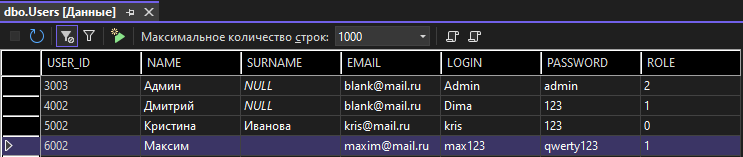


Рисунок 17 – Данные в таблице пользователей

При попытке зарегистрироваться, используя уже существующие данные, пользователю будет показана ошибка, показанная на рисунке 18.

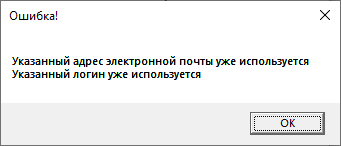


Рисунок 18 – Данные в таблице пользователей

**Авторизация**

Успешно пройдя регистрацию можно использовать уже введённые данные для входа в систему, но перед этим следует проверить правильность работы логики авторизации, окно авторизации представлено на рисунке 19.

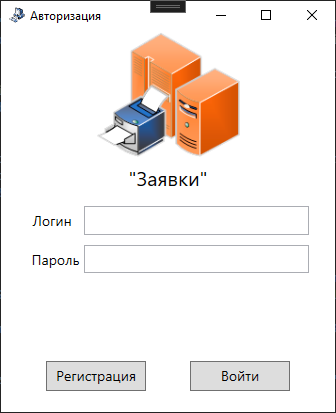


Рисунок 19 – Окно авторизации

Сначала проверяется попытка входа без ввода данных. Результатом будет ошибка, представленная на рисунке 20.

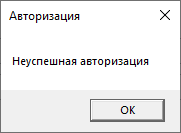


Рисунок 20 – Неуспешная авторизация

Таким же образом закончится попытка входа с использование неправильных данных. Для успешной авторизации используем те же данные что и при регистрации.

В результате должно быть показано сообщение об успешной авторизации, представленное на рисунке 21, и открыто главное окно приложения.

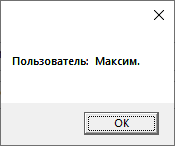


Рисунок 21 – Успешная авторизация

# **7** **Техническая документация**

В процессе разработки была разработана программная документация на систему, включающая в себя техническое задание на разработку (Приложение А), руководство пользователя (Приложение В).

**ЗАКЛЮЧЕНИЕ**

В результате выполнения курсовой работы была разработана информационная система для составления и управления заявками на обслуживание и ремонт компьютерной техники, позволяющая добавлять и управлять заявками на обслуживание компьютерной техники. Данная информационная система предназначена для ...., включает в себя приложение для взаимодействия с пользователем и базой данных, и базу данных для хранения заявок. В результате применения данной информационной системы будет сокращено время обслуживания клиентов, ускорен процесс работы, повышено качество обслуживания, сокращена трудоемкость на составление заявок.

Дальнейшее улучшение возможно за счёт использования новых web-технологий, создание браузерной версии приложения для привлечения новой аудитории, расширение функциональных возможностей приложения (добавление новых ролей пользователей, статусов заявок), а так же общего улучшения качества программы.

**Список информационных источников**

1. Вендров А.М. Один из подходов к выбору средств проектирования баз данных и приложений. "СУБД", 2003, №3.
2. Зиндер Е.З. Бизнес-реинжиниринг и технологии системного проектирования. Учебное пособие. М., Центр Информационных Технологий, 2000
3. Калянов Г.Н. CASE. Структурный системный анализ (автоматизация и применение). М., "Лори", 1999.
4. Международные стандарты, поддерживающие жизненный цикл программных средств. М., МП "Экономика", 1998
5. Новоженов Ю.В. Объектно-ориентированные технологии разработки сложных программных систем. М., 1999.
6. Обучающий сайт Microsoft. <https://learn.microsoft.com/ru-ru/>.
7. Пример работы с базой данных на основе Entity Framework. <https://github.com/artemovsergey/StudyPractice-SportStore>.
8. Зиборов, В.В. Visual C# 2012 на примерах / В.В. Зиборов. - М.: БХВ-Петербург, 2013.
9. Рихтер, Джеффри CLR via C#. Программирование на платформе Microsoft .NET Framework 4.0 на языке C# / Джеффри Рихтер. - М.: Питер, 2013.
10. Лотка, Рокфорд C# и CSLA .NET Framework. Разработка бизнес-объектов / Рокфорд Лотка. - М.: Вильямс, 2010.
11. Бишоп, Дж. C# в кратком изложении / Дж. Бишоп, Н. Хорспул. - М.: Бином. Лаборатория знаний, 2013.
12. Вагнер, Билл С# Эффективное программирование / Билл Вагнер. - М.: ЛОРИ, 2013.

# **Приложение А**

**Техническое задание на разработку системы "Заявки"**

**Содержание**

1. Введение

1.1. Наименование системы

1.2. Назначение и область применения

2. Требования к программе

2.1. Требования к функциональным характеристикам

2.2. Требования к надежности

2.2.1. Требования к обеспечению надежного функционирования программы

2.2.2. Время восстановления после отказа

2.2.3. Отказы из-за некорректных действий пользователей системы

3. Условия эксплуатации

3.1. Климатические условия эксплуатации

3.2. Требования к квалификации и численности персонала

3.3. Требования к составу и параметрам технических средств

3.4. Требования к информационной и программной совместимости

3.4.1. Требования к информационным структурам и методам решения

3.4.1.1. Структура баз данных

3.4.1.2. Требования к запросам пользователей данных из базы

3.4.2. Требования к исходным кодам и языкам программирования

3.4.3. Требования к программным средствам, используемым программой

3.4.4. Требования к защите информации и программ

3.5. Специальные требования

4. Требования к программной документации

4.1. Предварительный состав программной документации

5. Технико-экономические показатели

5.1. Экономические преимущества разработки

6. Стадии и этапы разработки

6.1. Стадии разработки

6.2. Этапы разработки

6.3. Содержание работ по этапам

7. Порядок контроля и приемки

7.1. Виды испытаний

7.2. Общие требования к приемке работы

**1. Введение**

**1.1. Наименование системы**

Наименование системы: "Заявки"

**1.2. Назначение и область применения**

Система предназначена для создания и управления содержимым базы данных, содержащим следующие данные:

1.2.1. Заявки на ремонт/обслуживание компьютерной техники  
1.2.2. Клиентов  
1.2.3. Операторов/Администраторов

Система предоставляет Веб-интерфейс и клиентский графический интерфейс для управления содержимым базы данным в соответствии с предъявляемыми требованиями и разделена на две части.

**2. Требования к программе**

**2.1. Требования к функциональным характеристикам**

Система должна обеспечивать возможность выполнения перечисленных ниже функций:  
2.1.1. Разделение пользователей подключаемых через Веб интерфейс на группы:  
2.1.1.1. Клиентов  
2.1.1.2. Администраторов базы данных  
2.1.2. Возможность поиска (фильтрации) по базе данных информации по заявкам  
2.1.3. Возможность создания заявки в режиме онлайн авторизованными пользователями (клиентами).

2.1.4. Возможность изменения/удаления заявки в режиме онлайн администраторами.  
2.1.5. Для Администраторов базы данных возможность поиска (фильтрации) по базе данных информации по клиентам.  
2.1.6. Для Администраторов базы данных возможность анализа в базе данных динамики появления новых заявок.  
2.1.7. Возможность подключения независимых модулей импорта из внешних источников данных в текущую базу.  
**2.2. Требования к надежности**

**2.2.1. Требования к обеспечению надежного функционирования программы**

Надежное (устойчивое) функционирование программы должно быть обеспечено выполнением Заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:   
а) организацией бесперебойного питания технических средств;   
б) использованием лицензионного программного обеспечения;   
в) 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»;   
г) регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов

**2.2.2. 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 30-ти 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.

**2.2.3. Отказы из-за некорректных действий пользователей системы**

Отказы программы вследствие некорректных действий пользователя при взаимодействии с программой через графический интерфейс недопустимы.

**3. Условия эксплуатации**

**3.1. Климатические условия эксплуатации**

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации

**3.2. Требования к квалификации и численности персонала**

Минимальное количество персонала, требуемого для работы программы, должно составлять не менее 2 штатных единиц — системный администратор и конечный пользователь программы — оператор. Системный администратор должен иметь высшее профильное образование и сертификаты компании-производителя операционной системы. В перечень задач, выполняемых системным администратором, должны входить:   
а) задача поддержания работоспособности технических средств;   
б) задачи установки (инсталляции) и поддержания работоспособности системных программных средств — операционной системы;   
в) задача установки (инсталляции) программы.   
г) задача создания резервных копий базы данных.

**3.3. Требования к составу и параметрам технических средств**

3.3.1. В состав технических средств должен входить IВМ-совместимый персональный компьютер (ПЭВМ), выполняющий роль сервера, включающий в себя:

3.3.1.1. процессор i5-9600K 3.70GHz, не менее;   
3.3.1.2. оперативную память объемом, 8Гигабайт, не менее;   
3.3.1.3. HDD, 216 Гигабайт, не менее;   
3.3.1.4. операционную систему Windows Server или Windows 7/10;  
3.3.1.6. Microsoft SQL Server

**3.4. Требования к информационной и программной совместимости**

**3.4.1. Требования к информационным структурам и методам решения**

База данных работает под управлением Microsoft SQL Server. Используется много поточный доступ к базе данных. Необходимо обеспечить одновременную работу с программой с той же базой данной модулей экспорта внешних данных

**3.4.1.1. Структура баз данных**

Таблица пользователей – USERS

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| USER\_ID | Int | [уникальное] Идентификатор пользователя |
| NAME | VarChar(255) | Имя пользователя |
| SURNAME | VarChar(255) | [необязательно] Фамилия пользователя |
| EMAIL | VarChar(255) | Электронная почта |
| LOGIN | VarChar(255) | Логин |
| PASSWORD | VarChar(255) | Пароль |
| ROLE | Int | [внешний ключ] Роль пользователя |

Таблица заявок - REQUESTS

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| REQUEST\_ID | Int | [уникальное] Идентификатор заявки |
| USER\_ID | Int | [внешний ключ] Идентификатор пользователя |
| CREATION\_DATE | Date | Дата создания |
| LAST\_CHANGE\_DATE | Date | Дата последнего изменения |
| TROUBLE\_DEVICE | VarChar(255) | Проблемные устройства |
| PROBLEM\_DESCRIPTION | VarChar(255) | Описание проблемы |
| STATUS | Int | [внешний ключ] Статус заявки |

Таблица ролей - ROLES

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| ROLE\_ID | Int | [уникальное] Идентификатор роли |
| ROLE\_NAME | VarChar(100) | Название роли |

Таблица статусов - STATES

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| STATUS\_ID | Int | [уникальное] Идентификатор статуса |
| STATUS\_NAME | VarChar(50) | Название статуса |

**3.4.1.2. Требования к запросам пользователей данных из базы**

Пользователи и администраторы работают с базой данных через графический интерфейс.

Администраторы системы должны иметь возможность редактировать таблицы, перечисленные п. 3.4.1.1. (добавление, редактирование)

Пользователи системы должны иметь возможность производить поиск по таблице REQUESTS, просматривать детальную информацию по каждому результату выборки

**3.4.2. Требования к исходным кодам и языкам программирования**

Дополнительные требования не предъявляются.

**3.4.3. Требования к программным средствам, используемым программой**

Системные программные средства, используемые программой, должны быть представлены лицензионной локализованной версией операционной системы Windows Server или Windows 7/10 и Microsoft SQL Server.

**3.4.4. Требования к защите информации и программ**

Требования к защите информации и программ не предъявляются.

**3.5. Специальные требования**

Программа должна обеспечивать одновременную работу пользователей посредством графического интерфейса.

**4. Требования к программной документации**

**4.1. Предварительный состав программной документации**

Состав программной документации должен включать в себя:   
4.1.1. техническое задание;  
4.1.2. программу и методики испытаний;  
4.1.3. руководство оператора;

**5. Технико-экономические показатели**

**5.1. Экономические преимущества разработки**

Ориентировочная экономическая эффективность не рассчитываются. Аналогия не проводится ввиду уникальности предъявляемых требований к разработке.

**6. Стадии и этапы разработки**

**6.1. Стадии разработки**

Разработка должна быть проведена в три стадии:   
1. разработка технического задания;   
2. рабочее проектирование;   
3. внедрение.

**6.2. Этапы разработки**

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.   
На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

1. разработка программы;   
2. разработка программной документации;   
3. испытания программы.

На стадии внедрения должен быть выполнен этап разработки подготовка и передача программы.

**6.3. Содержание работ по этапам**

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:   
1. постановка задачи;   
2. определение и уточнение требований к техническим средствам;   
3. определение требований к программе;  
4. определение стадий, этапов и сроков разработки программы и документации на неё;   
5. согласование и утверждение технического задания.   
На этапе разработки программы должна быть выполнена работа по программированию (кодированию) и отладке программы.   
На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями к составу документации.   
На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:   
1. разработка, согласование и утверждение и методики испытаний;   
2. проведение приемо-сдаточных испытаний;   
3. корректировка программы и программной документации по результатам испытаний.   
На этапе подготовки и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию на объектах Заказчика.

**7. Порядок контроля и приемки**

**7.1. Виды испытаний**

Приемо-сдаточные испытания должны проводиться на объекте Заказчика в оговоренные сроки.   
Приемо-сдаточные испытания программы должны проводиться согласно разработанной Исполнителем и согласованной Заказчиком Программы и методик испытаний.   
Ход проведения приемо-сдаточных испытаний Заказчик и Исполнитель документируют в Протоколе проведения испытаний

**7.2. Общие требования к приемке работы**

На основании Протокола проведения испытаний Исполнитель совместно с Заказчиком подписывает Акт приемки-сдачи программы в эксплуатацию

# **Приложение Б**

Исходный программный код информационной системы «Заявки»

**ПРИЛОЖЕНИЕ В**

Руководство пользователя