|  |
| --- |
|  |
|  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **КУРСОВАЯ РАБОТА** | | | | | | | | | | | | | | |
| по: | ПМ.05 Проектирование и разработка информационных систем | | | | | | | | | | | | | |
| междисциплинарного курса: | | | | | | | МДК 05.01 Проектирование и дизайн информационных систем | | | | | | | |
| на тему: | | | | | | | | | | | | | | |
| Разработка модуля информационной системы для автоматизации работы  Волонтерского центра | | | | | | | | | | | | | | |
| Выполнена студентом: | | | | | | Назаркиным Романом | | | | | | | | |
|  | | | | | | И.О. Фамилия | | | | | | | | |
| Группы: | | 3ИСП9-22 | | | | | | | | | | | | |
|  |  | номер группы | | | | | | | | | | | | |
| Основная профессиональная образовательная программа по специальности 09.02.07 Информационные системы и программирование | | | | | | | | | | | | | | |
| шифр и наименование специальности | | | | | | | | | | | | | | |
| Форма обучения | | | | | очная | | | | | | | | | |
| Руководитель | | |  | преподаватель | | | | |  | Кристина Александровна Бастрыкина | | | | |
|  | | | ученая степень, должность | | | | | | | И.О. Фамилия | | | | |
|  | | |  | | | | | | | оценка | | | | |
|  | | |  | | | | | | | подпись, дата | | | | |
| Зав. отделением | | |  |  | | | | |  | Ольга Николаевна Микловш | | | | |
|  | | |  | | | | | | | И.О. Фамилия | | | | |
|  |  |  | | | |  | |  | подпись, дата | | | | | |
|  |  |  | | | |  | |  |  | |  |  |  |  |
| г. Москва, 2024 г. | | | | | | | | | | | | | | |

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  | **УТВЕРЖДАЮ**  Зам.директора по ОУП  \_\_\_\_\_\_\_\_\_\_И.Г.Бозрова  «\_\_\_»\_\_\_\_\_\_\_\_\_\_2024 г. | |

**ЗАДАНИЕ**

к курсовой работе студента

Романа Назаркина

на тему: «Разработка модуля информационной системы для автоматизации отправки заявок на железнодорожные перевозки»

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Тема курсовой работы утверждена приказом директора № \_\_\_ от \_\_\_\_\_\_\_ | | | | | | | | | | | | |
| Цель работы: | | | Проектирование информационной системы, разработка сопроводительной документации на примере конкретной предметной области | | | | | | | | | |
| Основные вопросы, подлежащие разработке: | | | | | | | | | | | | |
| 1. Анализ предметной области. | | | | | | | | | | | | |
| 2. Проектирование информационной системы. | | | | | | | | | | | | |
| 3.Разработка ER-диаграммы. | | | | | | | | | | | | |
| 4.Разработка UML-Диаграммы. | | | | | | | | | | | | |
| Теоретическая часть: | | | | |  | | | | | | | |
| 1. Анализ особенностей предметной области. | | | | | | | | | | | | |
| 2. Сравнительный анализ программных средств. | | | | | | | | | | | | |
| Практическая часть: | | | | |  | | | | | | | |
| 1.Разработка базы данных автоматизированной информационной системы (создание ERВ диаграммы в MS SQL Server). | | | | | | | | | | | | |
| 2. Разработка визуального интерфейса. | | | | | | | | | | | | |
| Основная литература: | | | | | | | | | | | | |
| 1. Информационные системы и технологии: Учебник / П.П. Иванов, А.А. Смирнов. - Москва: Издательство Юрайт, 2021. - 320 с. | | | | | | | | | | | | |
| 2. Управление информационными системами: Практическое пособие / В.И. Петров, Е.А. Сидорова. - Санкт-Петербург: БХВ-Петербург, 2020. - 256 с. | | | | | | | | | | | | |
| 3. Современные информационные технологии в бизнесе: Монография / А.Б. Козлов, Л.Г. Михайлов. - Москва: Инфра-М, 2019. - 184 с. | | | | | | | | | | | | |
| Руководитель: | | | | Кристина Александровна Бастрыкина | | | | | | |  | |
|  |  | | |  | |  |  |  |  |  |  | подпись |
| Дата выдачи задания: | | | | | | дд.мм.2024 г. | | | | | | |
| Задание получил: | | | | Роман Назаркин | | | | | | |  |  |
|  | |  | |  | | | | | | |  | подпись |
| Дата получения задачи: | | | | | | дд.мм.2024 г. | | | | | | |

**ПЛАН-ГРАФИК**

курсовой работы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Срок выполнения курсовой работы | |  | | |
| Выбор темы курсовой работы | |  | | |
| Подбор и анализ исходной информации | |  | | |
| Подготовка и утверждение плана курсовой работы | |  | | |
| Выполнение задач курсовой работы | |  | | |
| Оформление курсовой работы | |  | | |
| Исправление замечаний курсовой работы | |  | | |
| Предоставление готовой курсовой работы для рецензирования | |  | | |
| Предзащита курсовой работы | |  | | |
| Защита курсовой работы | |  | | |
|  |  | |  |  |

|  |  |
| --- | --- |
| Руководитель | Кристина Александровна Бастрыкина |
|  | подпись |
| План принял к исполнению | Роман Назаркин |
|  | подпись |
|  | дд.мм.2024 г. |
|  |  |

Оглавление

[ВВЕДЕНИЕ 5](#_Toc164380051)

[ГЛАВА 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПРИМЕНЯЕМЫХ ПРОГРАММНЫХ СРЕДСТВ 7](#_Toc164380052)

[1.1. Анализ особенностей предметной области 7](#_Toc164380053)

[1.2. Сравнительный анализ используемых программных средств 8](#_Toc164380054)

[1.2.1 Анализ и выбор средств разработки серверной части приложения 8](#_Toc164380055)

[1.2.2 Анализ и выбор средств разработки клиентской части 10](#_Toc164380056)

[1.2.3 Анализ и выбор базы данных 11](#_Toc164380057)

[1.2.4 Анализ IDE 12](#_Toc164380058)

[Выводы 12](#_Toc164380059)

[ГЛАВА II. РАЗРАБОТКА И ВНЕДРЕНИЕ ПРОГРАММНОГО ПРОДУКТА 14](#_Toc164380060)

[2.1. Проектирование базы данных 14](#_Toc164380065)

[2.2. Создание ERD-диаграммы 15](#_Toc164380066)

[2.3. Разработка визуального интерфейса 19](#_Toc164380067)

[Вывод 27](#_Toc164380068)

[ЗАКЛЮЧЕНИЕ 29](#_Toc164380069)

[Приложение 1. Полноразмерная схема базы данных, разработанная в Microsoft Office Visio 2022 4](#_Toc164380070)

[Приложение 2. Диаграмма базы данных, разработанная в Microsoft SQL Server Management Studio 5](#_Toc164380071)

[Приложение 3. Use-case диаграмма, разработанная в Microsoft Visio 6](#_Toc164380071)

# ВВЕДЕНИЕ

В области железнодорожных перевозок важное место занимает простота и скорость оформления заявок. В условиях современного мира, характеризующегося ускорением информационных процессов, автоматизация процессов, связанных с оформлением различных документов (в первую очередь заявок) может оказаться необходимым шагом для эффективного функционирования и координации организаций, предоставляющих подобные услуги. Повышение качества железнодорожных перевозок за счет внедрения информационных систем является актуальным направлением для исследования и разработки.

Существующий уровень технологического развития позволяет создать условия для более оперативного и понятного интерфейса взаимодействия заказчика и организации, к тому же эта информационная система может быть универсальной и выступать отличным решением для ряда таких организаций.

Целью данной курсовой работы является разработка модуля информационной системы, который позволит автоматизировать отправление и просмотр заявок, редактирование и удаление пользователей. В рамках работы предполагается решение следующих задач:

* Исследовать и систематизировать заявки, их наполнение.
* Определить требования к функциональности модуля информационной системы.
* Провести анализ потенциальных технологий для реализации модуля.
* Разработать и тестировать прототип модуля информационной системы.

Объектом исследования выступают коммерческие железнодорожные перевозки.

Предметом исследования является процесс автоматизации оформления заявок с помощью разрабатываемого модуля информационной системы.

Таким образом, введение курсовой работы формулирует основные направления и план исследования, а также обосновывает важность решения задач автоматизации для повышения эффективности работы Волонтерского центра.

# ГЛАВА 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПРИМЕНЯЕМЫХ ПРОГРАММНЫХ СРЕДСТВ

* 1. Анализ особенностей предметной области

В рамках данной курсовой работы рассматривается предметная область железнодорожных перевозок.

В первую очередь необходимо проанализировать их структуру.

* Заказчик – конечный пользователь нашей информационной системы, с помощью которой может формировать и отправлять заявки на железнодорожные перевозки товара внутри страны.
* Оператор – связующее звено между организацией, предоставляющей услуги перевозок (в дальнейшем просто Организация), и заказчиком, может отклонять/одобрять заявки, вносить изменения в данные пользователей, удалять их.
* Организация – компания, предоставляющая услуги коммерческих железнодорожных перевозок.

Разрабатываемая мной информационная система должна позволить сократить временные затраты, разгрузить сотрудников и повысить производительность.

Данная информационная система должна:

* Регистрация новых пользователей – форма регистрации пользователей, которая защищает пользователей от рядовых ошибок заполнения профиля.
* Прием заявок – создание электронной системы для подачи и отслеживания заявок на перевозки товара.
* Операции с заявками – создание функционала, позволяющего одобрять или отклонять заявки.
* Операции с данными пользователей – создание функционала, с помощью которого администратор может помогать пользователям исправлять ошибки заполнения данных

Для того, чтобы приложение выполняло весь заявленный функционал, необходимо выбрать необходимые инструменты для разработки системы.

* 1. Сравнительный анализ используемых программных средств

Первое с чем необходимо определиться – это формат приложения. Лучшим вариантом будет разработка клиент-серверного приложения. Во-первых, этот вариант позволит легко масштабировать будущие приложения, реализовать клиентскую часть на настольном компьютере или веб-сайте. Во-вторых, это позволяет использовать одну реализацию бизнес-логики программы. В-третьих, это возможность иметь единый набор юнит-тестов.

Для того чтобы пользователь мог пользоваться приложением любым удобным ему способом, было принято решение разработать клиентскую часть приложения на настольном компьютере.

* + 1. Анализ и выбор средств разработки серверной части приложения

Для выбора языка программирования для разработки серверной части рассмотрим несколько вариантов: C#, C++, Java. Рассмотрим каждый из них.

C# - это современный, объектно-ориентированный язык программирования. C# позволяет разработчикам создавать множество типов безопасных и надежных приложений, работающих в .NET. C# уходит своими корнями в семейство языков C и будет сразу знаком программистам на C, C++, Java и JavaScript.

На языке C# реализован фреймворк ASP.NET Core. ASP.NET Core — свободно-распространяемый кроссплатформенный фреймворк для создания веб-приложений на платформе .NET с открытым исходным кодом. Данная платформа разрабатывается компанией Майкрософт совместно с сообществом и имеет большую производительность по сравнению с ASP.NET. Имеет модульную структуру и совместима с такими операционными системами как Windows, Linux и macOS.

Язык C# предоставляет удобный язык для написания простых и сложных систем, активно развивается компанией Microsoft, имеет множество удобных библиотек и интерфейсов, необходимых в ходе разработки.

C++ — компилируемый строго типизированный язык программирования общего назначения. Поддерживает разные парадигмы программирования: процедурную, обобщённую, функциональную; наибольшее внимание уделено поддержке объектно-ориентированного программирования.

Разработка языка началась в 1979 году. Целью создания C++ было дополнение C возможностями, удобными для масштабной разработки ПО, с сохранением гибкости, скорости и портабельности C. Вместе с тем создатели C++ стремились сохранить совместимость с C: синтаксис первого основан на синтаксисе последнего, и большинство программ на C будут работать и как C++. Изначально новый язык назывался “C с классами”, но затем имя было изменено на C++ — это должно было подчеркнуть, как его происхождение от C, так и его превосходство над последним.

Java — это объектно-ориентированный язык программирования, разработанный компанией Sun Microsystems (в последующем приобретённой компанией Oracle). Дата официального выпуска — 23 мая 1995 года.

Программы на Java транслируются в байт-код, который затем выполняется виртуальной машиной Java (JVM). JVM — это программа, которая обрабатывает байтовый код и передает инструкции оборудованию как интерпретатор. Достоинством подобной реализации является независимость байт-кода от операционной системы и оборудования, что позволяет выполнять Java-приложения на любом устройстве, для которого существует JVM.

В ходе анализа средств разработки, было принято решение выбрать язык программирования C#, как самый удобный и включающий все удобные библиотеки для реализации и тестирования.

* + 1. Анализ и выбор средств разработки клиентской части

Так как перечень задач не представляет ничего сложного, инструментарий стоит выбирать соответствующий.

Для разработки настольного приложения будем использовать WPF.

Windows Presentation Foundation — это платформа пользовательского интерфейса для создания клиентских приложений для настольных систем. Платформа разработки WPF поддерживает широкий набор компонентов для разработки приложений, включая модель приложения, ресурсы, элементы управления, графику, макет, привязки данных, документы и безопасность.

WPF является частью платформы .NET, и если ранее вы создавали приложения в .NET с помощью ASP.NET или Windows Forms, то общий процесс программирования вам уже знаком. WPF использует расширяемый язык разметки для приложений (XAML), чтобы предоставить декларативную модель для программирования приложений.

Также для WPF существует библиотека Material Design, которая позволяет реализовать интерфейс согласно лучшим практикам проектирования пользовательского интерфейса.

Так как WPF использует разметку XAML, в будущем будет возможно реализовать веб-интерфейс, выполняющий те же самые функции (это возможно благодаря XBAP - XBAP — это WPF-приложение, которые выполняется внутри браузера Web-страниц. Приложение ХВАР является полноценным приложениям WPF, имеющим некоторые основные отличия: ХВАР-приложение выполняется внутри окна браузера и может занимать всю область Web-страницы или только часть).

* + 1. Анализ и выбор базы данных

Для хранения данных был выбран SQL Server и технология Entity Framework для работы с SQL Server на серверной части приложения. Entity Framework представляет собой более высокий уровень абстракции, который позволяет абстрагироваться от самой базы данных и работать с данными независимо от типа хранилища.

Преимущества MS SQL:

* документация и много обучающих материалов;
* бесшовная с разнообразными фичами база данных;
* полезные инструменты разработчиков и GUI-based подход;
* доступна как SAAS, который используется в больших компаниях с облачными технологиями;
* удобный интерфейс;

Присутствует один существенный недостаток: для запуска рабочей версии приложения хотя бы среднего уровня требуются большие затраты.

* + 1. Анализ IDE

Для разработки настольного приложения и серверной части будет использоваться Microsoft Visual Studio.

Microsoft Visual Studio — это интегрированная среда разработки (IDE) от Microsoft. Он используется для разработки компьютерных программ, а также веб-сайтов, веб-приложений, веб-сервисов и мобильных приложений. Visual Studio использует платформы разработки программного обеспечения Microsoft, такие как Windows API, Windows Forms, Windows Presentation Foundation, Windows Store и Microsoft Silverlight. Он может создавать как собственный код, так и управляемый код.

В качестве среды для управления инфраструктурой SQL Server, была выбрана — SQL Server Management Studio.

SQL Server Management Studio (SSMS) — утилита из Microsoft SQL Server для конфигурирования, управления и администрирования всех компонентов Microsoft SQL Server. Утилита включает скриптовый редактор и графическую программу, которая работает с объектами и настройками сервера. Главным инструментом SQL Server Management Studio является Object Explorer, который позволяет пользователю просматривать, извлекать объекты сервера, а также полностью ими управлять.

Выводы

Изучив теоретический материал и проанализировав преимущества средств разработки программного обеспечения, сделаем выводы:

Данные средства разработки интуитивно понятны и обладают всеми необходимыми функциями, использование технологии обеспечит гибкость построения интерфейса, а язык программирования C# позволит обеспечить наилучшую логику взаимодействия элементов приложения.

# ГЛАВА II. РАЗРАБОТКА И ВНЕДРЕНИЕ ПРОГРАММНОГО ПРОДУКТА

4. 1. Проектирование базы данных

Прежде чем строить физическую модель, необходимо решить, какая информация будет храниться в базе данных. Чтобы система смогла выполнять свои функции база данных должна включать в себя основные сущности:

* пользователь
* заявки
* товары

Для представления будущей базы данных создадим ER-диаграмму.

ER-модель (от англ. Entity-Relationship model, модель «сущность — связь») — модель данных, позволяющая описывать концептуальные схемы предметной области.

ER-модель используется при высокоуровневом (концептуальном) проектировании баз данных. С её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между этими сущностями.

ER-модель представляет собой формальную конструкцию, которая сама по себе не предписывает никаких графических средств её визуализации. В качестве стандартной графической нотации, с помощью которой можно визуализировать ER-модель, была предложена диаграмма «сущность-связь» (англ. Entity-Relationship diagram, ERD, ER-диаграмма).

Для проверки правильности построенной модели она приводится к 3 нормальным формам:

- отношение находится в 1НФ, если все его атрибуты являются простыми. Исходное отношение строится таким образом, чтобы оно было в 1НФ;

- отношение находится в 2НФ, если оно находится в 1НФ и каждый не ключевой атрибут функционально полно зависит от первичного ключа (составного);

- отношение находится в ЗНФ в том и только в том случае, если все не ключевые атрибуты отношения взаимно независимы и полностью зависят от первичного ключа.

Логическая модель базы данных, разработанная в Microsoft Visio, представлена в Приложении №1.

* 1. Создание ERD-диаграммы

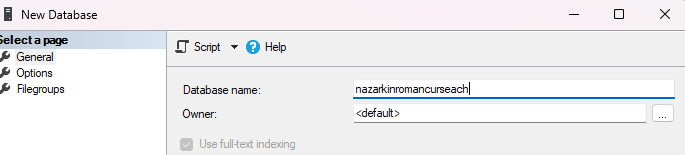
Формировать базу данных будем сразу в диаграмме базы данных в Microsoft SQL Server. В первую очередь, создаем саму базу данных (рис.1):

Рисунок 1. Создание базы данных nazarkinromancurseach

Далее создаем необходимые сущности.

Сущность user(рис.2):

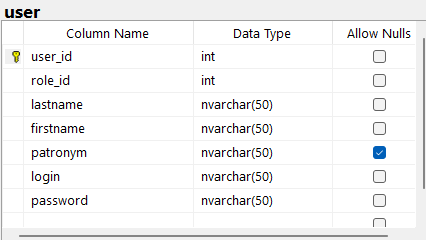


Рисунок 2. Сущность user

Данная сущность включает в себя: уникальный идентификатор, фамилию, имя и отчество пользователя, логин, пароль, роль.

Сущность client(рис3):

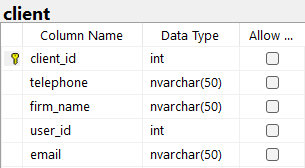
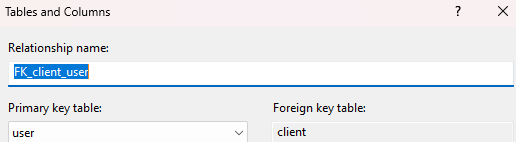


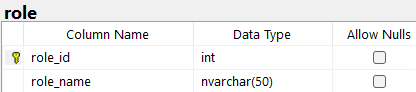
Рисунок 3. Сущность client

Сущность client включает в себя: уникальный идентификатор, телефон, название фирмы пользователя, внешний ключ уникального идентификатора user, почту.

Связь «один-к-одному» между сущностями пользователь-клиент(рис.4):

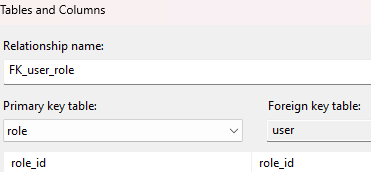
  
Рисунок 4. Связь user-client

Сущность role(рис5):

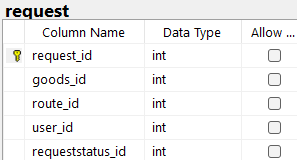
  
Рисунок 5. Сущность role

Включает в себя: уникальный идентификатор и название роли.

Добавляем необходимые связи, такие как пользователь-роль(рис.7):

  
Рисунок 7. Связь user-role

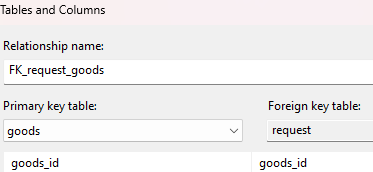
Сущность request(рис.8)

  
Рисунок 8. Сущность request

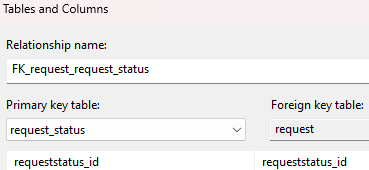
Включает в себя уникальный ключ, внешний ключ идентификатора товаров, внешний ключ идентификатора маршрутов, внешний ключ идентификатора пользователя, внешний ключ статуса заявки.

Добавляем необходимые связи

Связь заявка-товар(рис.9):

  
Рисунок 9. Связь request-goods

Связь заявка-статус заявки(рис.10):

  
Рисунок 10. Связь request-request status

Связь заявка-маршрут(рис. 11):

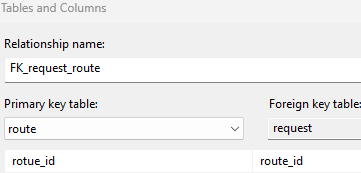


Рисунок 11. Связь request-route status

Связь заявка-пользователь(рис. 12):

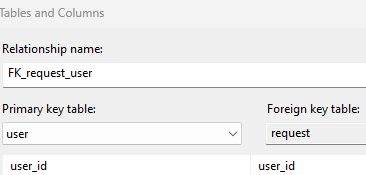


Рисунок 12. Связь request-user

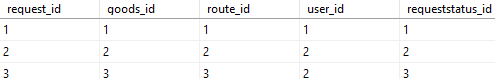
Полная ERD-диаграмма представлена в приложении №2.

Теперь, для того чтобы в будущем иметь возможность проверить функционал заданной системы, необходимо заполнить базу данных тестовыми данными. Таким образом заполняем основную сущность Пользователь (рис.13):

  
Рисунок 13. Заполнение сущности user

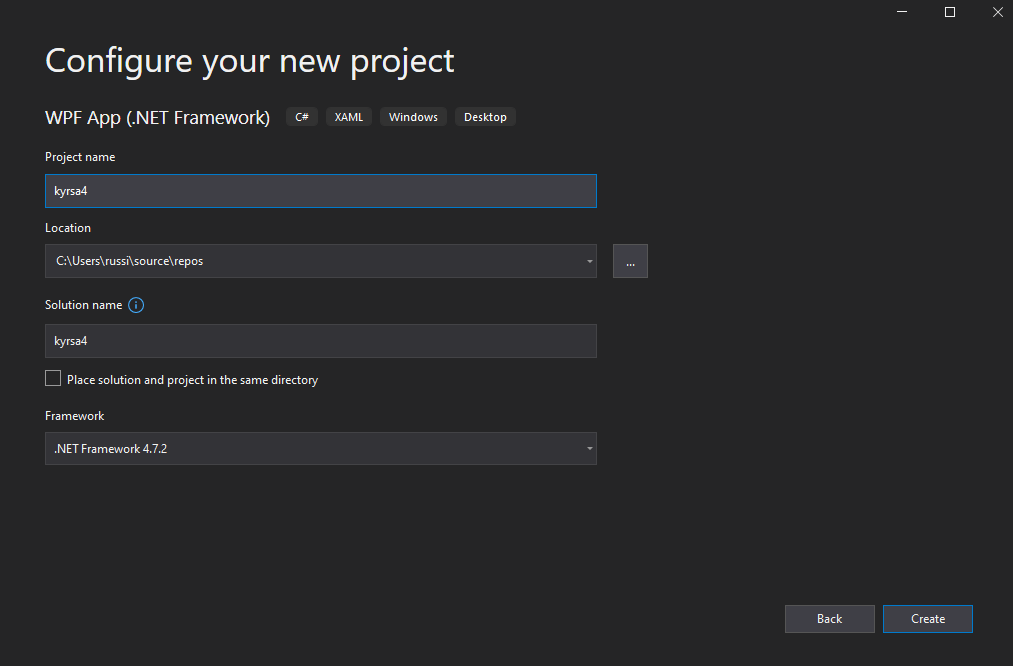
Таким же образом заполнили сущности Клиент и Заявки(рис.14):



  
Рисунок 14. Заполнение сущностей client, request

* 1. Разработка визуального интерфейса

В первую очередь создаем проект в интегрированной среде разработки Visual Studio(рис.15):

  
Рисунок 15. Создание приложения

Добавим в проект базу данных(рис.14):

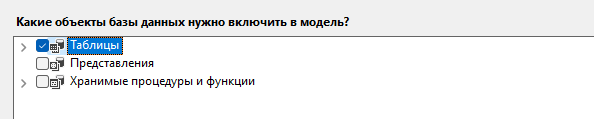
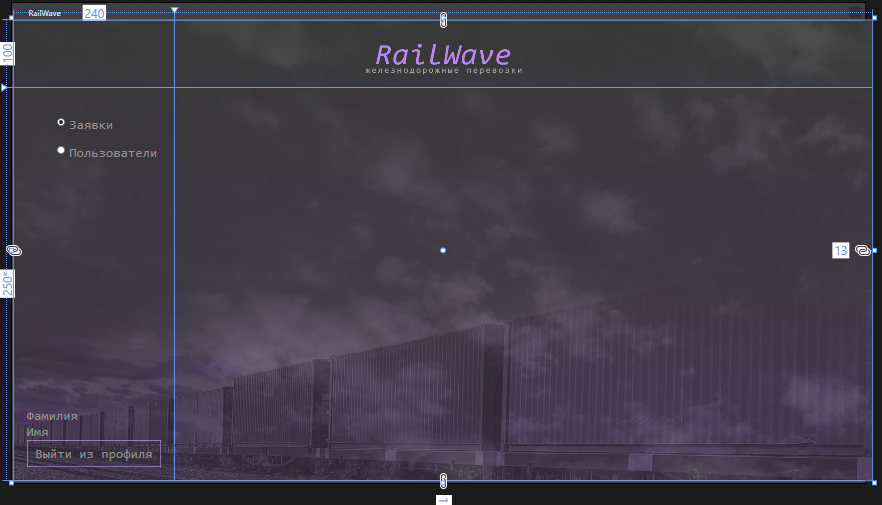
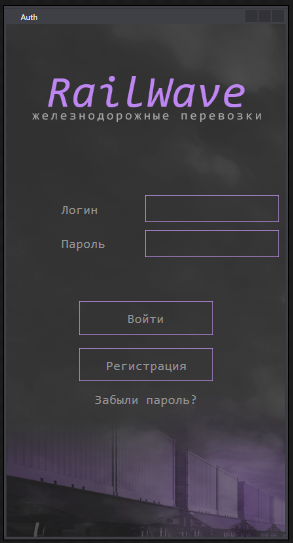


Рисунок 16. Добавление БД

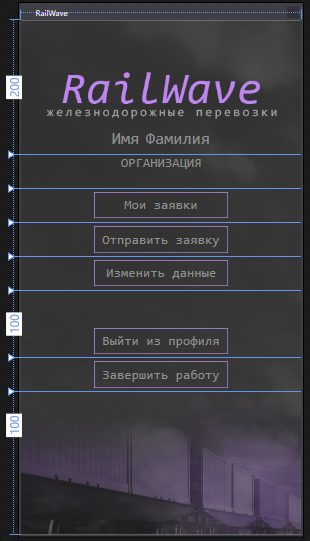
Теперь можно перейти к оформлению визуальной части приложения. Окно оператора будет выглядеть следующим образом: (рис.17):

  
Рисунок 17. Окно администратора

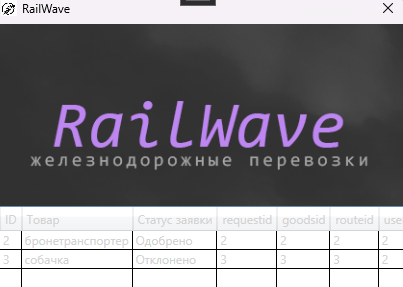
Для того чтобы пользователь мог пройти процедуру авторизации была создана страница авторизации(рис.18):

  
Рисунок 18. Страница авторизации

После успешной авторизации окно входа закрывается и открывается главное окно системы, наполнение которого зависит от роли пользователя (1 – окно администратора с выводом данных(рис.17), 2 – навигационное меню клиента(рис.19)):

  
Рисунок 19. Главное окно системы с отображением текущего времени даты

При нажатии кнопки «Мои заявки» выводятся все заявки пользователя(рис.20):

  
Рисунок 20. Главное окно со страницей статистики

При нажатии на пункте меню «Отправить заявку» открывается форма заполнения заявки, в которой пользователь формирует свою заявку, после чего отправляет ее (пере отправкой происходит проверка на пустые поля) (рис.21):

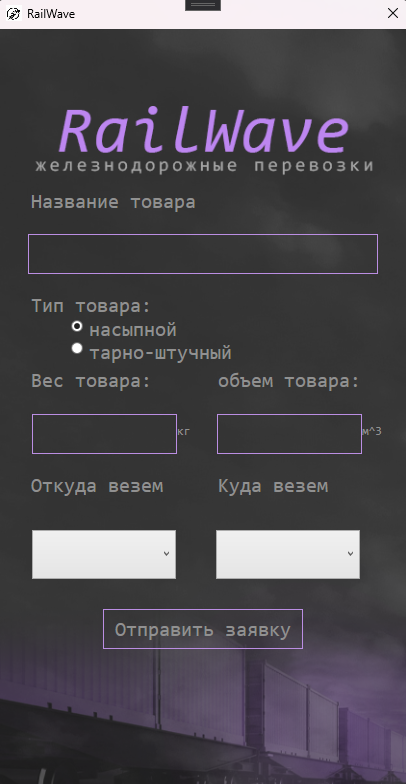
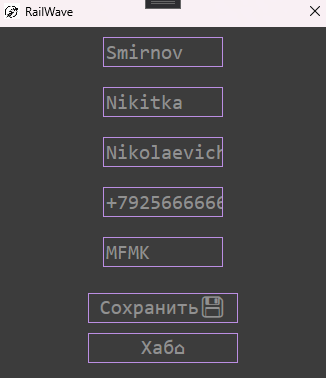


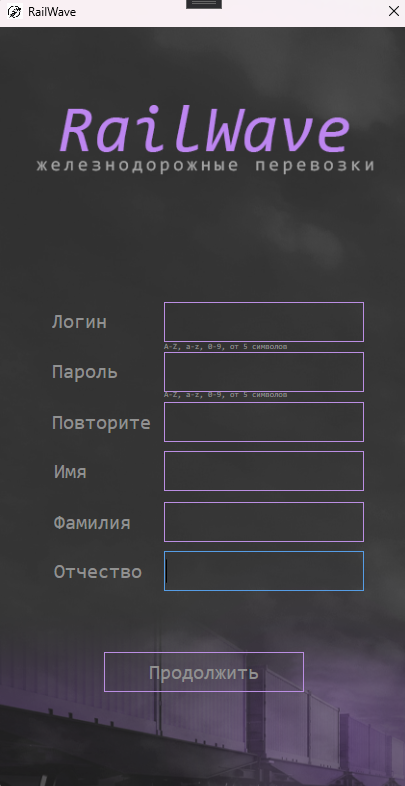
Рисунок 21. Главное окно с информацией о волонтерах

По нажатию на «Изменить данные» открывается форма с данными пользователя, которые он может поменять (сохранение данных происходит по нажатию сохранить, что предусматривает случайное изменение в полях ввода)(рис.22):

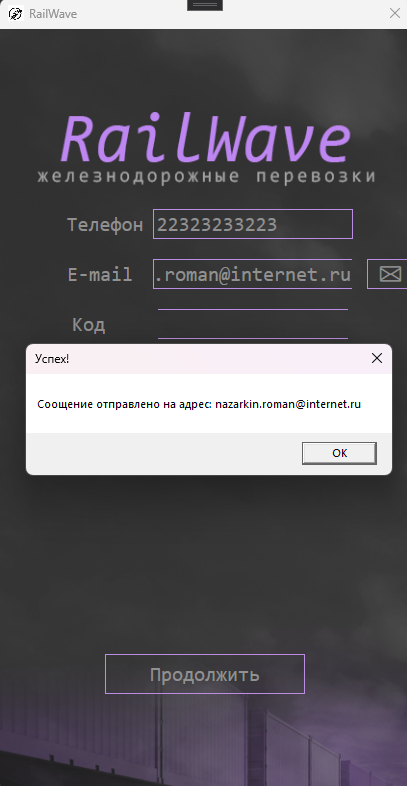
  
Рисунок 22. Окно изменения данных клиента

При нажатии на «Выйти из профиля» пользователь возвращается на форму авторизации (рис.18), а при нажатии на кнопку «Завершить работу» приложение закрывается.

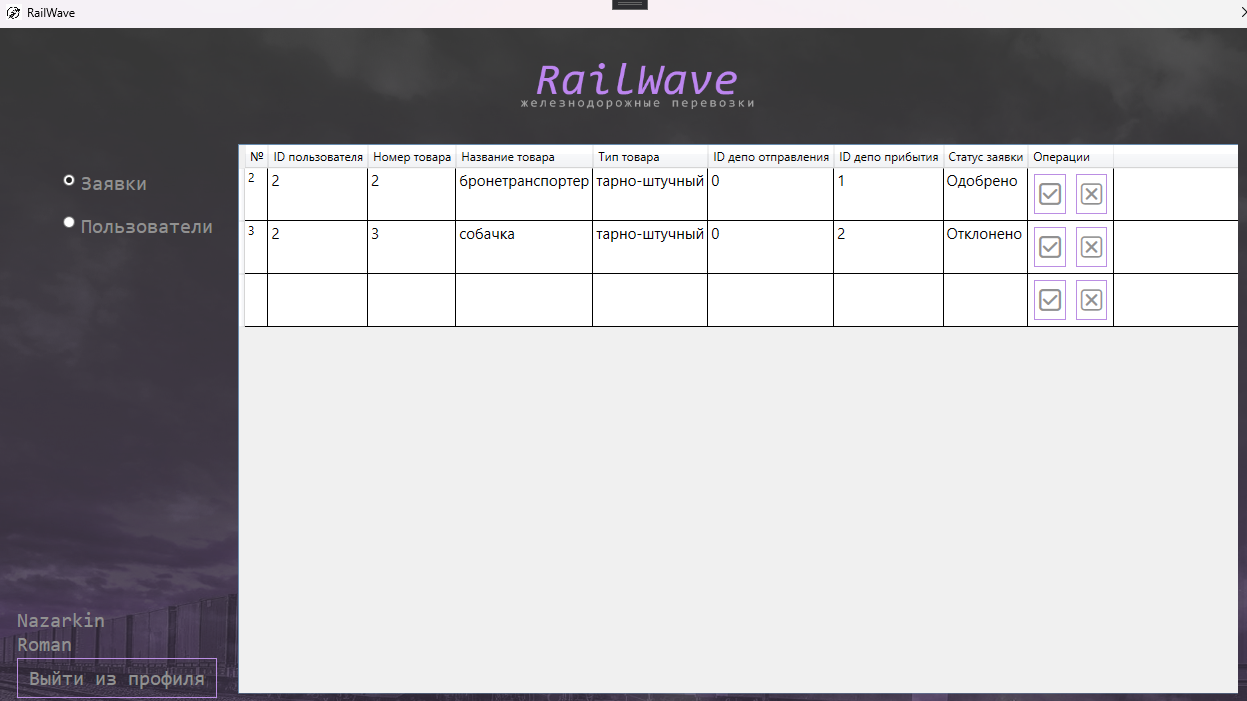
На форме авторизации присутствует кнопка «Регистрация», по нажатию на которую открывается окно регистрации, которое состоит из текстовых полей, подсказок и кнопки «Продолжить» (рис.23):

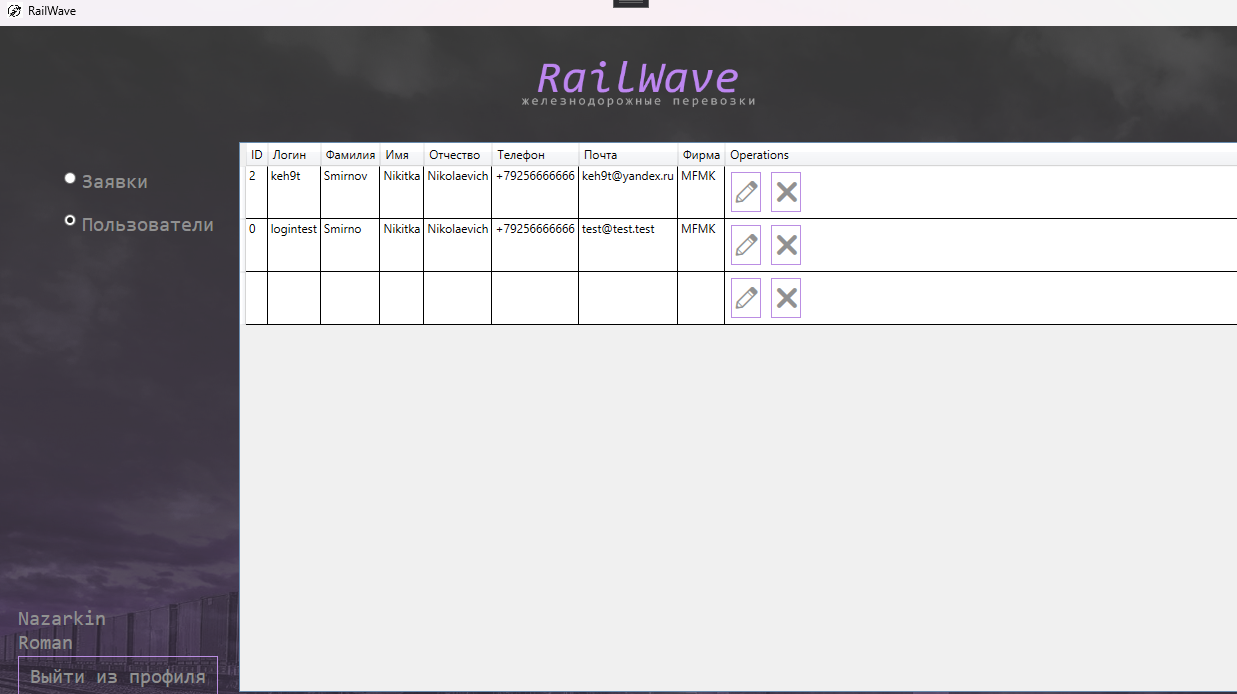
  
Рисунок 23. Окно регистрации (1)

По нажатии на кнопку «Продолжить» пользователь вводит оставшиеся данные, нажимает на кнопку с конвертом, после чего в случае успешной отправки сообщения появляется уведомление об этом (рис.24):

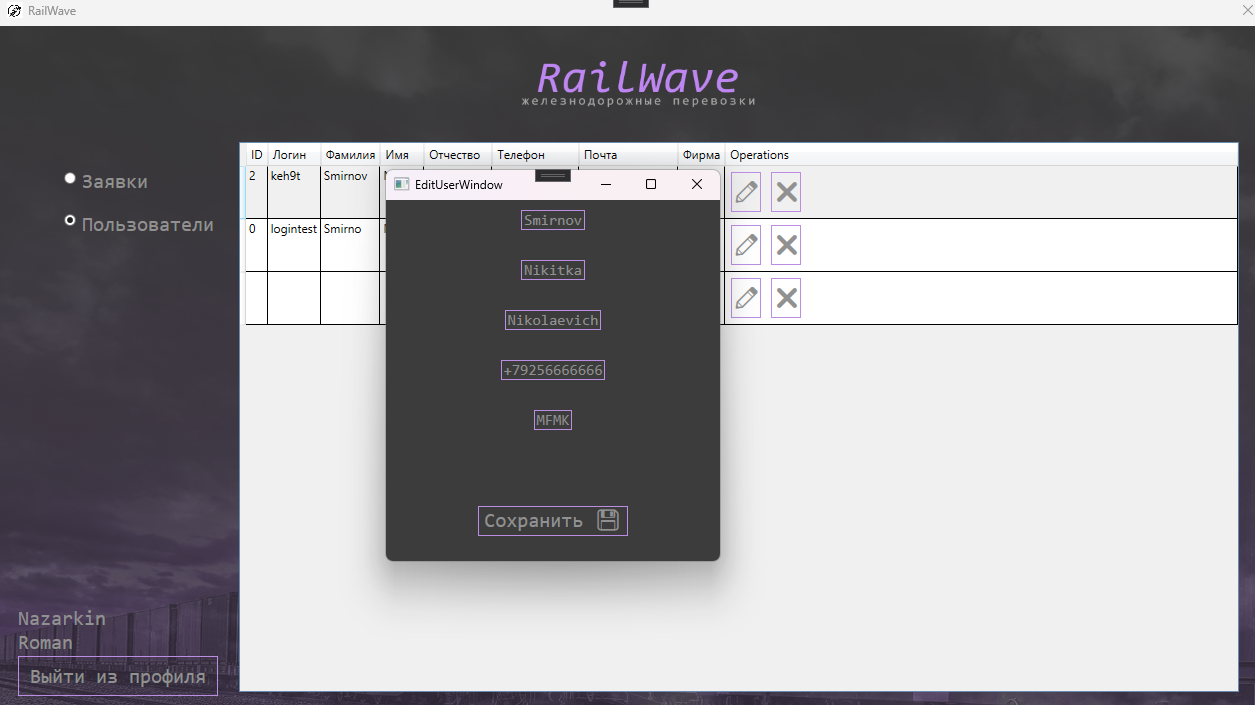
  
Рисунок 24. Окно добавления волонтеров

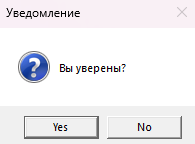
Вывод данных у оператора содержит кнопки переключения выводов данных, имя с фамилией, кнопку «Выйти из профиля», вывод. На выводе помимо самих данных есть столбец «Операции». На вкладке заявки операции отвечают за одобрение или отклонение заявок (рис.25):



  
Рисунок 25. Окно добавления волонтеров

На странице «Пользователи» кнопки из столбца «Операции» отвечают за изменение данных пользователя (рис.26) и удаления пользователя (перед удалением появляется уведомление, которое опрашивает оператора на готовность) (рис.26):



  
Рисунок 26. Окно изменения данных пользователя и уведомление перед удалением пользователя

Система функционирует исправно и доступна для использования. Благодаря возможной интеграции с облачной базой данных, данное приложение может быть запущено на любом компьютере без привязки к конкретному рабочему месту, что обеспечивает удобный и гибкий доступ к информации и инструментарию для работы.

Вывод

Анализ в первой главе позволил нам создать ERD диаграмму, на основе которой была создана база данных, которая стала удобным хранилищем информации для будущего приложения. Также была разработано настольное приложение. Интерфейс, созданный для этого приложения понятен и удобен. Применение разработанной автоматизированной системы позволит увеличить скорость операций взаимодействия клиентов и организаций, предоставляющих услуги коммерческих железнодорожных перевозок, что приводит к повышению эффективности компании.

**ЗАКЛЮЧЕНИЕ**

Целью данной курсовой работы являлось закрепление полученных теоретических знаний и практических умений, путем проектирования и разработки информационной системы. Поставленная цель была достигнута путем использования таких программных средств как: СУБД Microsoft SQL Server, языков программирования C# и технологии Windows Presentation Foundation. База данных была приведена к третьей нормальной форме, а приложение написано в объектно-ориентированном стиле.

Созданная система, может быть расширена различными интеграциями.

Разработанный программный продукт удобен в использовании и предоставляет приятный для человека визуальный интерфейс, позволяющий любому пользователю без труда освоить работу с ним.

Всё вышеперечисленное позволят повысить доступность информации и её скорость получения. В дальнейшем планируются увеличение количества интеграций, улучшение визуального интерфейса, поддержка удаленного доступа к базе данных, а также устранение ошибок и поддержка, с реализацией дополнительных функций по требованию.

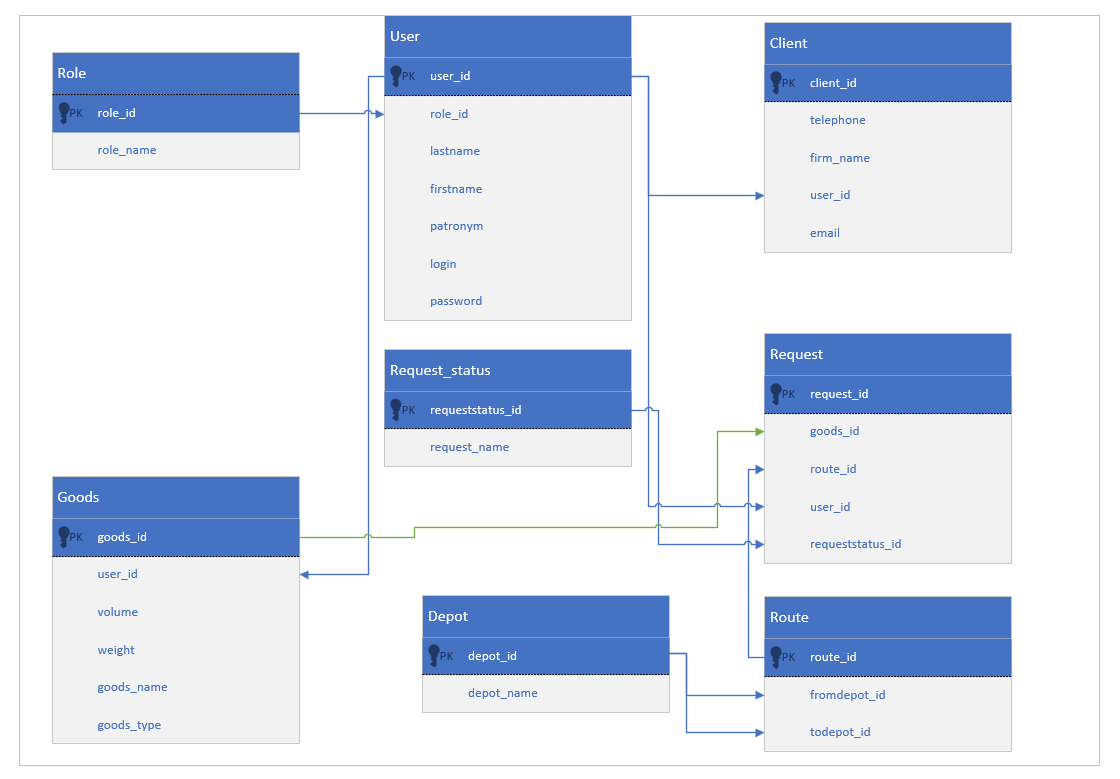
**СПИСОК ЛИТЕРАТУРЫ**

1. Евдокимов, П. В. C# на примерах : учебное пособие / П. В. Евдокимов. - 2-е изд. - Санкт-Петербург : Наука и Техника, 2021. - 320 с.
2. Полякова, Л. Н. Основы SQL : учебное пособие / Л. Н. Полякова. — 2-е изд. — Москва : ИНТУИТ, 2020. — 273 с.
3. Тюкачев, Н. А. C#. Алгоритмы и структуры данных : учебное пособие для спо / Н. А. Тюкачев, В. Г. Хлебостроев. - Санкт-Петербург, 2022. - 232 с.
4. Программирование на C#: самоучитель для начинающих / З.А. Шалтыкова - БХВ-Петербург 2023г. - 496с.
5. C# 9.0 и .NET 5. Пошаговое руководство / А.Ю. Абрамов - ДМК Пресс - 2022г. - 816с.
6. WPF. Разработка приложений для Windows / А.Ю. Абрамов - ДМК Пресс 2021г. - 752с.
7. C# 7.0. Карманный справочник / Н.В. Макаров - Питер 2017г. - 304с.
8. Microsoft Windows Presentation Foundation / А.Ю. Абрамов - ДМК Пресс 2014г. - 736с.
9. Разработка REST API с помощью ASP.NET Core / А.Ю. Абрамов - ДМК Пресс 2020г. - 640с.
10. Паттерны проектирования. Классическое руководство / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес - Addison-Wesley Professional 2015г. - 416с.
11. Чистый код. Создание, анализ и рефакторинг / Роберт Мартин - Prentice Hall 2010г. - 432с. Мурлин, А. Г. Объектно-ориентированное программирование : учебное пособие / А. Г. Мурлин, В. А. Мурлина, М. В. Янаева. — Краснодар : КубГТУ, 2022. — 151 с.

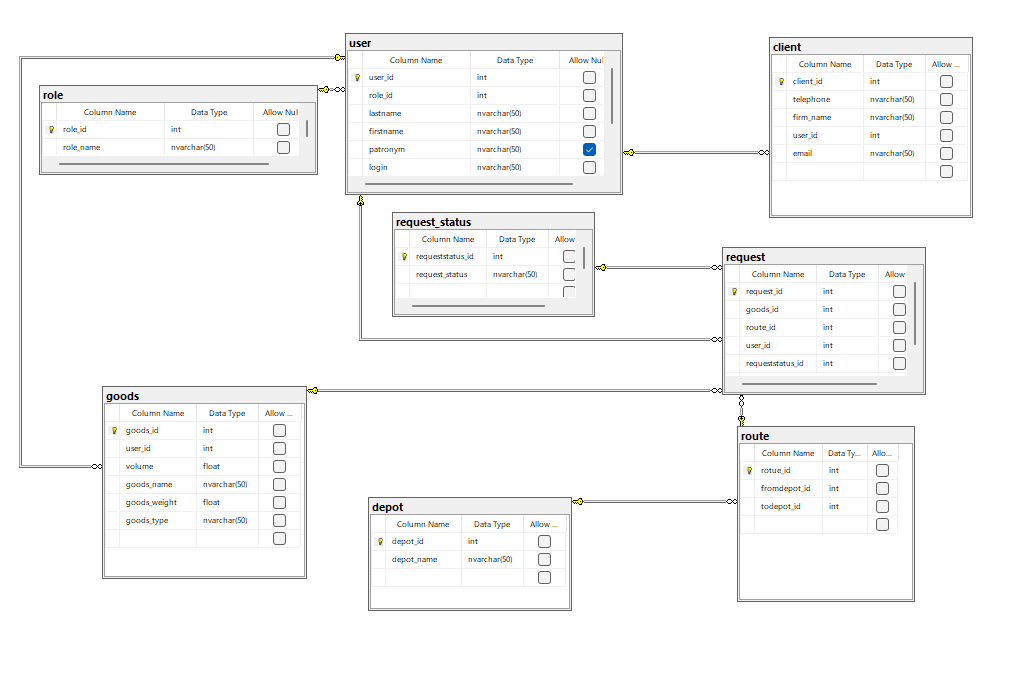
Электронные ресурсы

1. Разработка программных модулей программного обеспечения для компьютерных систем. Прикладное программирование : учебное пособие для спо Белугина С.В. — 3-е изд., — 2022. — (https://e.lanbook.com/book/200390).
2. Базы данных. Проектирование, программирование, управление и администрирование : учебник для вузов Волк В.К. — 3-е изд., — 2022. —(https://e.lanbook.com/book/193373).
3. Програмирование. Структурное програмирование, подпрограммы, строки : учебное пособие — Зайцев М.Г. — 2018. — (https://e.lanbook.com/book/118260).
4. Программирование : учебное пособие — Зайцев М. Г. — 2019. —(https://e.lanbook.com/book/118261).
5. Основы объектно-ориентированного программирования на базе языка C# : учебное пособие для спо — Залогова Л. А. — 2-е изд. — 2021. —(https://e.lanbook.com/book/164956).
6. Разработка баз данных в MS SQL Server 2018 : учебное пособие — Осетрова И.С. — 2019. — (https://e.lanbook.com/book/110470).
7. Проектирование информационных систем : монография — Остроух А. В., Суркова Н.Е. — 2-е изд. — 2021. — (https://e.lanbook.com/book/175513).
8. Основы тестирования программного обеспечения : учебное пособие для спо — Старолетов С.М. — 2-е изд. — 2022. (https://e.lanbook.com/book/189493).
9. Очерки истории информатики: введение в специальность : учебное пособие — Тынкевич М.А., Пимонов А.Г., Тайлакова А.А. — 2019. — (https://e.lanbook.com/book/133882).
10. Программирование и алгоритмизация : учебно-методическое пособие — Юрина Т.А. — 2021. — (https://e.lanbook.com/book/179228

Приложение 1. Полноразмерная схема базы данных, разработанная в Microsoft Office Visio



Приложение 2. Диаграмма базы данных, разработанная в Microsoft SQL Server Management Studio



Приложение 3. Use-case диаграмма, разработанная в Microsoft Office Visio

