|  |
| --- |
|  |
| (наименование образовательного учреждения) |

|  |
| --- |
| **ДОПУСКАЮ К ЗАЩИТЕ** |
| **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| Заместитель директора по УПР О.В.Корешков  *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* |
| (дата) |
|  |

**ДИПЛОМНАЯ РАБОТА**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Создание автоматизированной информационной системы для управления процессом городского планирования и развития. | | | | |
| (тема) | | | | |
| Выпускная квалификационная работа должна быть выполнена в виде: | | | | |
| дипломной работы и демонстрационного экзамена | | | | |
| студентом группы | | 4ИСП9-14 | | |
|  | | (номер группы) | | |
| Максимом Павловичем Кириллиным | | |  | |
| (И. О. Фамилия) | | | (подпись, дата) | |
| Основная профессиональная образовательная программа по специальности | | | | |
| 09.02.07 Информационные системы и программирование | | | | |
| (шифр и наименование специальности) | | | | |
| Форма обучения | | очная | | |
| Руководитель |  | преподаватель |  | Олеся Павловна Куропаткина |
|  | (ученая степень, должность, И. О. Фамилия) | |  | |
|  |  | | (подпись, дата) | |
|  |  | |  | |
| Председатель предметной (междисциплинарной, модульной) комиссии | | | | |
| Кирилл Михайлович Бастрыкин | | |  |  |
| (И. О. Фамилия) | | |  | (подпись, дата) |

Москва

2024

|  |
| --- |
| титул |

полное название образовательного учреждения

|  |
| --- |
| **УТВЕРЖДАЮ** |
| **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| Заместитель директора по УПР О.В. Корешков  04 апреля 2023 года |
| (дата) |
| **ЗАДАНИЕ НА ДИПЛОМНУЮ РАБОТУ**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Студенту | *Кириллину Максиму Павловичу* | | | | | (фамилия, имя, отчество полностью) | | | | | | I. Тема выпускной квалификационной работы | | | *Создание автоматизированной информационной системы для управления процессом городского планирования и развития.* | | | II. Срок сдачи студентом законченной работы | | | | дд мм 2024 г. | | III. Исходные данные | | *Программные средства: Microsoft Office Visio 2019,* | | | | *СУБД SQL Server Management Studio 2022, среда разработки Visual Studio 2022, среда разработки Android Studio* | | | | |   IV. Перечень подлежащих разработке вопросов   |  |  | | --- | --- | | 1. | *Анализ особенностей предметной области* | | 2. | *Сравнительный анализ программных средств* | | 3. | *Проектирование и разработка автоматизированной информационной системы (создание Use Case диаграммы, ER диаграммы, прототипа настольного приложения)* | | 4. | *Разработка автоматизированной информационной (создание базы данных в*  *СУБД MS SQL Server, настольного приложения на языке программирования C# в Visual Studio), (мобильного приложения на языке программирования Java в Android studio),(создание API на языке программирования С# в Visual Studio).* |   V. Перечень графического/иллюстрационного материала   |  |  | | --- | --- | | 1. | *Use case диаграмма* | | 2. | *ER диаграмма базы данных, разработанная в Microsoft Office Visio* | | 3. | *Диаграмма базы данных, разработанная в СУБД MS SQL Server* | | 4. | *Скриншоты экрана разработанного настольного приложения* | | 5. | *Скриншоты экрана разработанного мобильного приложения* | | 6. | *Презентация* |   VI. Дата выдачи задания «дд» мм 2023 г.   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Руководитель | |  | | Куропаткина О.П. | |  | | | (подпись) |  | | Задание принял к исполнению |  | | | Кириллин М.П. | |  | | | (подпись) |  |   «дд» мм 2024 г. |



КАЛЕНДАРНЫЙ ПЛАН ВКР

для специальности 09.02.07 Информационные системы и программирование

(группа 4ИСП9-14)

|  |  |  |
| --- | --- | --- |
| **Утверждение темы и руководителя ВКР** | |  |
| **Выдача задания на ВКР** | |  |
| **Сроки преддипломной практики** | |  |
|  | Выполнение задания по теме ВКР |  |
|  | Предоставление отчета по практике руководителю |  |
|  | Аттестация по практике |  |
| **Подготовка ВКР** | |  |
|  | Подбор и анализ исходной информации |  |
|  | Подготовка и утверждение плана (оглавления) ВКР |
|  | Работа над разделами (главами) и устранение замечаний руководителя ВКР |  |
|  | Согласование содержания ВКР, устранение замечаний |  |
|  | **Оформление и представление руководителю полного текста ВКР на проверку** |  |
|  | **Сдача демонстрационного экзамена** |  |
|  | **Подготовка к защите, получение допуска к защите ВКР** |  |

**Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** **Куропаткина О.П.**

План принял к исполнению

«дд» мм 2024 г. \_\_\_\_\_\_\_\_\_\_\_\_\_ Кириллин М.П.

**ОГЛАВЛЕНИЕ**

[Введение 5](#_Toc136687272)

[Глава 1. Анализ предметной области и применяемых программных средств 6](#_Toc136687273)

[1.1. Анализ особенностей предметной области 6](#_Toc136687274)

[1.2. Анализ применяемых программных средств 7](#_Toc136687275)

[1.2.1 Анализ системы управления баз данных Microsoft SQL Server 2019 7](#_Toc136687276)

[1.2.2 Анализ программы для разработки диаграмм Microsoft Visio 8](#_Toc136687277)

[1.2.3 Анализ объектно-ориентированного языка программирования C# 10](#_Toc136687278)

[1.2.4 Анализ интегрированной среды разработки Visual Studio 2019 11](#_Toc136687279)

[1.2.5 Анализ интегрированной среды разработки Android Studio 12](#_Toc136687279)

[Глава 2. Разработка автоматизированной информационной системы тюнинг ателье 14](#_Toc136687280)

[2.1 Логическое проектирование базы данных 14](#_Toc136687281)

[2.2 Разработка базы данных для информационной системы 16](#_Toc136687282)

[2.3 Разработка визуального интерфейса и функционала системы 21](#_Toc136687283)

[Заключение 26](#_Toc136687284)

[Список литературы 27](#_Toc136687285)

[Приложение 1. ER-диаграмма 29](#_Toc136687286)

[Приложение 2. ERD-диаграмма базы данных 30](#_Toc136687288)

# **ВВЕДЕНИЕ**

Городское планирование — это процесс разработки и реализации стратегий и планов, нацеленных на улучшение использования городской территории, обеспечение экономического развития и качества жизни горожан. В современном мире крайне важной является работа автоматизированных систем, ведь благодаря им большое количество процессов можно оптимизировать, повысить эффективность. Многие процессы в целом невозможны без автоматизированных систем, так как человек просто не может обрабатывать такой объем данных без посторонней помощи компьютера.

Таким образом удастся сэкономить время и исключить ошибки, более того можно автоматизировать и сделать удобным весь процесс управления какой-либо системой.

Для информационной системы управления процессом городского планирования и развития, она может быть полезна следующим:

* понятное и доступное описание всех видов планирования;
* учет объектов недвижимости и земельных участков;
* управление заказами;
* наглядное расписание предоставляемых услуг.

Актуальность данной курсовой работы обусловлена необходимостью создания автоматизированной системы для городского планирования и повышения её эффективности.

Целью курсовой работы является систематизация и закрепление полученных теоретических знаний и практических умений, проектирование и разработка базы данных для учёта планируемых и реализованных объектов недвижимости, для организации, осуществляющей услуги по планированию районов и их урбанизации. Создание настольного приложения для учёта и продажи недвижимости, создание мобильного приложения для возможности просмотра недвижимости.

Для достижения цели были поставлены следующие задачи:

* изучение особенностей конкретной предметной области, относящихся к теме курсовой работы;
* анализ программных средств с обоснованием выбора;
* проектирование и разработка базы данных в СУБД;
* создание объектов базы (представления, хранимые процедуры, пользовательские функции, триггеры) данных для автоматизации работы;
  + анализ полученных результатов работы разработанного программного обеспечения;
  + проектирование и разработка настольного приложения;
  + проектирование и разработка мобильного приложения;
  + создание клиент-серверной системы через API, для связи приложений с базой данных.

Объектом исследования является система управления процессом городского планирования и развития.

Предметом исследования является создание автоматизированной информационной системы для управления процессом городского планирования и развития.

# **ГЛАВА 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПРИМЕНЯЕМЫХ ПРОГРАММНЫХ СРЕДСТВ**

1.1. Анализ особенностей предметной области

В рамках данной курсовой работы рассматривается предметная область организации, осуществляющей управление процессом городского планирования, развития и продажи недвижимости.

Основной деятельностью организации, осуществляющей процесс городского планирования, развития и продажи недвижимости является предоставление информации о недвижимости, земельных участках и возможности её приобретения. К таким услугам относятся:

* возможность просмотра доступных строений и земельных участков со стороны организации;
* возможность планировки строений со стороны организации;
* просмотр информации о недвижимости со стороны организации и клиента;
* покупка недвижимости со стороны клиента.

Данная автоматизированная система предназначена для оптимизации работы сотрудников и для удобства выбора недвижимости организации. Сотрудники будут разделены в соответствии с их должностями.

Роль менеджера – учёт, распределение и продажа недвижимости клиентам (физические и юридические лица).

Кадастровый инженер – это специалист, который производит межевание территории и вносит актуальную информацию в систему.

Архитектор – специалист, который создаёт планировки домов и квартир, внутри системы архитектор сможет добавлять планировки домов и квартир, указывать информацию о них.

Клиент – юридическое или физическое, на которого ориентирован весь проект. Внутри системы он сможет просматривать информацию о районе, домах и земельных участках, в следствии выбрать недвижимость для покупки или аренды.

Планируется реализация клиент-серверной части через разработку API на базе ASP.NET, удобного инструмента внутренней инфраструктуры Visual Studio. API необходима для привязки к серверу как настольного так и мобильного приложения.

Анализ предметной области для учета планируемых и реализованных объектов недвижимости включает в себя анализ факторов, влияющих на успешность реализации объектов недвижимости. Кроме того, важно иметь систему отчетности, которая позволяет отслеживать выполнение плановых показателей и анализировать результаты проектов недвижимости.

1.2. Анализ применяемых программных средств

Для разработки информационной системы будет использоваться ряд программных средств, а именно:

* Microsoft Visio;
* Microsoft SQL Server 2019;
* язык программирования C#;
* Visual Studio 2019;
* Android Studio.

Необходимо рассмотреть данные продукты более подробно.

1.2.1 Анализ программы для разработки диаграмм Microsoft Visio

Microsoft Visio - векторный графический редактор, редактор диаграмм и блок-схем, который является частью составного пакета Microsoft Office. Программа предназначена для создания различного вида чертежей: от схем до календарей.

В программе представлено множество различных фигур, как простых, так и сложных. Каждый шаблон предназначен для определенной цели - от создания планов водопроводных сетей до компьютерных сетей. Для поиска сведений о предназначении шаблона является специальное окошко "Поиск". Программа выпускается в двух комплектациях Office Visio Professional и Office Visio Standard. Для черчения схем вполне достаточно Office Visio Standard.

Основные функции

* Изменение масштаба документа и единиц измерения линейки;
* Добавление фона, слоя или подложки;
* Форматирование фигур (линия, заливка, эффекты);
* Добавление данных в фигуру;
* Использование гистограмм, наборов значков, рисунков, связанных с данными;
* Автоматическое создание организационной диаграммы на основе данных сотрудников.

Microsoft Visio универсальная программа, которая более проста в работе, она позволяет легко начертить не только электрическую схему, но и выполнить другие необходимые в работе инженера функции. Программа отличается простотой использования, и не требует специальной подготовки для работы с ней, а также обладает невысокой стоимостью.

Недостатки программы

* Отсутствие в составе полноценной библиотеки условных графических обозначений (УГО) элементов электрических схем;
* Набор элементов не соответствует ГОСТ (GOST Electro for Visio дополнительная платная функция);
* Авто нумерация (требуется специальный алгоритм);
* Создание блоков;
* Новый созданный слой добавляется только на текущую страницу, а не на все страницы файла;
* Отсутствие расчетных функций.

1.2.2 Анализ системы управления баз данных Microsoft SQL Server 2019

SQL Server – это программа, которая предназначена для хранения и обработки данных. При взаимодействии с ней пользователи могут отправлять запросы и получать ответы – причем как локально, так и по сети. Функционирует программа следующим образом: открывает сетевой порт, принимает команды и выдает результат.

Для работы по локальной сети СУБД устанавливается на рабочем устройстве юзера. Режим работы зависит от выбранной системы. По умолчанию пользователи взаимодействуют с базой данных MS Access. Такой режим подходит в тех случаях, если с БД будут работать не более 10-12 человек.

Для работы в сетевой среде файл базы данных заливается на общедоступный ресурс. И пользователи подключаются к нему со своих ПК (на каждом устройстве устанавливается специальное программное обеспечение).

Если с базой данных будут работать более 12 человек одновременно или если обнаружится, что сервер не выдерживает нагрузки, лучше перевести БД в формат Microsoft SQL. В данном режиме все вычисления происходят на стороне сервера, а не на компьютере сотрудника, что гарантирует высокое быстродействие даже при одновременном подключении большого количества пользователей.

Рассмотрим, какие у Microsoft SQL Server преимущества и недостатки. Это нужно, чтобы вы понимали, насколько данная СУБД подходит для ваших целей.

Основные достоинства:

Масштабирование системы. Взаимодействовать с ней можно как на простых ноутбуках, так и на ПК с мощным процессором, который способен обрабатывать большой объем запросов.

Размер страниц – до 8 Кб. Данные извлекаются быстро, а сложную информацию удобнее хранить. Система обрабатывает транзакции в интерактивном режиме, есть динамическая блокировка.

Автоматизация рутинных административных задач. Например, управление блокировками и памятью, редактура размеров файлов. В программе продуманы настройки, можно создавать профили пользователей.

Удобный поиск. Его можно осуществлять по фразам, словам, тексту либо создавать ключевые индексы.

Поддержка работы с другими решениями Майкрософт, в том числе с Excel, Access.

Также в программе предусмотрена синхронизация, есть репликации через интернет, службы преобразования информации и полноценный web-ассистент для форматирования страниц. Дополнительно в нее интегрирован сервис интерактивного анализа (можно принимать решения, создавать корпоративные отчеты).

Основные минусы:

Зависимость от ОС. Система работает только с Windows.

Высокая стоимость.

1.2.3 Анализ объектно-ориентированного языка программирования C#

C# — это объектно-ориентированный язык программирования. Его разработка велась в 1998-2001 годах под руководством группы программеров из корпорации Microsoft. Изначально он рассматривался как средство создания утилит для платформ Microsoft .NET Framework и .NET Core.

Некоторые полагают, что С# – это просто версия С или C++. Данное утверждение неверное. Соответствующий язык программирования был создан «с нуля».

C # изначально был придуман компанией Microsoft для собственных целей и служб. Он предусматривает следующие преимущества:

* строгую типизацию;
* сохранение концепций объектно-ориентированного программирования;
* функциональность;
* достаточно мощный инструментарий;
* стабильную работу через Visual Studio;
* компактный и легко читаемый код;
* понятный даже новичкам синтаксис.

При использовании этого языка можно насладиться обработкой исключений, а также наличием сборщика мусора. Здесь все продумано так, чтобы программеру было легко писать и считывать итоговые кодификации.

Синтаксис языка чем-то напоминает не только С и С++, но и Java.

Автоматическая «сборка мусора» Это значит, что нам в большинстве случаев не придётся заботиться об освобождении памяти. Вышеупомянутая общеязыковая среда CLR сама вызовет сборщик мусора и очистит память.

Низкий порог вхождения. Синтаксис C# имеет много схожего с другими языками программирования, благодаря чему облегчается переход для программистов. Язык C# часто признают наиболее понятным и подходящим для новичков.

1.2.4 Анализ интегрированной среды разработки Visual Studio 2019

Microsoft Visual Studio — это программная среда по разработке приложений для ОС Windows, как консольных, так и с графическим интерфейсом.

Интегрированная среда разработки (IntegratedDevelopmentEnvironment - IDE) Visual Studio предлагает ряд высокоуровневых функциональных возможностей, которые выходят за рамки базового управления кодом.

Ниже перечислены основные преимущества IDE-среды Visual Studio.

Встроенный Web-сервер. Для обслуживания Web-приложения ASP.NET необходим Web-сервер, который будет ожидать Web-запросы и обрабатывать соответствующие страницы. Наличие в Visual Studio интегрированного Web-сервера позволяет запускать Web-сайт прямо из среды проектирования, а также повышает безопасность, исключая вероятность получения доступа к тестовому Web-сайту с какого-нибудь внешнего компьютера, поскольку тестовый сервер может принимать соединения только с локального компьютера.

Меньше кода для написания. Для создания большинства приложений требуется приличное количество стандартного стереотипного кода, и Web-страницы ASP. NET тому не исключение. Например, добавление Web-элемента управления, присоединение обработчиков событий и корректировка форматирования требует установки в разметке страницы ряда деталей. В Visual Studio такие детали устанавливаются автоматически.

Интуитивный стиль кодирования. По умолчанию Visual Studio форматирует код по мере его ввода, автоматически вставляя необходимые отступы и применяя цветовое кодирование для выделения элементов типа комментариев. Такие незначительные отличия делают код более удобным для чтения и менее подверженным ошибкам. Применяемые Visual Studio автоматически параметры форматирования можно даже настраивать, что очень удобно в случаях, когда разработчик предпочитает другой стиль размещения скобок (например, стиль K&R, при котором открывающая скобка размещается на той же строке, что и объявление, которому она предшествует).

Более высокая скорость разработки. Многие из функциональных возможностей Visual Studio направлены на то, чтобы помогать разработчику делать свою работу как можно быстрее. Удобные функции, вроде функции IntelliSense (которая умеет перехватывать ошибки и предлагать правильные варианты), функции поиска и замены (которая позволяет отыскивать ключевые слова как в одном файле, так и во всем проекте) и функции автоматического добавления и удаления комментариев (которая может временно скрывать блоки кода), позволяют разработчику работать быстро и эффективно.

1.2.5 Анализ интегрированной среды разработки Android Studio

Android Studio – это интегрированная среда разработки для работы с платформой Android. ПО было анонсировано 16 мая 2013 года компанией Google. Софт помогает разрабатывать разнообразные приложения и игры под Android.

Это – официальная среда программирования, находящаяся под поддержкой Google. В основе лежит IntelliJ IDEA от JetBeans.

Выделяется «студия» следующими особенностями:

* наличие встроенного эмулятора;
* мощный функционал и инструментарий для разработчиков;
* встроенный отладчик;
* понятный и хорошо продуманный интерфейс;
* документация на русском языке;
* множество туториалов и уроков, при помощи которых удастся быстро освоить платформу;
* «горячие клавиши»;
* возможность настроить Android Studio за несколько минут;
* совместимость с большинством популярных ОС.

Среда Android Studio предназначена как для небольших команд разработчиков мобильных приложений (даже в количестве одного человека), или же крупных международных организаций с GIT или другими подобными системами управления версиями. Опытные разработчики смогут выбрать инструменты, которые больше подходят для масштабных проектов. Решения для Android разрабатываются в Android Studio с использованием Java или C++.

# **ГЛАВА 2. РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ ТЮНИНГ АТЕЛЬЕ**

В первой главе была рассмотрена предметная область тюнинг ателье и приведены основные моменты.

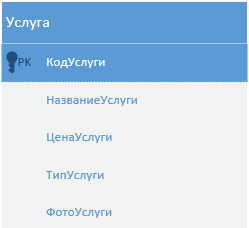
В качестве инструментов разработки было выбрано следующее программное обеспечение: Microsoft Visio, Microsoft SQL Server, Visual Studio.

## 2.1 Логическое проектирование базы данных

**Логическое проектирование – это процесс конструирования информационной модели на основе существующих моделей данных, не зависимо от используемой СУБД и других условий физической реализации.**

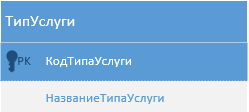
**Для логического проектирования используется Microsoft Visio.**

**Первая модель нашей системы «Услуга»(рис 2.1.1):**

****

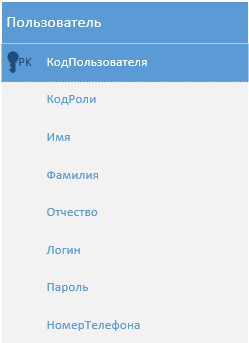
**Рисунок 2.1.1 Модель «Услуга»**

**Данная модель включает в себя название и основные характеристики, а также тип услуги (рис 2.1.2):**

****

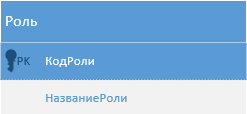
**Рисунок 2.1.2 Тип услуги**

**Помимо оказываемых услуг необходимо предусмотреть возможность хранения информации о пользователе (рис 2.1.3):**

****

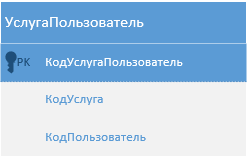
**Рисунок 2.1.3 Модель пользователя**

**Модель пользователя включает в себя роль(рис 2.1.4):**

****

**Рисунок 2.1.4 Модель роли**

**И еще одна важная модель услуги, которые заказал пользователь (рис 2.1.5):**

****

**Рисунок 2.1.5 Услуги пользователя.**

**Полноценную ER-диаграмму можно посмотреть в приложении 1.**

## 2.2 Разработка базы данных для информационной системы

Для разработки базы данных используется программное обеспечение Microsoft SQL Server.

В соответствии с ER-диаграммой была создана сущность «услуги» (рис 2.2.1):

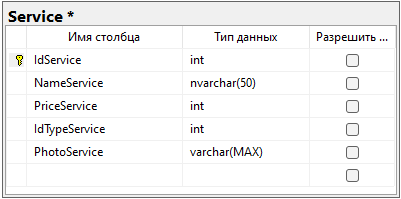


Рисунок 2.2.1 Сущность «Услуга»

Данная сущность включает в себя код услуги, название услуги, приблизительная стоимость, тип услуги и фото для ознакомления.

Тип услуги является отдельной сущностью (рис 2.2.2):

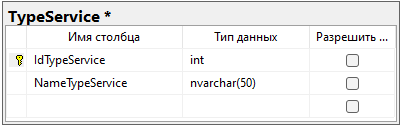


Рисунок 2.2.2 Сущность «Тип услуги»

Данная сущность включает в себя только код и название. Следующая важная сущность – это пользователи (рис 2.2.3). О пользователях необходимо хранить достаточно большой объем информации, сюда входит: фамилия, имя и отчество пользователя, данные для авторизации в системе, роль пользователя, так как это может быть сотрудник, а может быть клиент, а также номер телефона для связи.

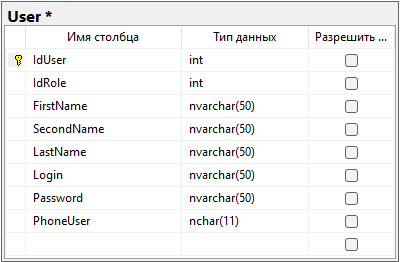


Рисунок 2.2.3 Сущность «Пользователь»

Исходя из того, что пользователи могут иметь различные роли в системе, создана отдельная сущность «роли» (рис 2.2.4)

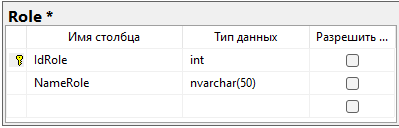


Рисунок 2.2.4 Сущность «Роль»

Для правильного функционирования системы необходимо создать промежуточную таблицу, чтобы была возможность отслеживать какая услуга оказывается какому пользователю и каким сотрудником (рис 2.2.5):

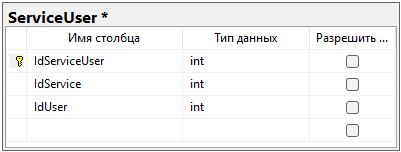


Рисунок 2.2.5 Сущность «Услуга пользователя»

После создания всех основных сущностей необходимо создать и связи между ними.

Первая и очень важная связь, это связь пользователь – роль(рис 2.2.6):

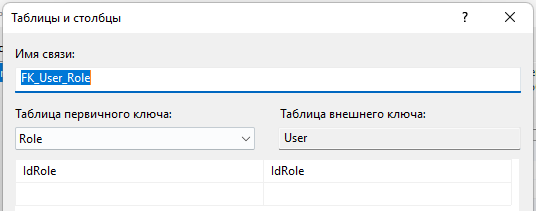


Рисунок 2.2.6 Связь «Пользователь» - «Роль»

Так как выше было определено, что необходима дополнительная сущность услуги пользователя, необходимо предусмотреть и связи этой сущность(рис 2.2.7 и 2.2.8):

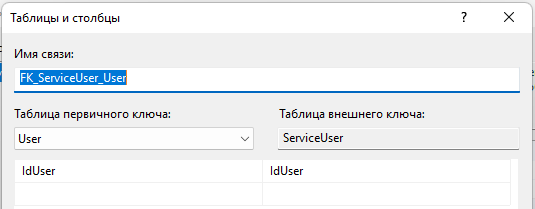


Рисунок 2.2.7 Связь «Пользователь» - «Услуга пользователя»

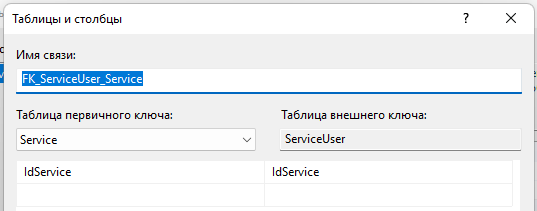


Рисунок 2.2.8 Связь «Услуга» - «Услуга пользователя»

И последняя связь — это связь услуги с типом оказываемой услуги (рис 2.2.9):

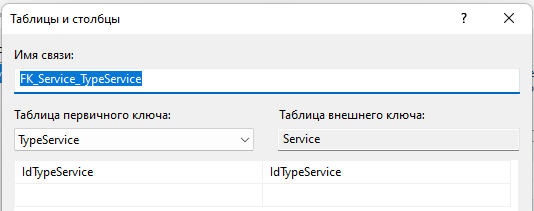


Рисунок 2.2.9 Связь «Услуга» - «Тип услуги»

Для проверки будущего функционала разрабатываемой системы были внесены тестовые данные.

Для начала необходимо заполнить сущности без которых заполнение основных сущностей не возможно, это роль пользователя(рис 2.2.10), тип услуги (рис 2.2.11):

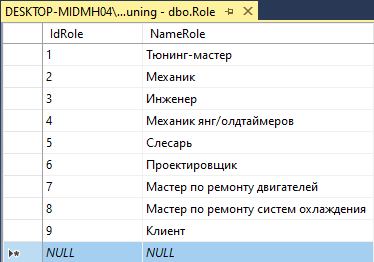


Рисунок 2.2.10 Тестовые данные сущности «Роль»

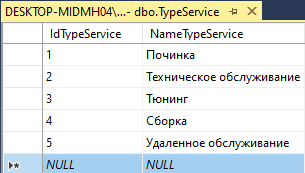


Рисунок 2.2.11 Тестовые данные сущность «Тип услуги»

Далее наполняем сущность пользователя (рис 2.2.12):

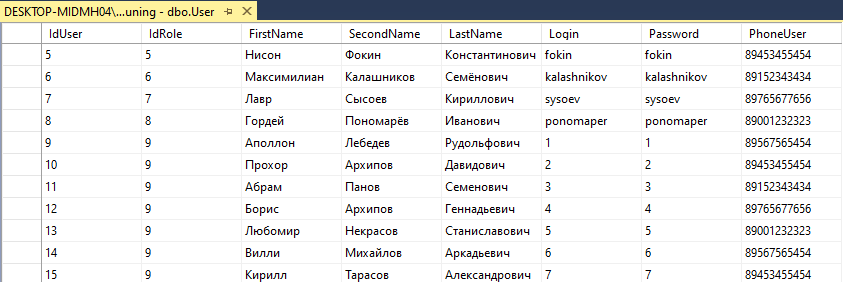


Рисунок 2.2.12 Тестовые данные для сущности «Пользователь»

Наша основная сущность по-прежнему «Услуги», поэтому наполняем тестовыми данными ее (рис 2.2.13):

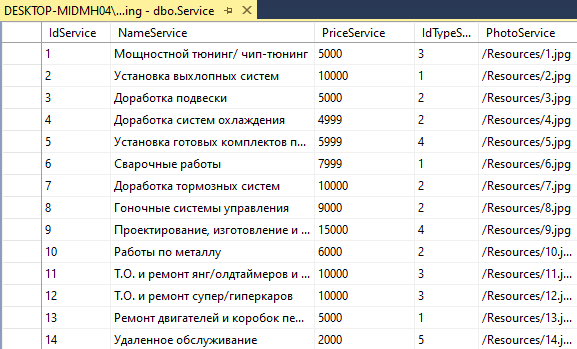


Рисунок 2.2.13 Тестовые данные сущности «Услуга»

Полная ERD-диаграмма представлена в приложении 2.

## 2.3 Разработка визуального интерфейса и функционала системы

Для разработки визуального интерфейса используется Visual Studio. Самое первое, что необходимо сделать, это создать правильную структуру проекта (рис 2.3.1):

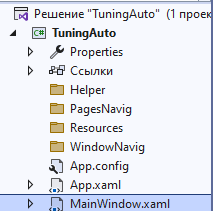


Рисунок 2.3.1 Структура проекта разрабатываемой системы

Далее необходимо подключить базу данных к проекту (рис 2.3.2):

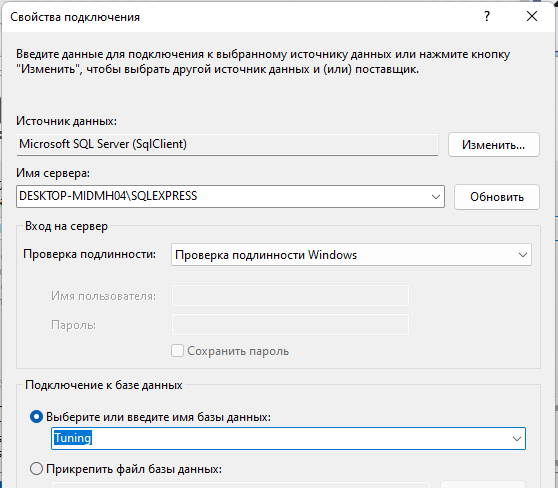


Рисунок 2.3.2 Подключение базы данных

Дополнительно, необходимо создать как минимум два вспомогательных класса, а именно: класс для работы с базой данных (рис 2.3.4) и класс для работы с фреймами (рис 2.3.3) так как в системе планируется постраничная навигация.

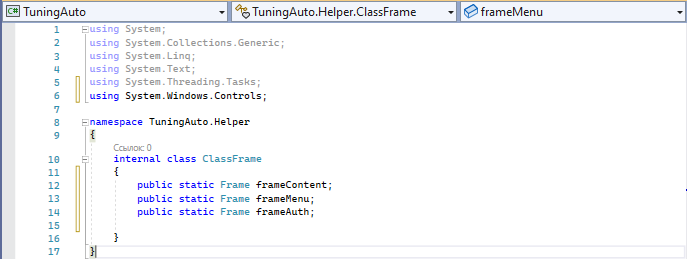


Рисунок 2.3.3 Код класса фреймов

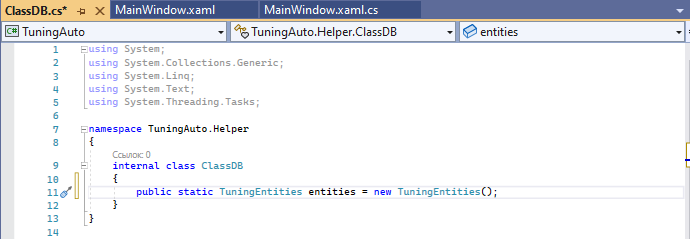


Рисунок 2.3.4 Код класса с подключением базы

Дальше идет разметка основного окна системы (рис 2.3.5):

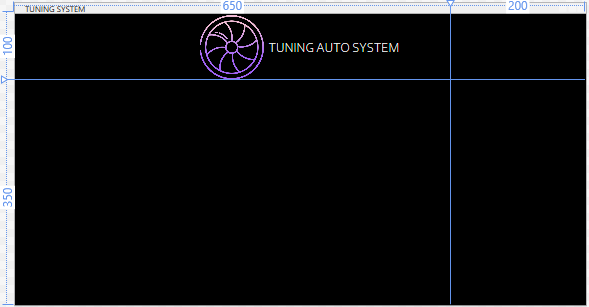


Рисунок 2.3.5 Разметка главного окна системы

Код разметки на рисунок 2.3.6:

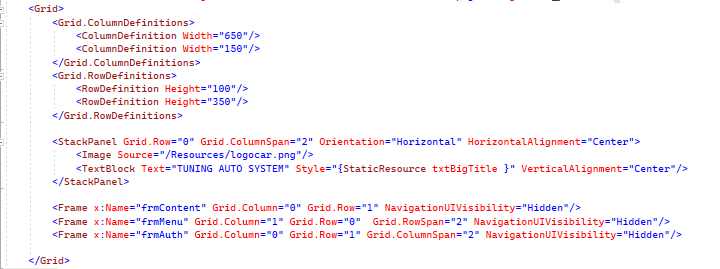


Рисунок 2.3.6 Листинг кода разметки WPF

На рисунке 2.3.7 код на языке С# для главного окна:

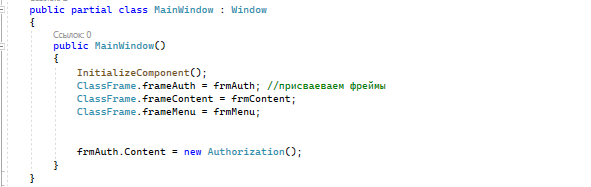


Рисунок 2.3.7 Код главного кода системы

Страница авторизации должна подгружаться на главное окно системы, поэтому разметка страницы авторизации, следующая (рис 2.3.8):



Рисунок 2.3.8 Разметка страницы авторизации

При запуске системы в режиме отладки страница подгружается на главное окно в предназначенный для этого фрейм (рис 2.3.9):

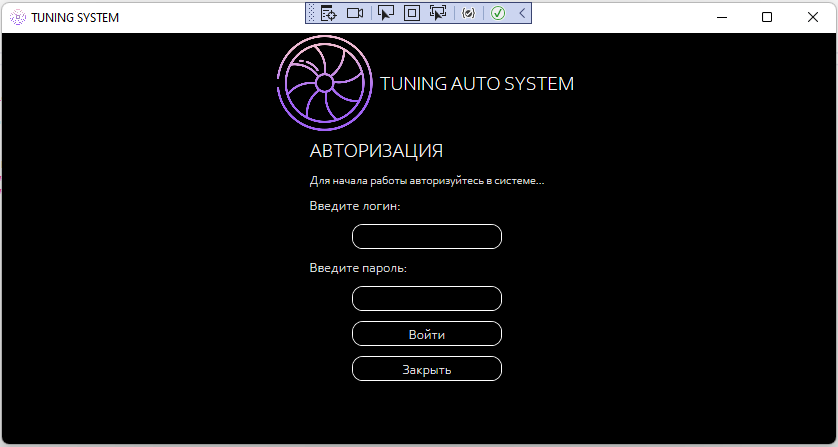


Рисунок 2.3.9 Начальное окно системы

После прохождения процедуры авторизации в специальный фрейм подгружается страница с меню справа (рис 2.3.10)

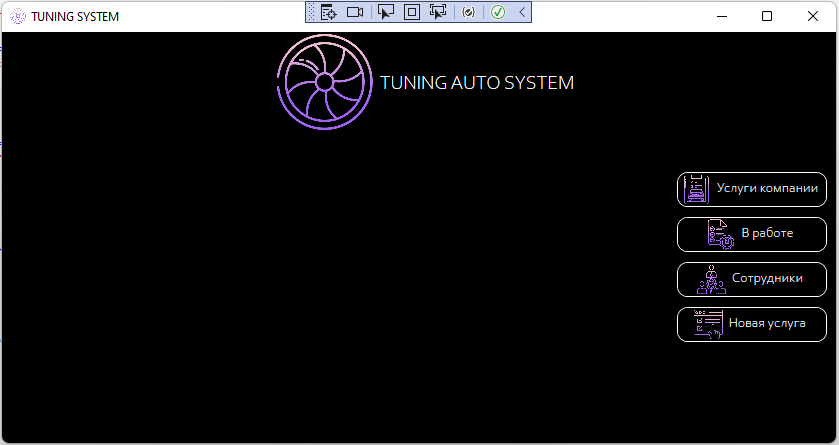


Рисунок 2.3.10 Главное окно системы с меню

Меню содержит в себе следующие пункты:

1. Услуги компании с полным перечнем предоставляемых компанией услугами
2. В работе – данный пункт подразумевает полный перечень услуг, находящихся в работе;
3. Сотрудники, чтобы иметь возможность оперативно добавлять или изменять существующих пользователей;
4. Новая услуга предполагает открытие нового окна меню.

При переходе на кнопку в меню во фрейме контента подгружаются страницы, например, при переходе по кнопке «услуги» (рис 2.3.11)



Рисунок 2.3.11 Кнопка меню «Услуги компании»

На страницу слева подгружается страница с услугами, которые предоставляет компания. В данном случае выводится картинка услуги и название, чуть ниже – цена услуги (рис 2.3.12)

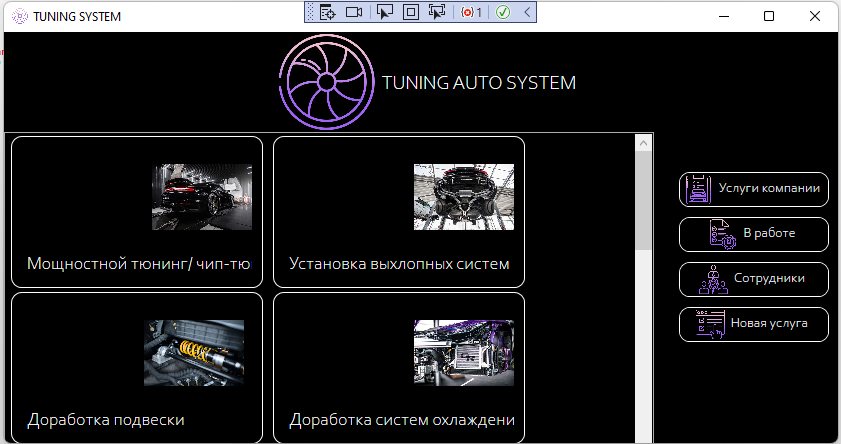


Рисунок 2.3.12 Страница «Услуги компании»

# **ЗАКЛЮЧЕНИЕ**

В первой главе данной выпускной работы была рассмотрена предметная область тюнинг сервиса с полным перечнем предоставляемых услуг. Далее выбраны программные инструменты для разработки будущей информационной системы.

После разбора и анализа применяемых программных средств и предметной области началась разработка самой системы.

Вторая глава началась с логического проектирования разрабатываемой информационной системы в программе Visio. Была создана ER-диаграмма информационной системы.

Дальше создана база данных для хранения информации в системе с помощью СУБД Microsoft SQL Server, включая все сущности, перечисленные при логическом проектировании, связи между сущностями чтобы система работала правильно и эффективно, а также наполнена тестовыми данными для проверки работоспособности основного функционала.

В конце разработан визуальный интерфейс информационной системы и основной функционал.

Цель выпускной квалификационной работы достигнута – разработана информационная система для автоматизации работы тюнинг ателье.

# **СПИСОК ЛИТЕРАТУРЫ**

1. Одерски, М. Scala. Профессиональное программирование / М. Одерски, Л. Спун, Б. Веннерс. - СПб.: Питер, 2018

2. Рихтер, Дж. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# / Дж. Рихтер. - СПб.: Питер, 2019.

3.  Эспозито, Д. Программирование с использованием Microsoft ASP. NET 4 / Д. Эспозито. - СПб.: Питер, 2019.

4. Бессмертный, И. А. Системы искусственного интеллекта : учеб. пособие для СПО / И. А. Бессмертный. — 2-е изд., испр. и доп. — М. : Издательство Юрайт, 2018.

5. Гордеев, С. И. Организация баз данных в 2 ч. Часть 2 : учебник для вузов / С. И. Гордеев, В. Н. Волошина. — 2-е изд., испр. и доп. — М. : Издательство Юрайт, 2019.

6. Зыков, С. В. Программирование. Объектно-ориентированный подход : учебник и практикум для академического бакалавриата / С. В. Зыков. — М. : Издательство Юрайт, 2019.

7. Иванов, В. М. Интеллектуальные системы : учеб. пособие для СПО / В. М. Иванов ; под науч. ред. А. Н. Сесекина. — М. : Издательство Юрайт, 2019.

8.  Ашарина, И.В. Основы программирования на языках С и С++: Курс лекций для высших учебных заведений / И.В. Ашарина. — М.: Гор. линия-Телеком, 2018.

9. Джейсон Visual C# .NET. Полное руководство / Джейсон, Майк Прайс; , Гандэрлой. - М.: Корона Принт, 2019.

10.Ник, Рендольф Visual Studio 2010 для профессионалов / Рендольф Ник. - М.: Диалектика / Вильямс, 2018.

11. Тюкачев Н.А., Хлебостроев В.Г. - C#. Алгоритмы и структуры данных: учебное пособие - Издательство "Лань" – 2018.

12. Бьянкуцци, Ф. Пионеры программирования. Диалоги с создателями наиболее популярных языков программирования / Ф. Бьянкуцци, Ш. Уорден. — М.: Символ, 2018.

Электронные ресурсы

1. Документация по Visual Studio https://docs.microsoft.com/ruru/visualstudio/install/install-visual-studio?view=vs-2019

## Приложение 1. ER-диаграмма

## 

## Приложение 2. ERD-диаграмма базы данных

