**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КОМИ**

**Государственное профессиональное образовательное учреждение**

**«Воркутинский арктический горно-политехнический колледж»**

**КУРСОВОЙ ПРОЕКТ**

По дисциплине МДК.07.01 Управление и автоматизация баз данных

**Разработка информационной системы «Ремонт мобильных устройств»**

Выполнил студент гр. ИСП-20 /\_\_\_\_\_\_\_\_\_\_\_\_\_/Ячменев Александр Павлович /

(подпись) (Ф.И.О.)

**ОЦЕНКА:\_\_\_\_\_\_\_\_\_\_\_\_\_**

Дата:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ПРОВЕРИЛ**

Научный руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_/ Егоров Данил Павлович/

(подпись) (Ф.И.О.)

Воркута

2023

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc153192825)

[ГЛАВА 1. ВЫБОР ИНСТРУМЕНТАРИЯ 4](#_Toc153192826)

[1.1 СУБД SQL Server 4](#_Toc153192827)

[1.2 Microsoft SQL Server Management Studio 5](#_Toc153192828)

[1.3 Entity Framework 6](#_Toc153192829)

[ГЛАВА 2. ПРОЕТИРОВАНИЕ БАЗЫ ДАННЫХ 8](#_Toc153192830)

[2.1 Разработка диаграммы ERD 8](#_Toc153192831)

[2.2 Разработка базы данных 9](#_Toc153192832)

[2.3 Словарь данных 13](#_Toc153192833)

[ГЛАВА 3. РАЗРАБОТКА БАЗЫ ДАННЫХ 16](#_Toc153192834)

[3.1 Введение в безопасность SQL Server 16](#_Toc153192835)

[3.2 Управление безопасностью уровня сервера 17](#_Toc153192836)

[3.3 Управление участниками уровня базы данных 20](#_Toc153192837)

[3.4 Управление разрешениями уровня базы данных 22](#_Toc153192838)

[3.5 Шифрование данных баз данных 23](#_Toc153192839)

[ЗАКЛЮЧЕНИЕ 25](#_Toc153192840)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 26](#_Toc153192841)

# ВВЕДЕНИЕ

При выполнении операций по регистрации поступления заказов на ремонт и их сдача используется труд ручной труд, а именно ­— ведутся книги прихода и расхода товаров. Очевидно, что использование программы значительно сократит время, затрачиваемое на регистрацию заказа. В конце каждого месяца ответственный работник мастерской составляет отчеты о поступивших и выполненных заказах. На эту операцию уходит 1 рабочий день, т.е. 6 ч. Формирование оборотной ведомости в компьютере займет 3 - 4 мин.

Экономический эффект от внедрения базы данных «Ремонт мобильных устройств» ожидается за счет сокращения времени на выполняемые менеджерами операции, исключения ошибок при формировании отчетов, увеличения времени на анализ хозяйственной деятельности и т.д., в результате чего клиент сможет быстрее получать заказ, а персоналу станет в целом комфортнее выполнять свою работу.

База данных для ИС «Ремонт мобильных устройств» предназначена для обобщения информации о сотовых телефонах и аксессуарах, поступивших в ремонт. Пользователями программы выступают специалист по ремонту мобильных устройств, администратор, работник склада и менеджер.

**Объект:** база данных для ИС «Ремонт мобильных устройств».

**Предмет:** автоматизация бизнес-процесса ремонта мобильных устройств.

**Цель:** разработка базы данных, которая обеспечит работу информационной системы «Ремонт мобильных устройств»

**Задачи**:

* выбрать инструментарий для создания базы данных для АИС «Ремонт мобильных устройств»;
* спроектировать базу данных;
* сделать разграничение ролей для базы данных;
* сделать шифрование данных для базы данных;
* сделать резервное копирование базы данных.

# ГЛАВА 1. ВЫБОР ИНСТРУМЕНТАРИЯ

# СУБД SQL Server

Microsoft SQL Server — [система управления реляционными базами данных (РСУБД)](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94), разработанная корпорацией [Microsoft](https://ru.wikipedia.org/wiki/Microsoft). Основной используемый язык запросов — [Transact-SQL](https://ru.wikipedia.org/wiki/Transact-SQL) (сокращённо T-SQL), создан совместно Microsoft и [Sybase](https://ru.wikipedia.org/wiki/Sybase). Transact-SQL является реализацией стандарта [ANSI](https://ru.wikipedia.org/wiki/%D0%90%D0%BC%D0%B5%D1%80%D0%B8%D0%BA%D0%B0%D0%BD%D1%81%D0%BA%D0%B8%D0%B9_%D0%BD%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%B8%D0%BD%D1%81%D1%82%D0%B8%D1%82%D1%83%D1%82_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BE%D0%B2)/[ISO](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D0%B6%D0%B4%D1%83%D0%BD%D0%B0%D1%80%D0%BE%D0%B4%D0%BD%D0%B0%D1%8F_%D0%BE%D1%80%D0%B3%D0%B0%D0%BD%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D0%BF%D0%BE_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8) по структурированному языку запросов ([SQL](https://ru.wikipedia.org/wiki/SQL)) с расширениями. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия; конкурирует с другими СУБД в этом сегменте рынка.

Используемая T-SQL является реализацией SQL-92 (стандарт ISO для SQL) с множественными расширениями. T-SQL позволяет использовать дополнительный синтаксис для хранимых процедур и обеспечивает поддержку транзакций (взаимодействие базы данных с управляющим приложением). Microsoft SQL Server и Sybase ASE для взаимодействия с сетью используют протокол уровня приложения под названием Tabular Data Stream (TDS, протокол передачи табличных данных). Протокол TDS также был реализован в проекте FreeTDS с целью обеспечить различным приложениям возможность взаимодействия с базами данных Microsoft SQL Server и Sybase.

Microsoft SQL Server также поддерживает Open Database Connectivity (ODBC) — интерфейс взаимодействия приложений с [СУБД](https://www.tadviser.ru/index.php/%D0%A1%D0%A3%D0%91%D0%94). Версия SQL Server 2005 обеспечивает возможность подключения пользователей через веб-сервисы, использующие протокол SOAP. Это позволяет клиентским программам, не предназначенным для [Windows](https://www.tadviser.ru/index.php/%D0%9F%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82:Microsoft_Windows), кроссплатформенно соединяться с SQL Server. Microsoft также выпустила сертифицированный драйвер JDBC, позволяющий приложениям под управлением [Java](https://www.tadviser.ru/index.php/%D0%9F%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82:Java) (таким как BEA и [IBM](https://www.tadviser.ru/index.php/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B0%D0%BD%D0%B8%D1%8F:IBM) WebSphere) соединяться с Microsoft SQL Server 2000 и 2005.

SQL Server поддерживает зеркалирование и кластеризацию баз данных. Кластер сервера SQL — это совокупность одинаково конфигурированных серверов; такая схема помогает распределить рабочую нагрузку между несколькими серверами. Все сервера имеют одно виртуальное имя, и данные распределяются по IP-адресам машин кластера в течение рабочего цикла. Также в случае отказа или сбоя на одном из серверов кластера доступен автоматический перенос нагрузки на другой сервер.

# Microsoft SQL Server Management Studio

SQL Server Management Studio (SSMS) — утилита из Microsoft SQL Server 2005 и более поздних версий 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.

Также есть SQL Server Management Studio Express для Express версии сервера, которая является бесплатной. Однако в ней нет поддержки ряда компонентов (Analysis Services, Integration Services, Notification Services, Reporting Services) и SQL Server 2005 Mobile Edition.

Начиная с версии 16.5.3 пакет SSMS выделен в отдельный обновляемый продукт, доступный для скачивания на сайте Microsoft. Текущая доступная версия SSMS 18.4 (15.0.18206.0) (general availability) поддерживает MS SQL server начиная с версии 2008 по 2019.

Среда SQL Server Management Studio – это основной, стандартный и полнофункциональный инструмент для работы с Microsoft SQL Server, разработанный компанией Microsoft, который предназначен как для разработчиков, так и для администраторов SQL Server.

С помощью SSMS Вы можете разрабатывать базы данных, выполнять инструкции T-SQL, а также администрировать Microsoft SQL Server.

Если в Ваши задачи входит полное сопровождение Microsoft SQL Server, начиная от создания баз данных, написания SQL запросов, создания хранимых процедур и функций, и заканчивая администрированием SQL Server, включая управление безопасностью, то основным Вашим инструментом будет как раз среда SQL Server Management Studio.

Среда SQL Server Management Studio реализована только под Windows, поэтому если Вам нужен инструмент для работы с Microsoft SQL Server, который будет работать на других платформах, например, на Linux или macOS, то Вам следует использовать инструмент Azure Data Studio, который также является официальным инструментом, разработанным компанией Microsoft.

# Entity Framework

ADO.NET Entity Framework (EF) — объектно-ориентированная технология доступа к данным, является object-relational mapping (ORM) решением для .NET Framework от Microsoft. Предоставляет возможность взаимодействия с объектами как посредством LINQ в виде LINQ to Entities, так и с использованием Entity SQL. Для облегчения построения web-решений используется как ADO.NET Data Services (Astoria), так и связка из Windows Communication Foundation и Windows Presentation Foundation, позволяющая строить многоуровневые приложения.

**Entity SQL** представляет собой язык, подобный языку [SQL](https://ru.wikipedia.org/wiki/SQL), который позволяет выполнять запросы к концептуальным моделям в Entity Framework.

**LINQ to Entities –** это альтернативный интерфейс LINQ API, используемый для обращения к базе данных. Он отделяет сущностную объектную модель данных от физической базы данных, вводя логическое отображение между ними. Так, например, схемы реляционных баз данных не всегда подходят для построения объектно-ориентированных приложений и в результате мы имеем объектную модель приложения, существенно отличающуюся от логической модели данных, в этом случае используется LINQ to Entities, который использует модель EDM (Entity Data Model). То есть, если вам нужно ослабить связь между вашей сущностной объектной моделью данных и физической моделью данных, например, если ваши сущностные объекты конструируются из нескольких таблиц или вам нужна большая гибкость в моделировании ваших сущностных объектов используйте LINQ to Entities

Релиз ADO.NET Entity Framework состоялся 11 августа 2008 года в составе .NET Framework 3.5 Service Pack 1 и Visual Studio 2008 Service Pack 1. В VS 2008 вошёл EDM Wizard для реверс-инжиниринга существующих баз данных и EDM Designer для редактирования сгенерированных моделей или создания их с нуля.

12 апреля 2010 года в составе релиза Visual Studio 2010 и .NET Framework 4.0 был представлена Entity Framework 4.0. Позже уже отдельно от фреймворка были представлены версии: 4.1 (апрель 2011), 4.2 (октябрь 2011), 4.3 (февраль 2012).

11 августа 2012 года была представлена версия 5.0.0, которая была предназначена для .NET Framework 4.5. А 17 октября 2013 года была представлена версия 6.0, которая вышла под лицензией Apache License v2, тем самым став open-source проектом.

Версия 6.0 была выпущена 17 октября 2013 года и сейчас это проект с открытым исходным кодом под лицензией Apache License v2. В версии 6.0 был сделан ряд улучшений в поддержке метода работы Code First.

Изначально с самой первой версии Entity Framework поддерживал подход Database First, который позволял по готовой базе данных сгенерировать модель edmx. Затем эта модель использовалась для подключения к базе данных. Позже был добавлен подход Model First. Он позволял создать вручную с помощью визуального редактора модель edmx, и по ней создать базу данных. Начиная с 5.0 предпочтительным подходом становится Code First. Его суть - сначала пишется код модели на C#, а затем по нему генерируется база данных. При этом модель edmx уже не используется.

# ГЛАВА 2. ПРОЕТИРОВАНИЕ БАЗЫ ДАННЫХ

# 2.1 Разработка диаграммы ERD

Диаграмма — (ER-модель данных) обеспечивает стандартный способ определения данных и отношений между ними. Она включает сущности и взаимосвязи, отражающие основные бизнес-правила предметной области. Диаграммы «сущность— связь» в отличие от функциональных диаграмм определяют спецификации структур данных программного обеспечения.

Базовыми понятиями ER-модели данных (ER — Entity— Relationship) являются сущность, атрибут и связь.

Сущность — это класс однотипных реальных или абстрактных объектов (людей, событий, состояний, предметов и т.п.), информация о которых имеет существенное значение для рассматриваемой предметной области.

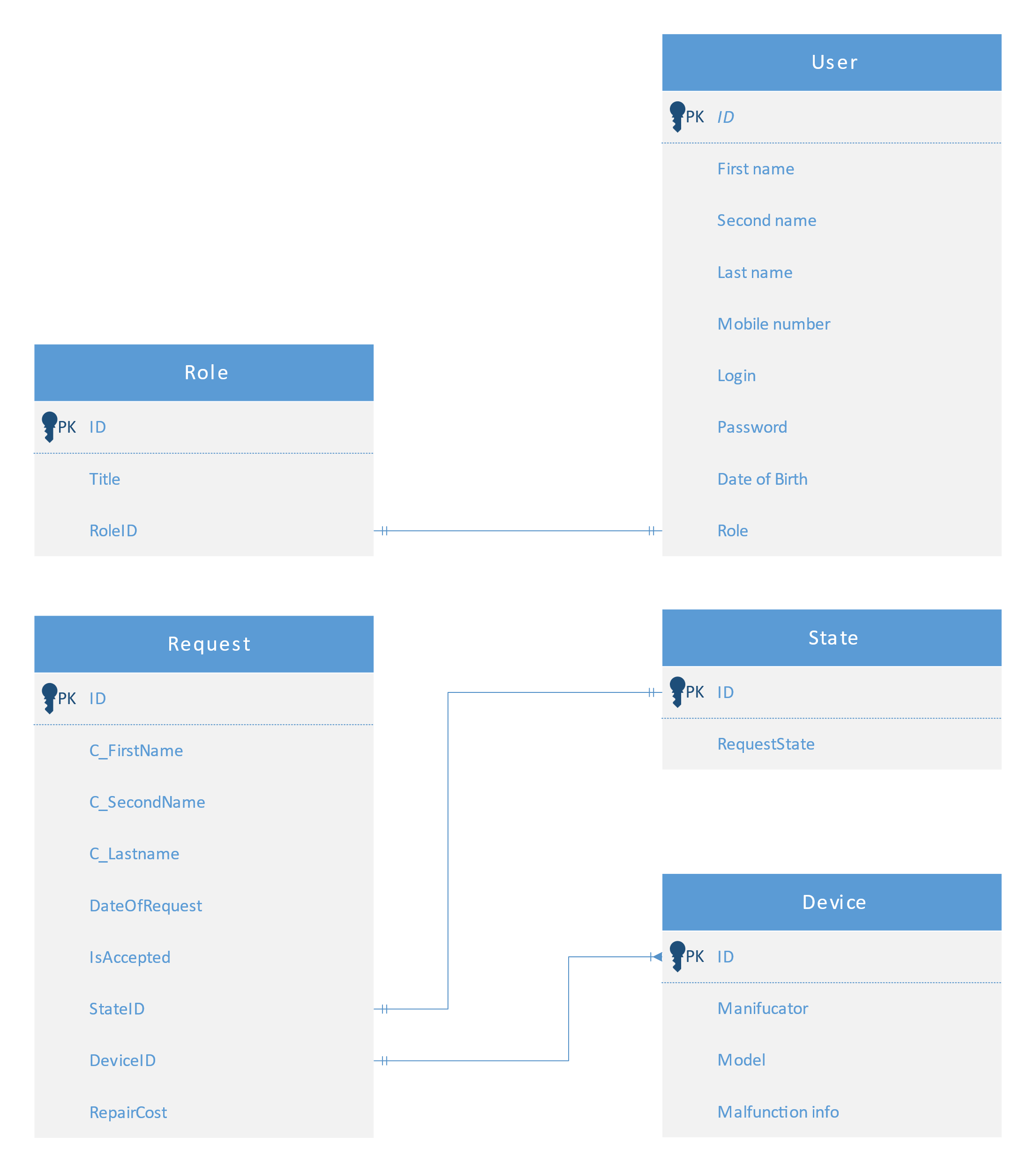
Атрибут — любая характеристика сущности, значимая для рассматриваемой предметной области и предназначенная для квалификации, идентификации, классификации, количественной характеристики или выражения состояния сущности. Атрибут, таким образом, представляет собой некоторый тип характеристик или свойств, ассоциированных с множеством реальных или абстрактных объектов.

Атрибуты делятся на ключевые, т. е. входящие в состав уникального идентификатора ключа, и описательные — прочие.

Первичный ключ — это атрибут или совокупность атрибутов и связей, предназначенная для уникальной идентификации каждого экземпляра сущности.

Описательные атрибуты могут быть обязательными или необязательными. Обязательные атрибуты для каждой сущности всегда имеют конкретное значение, необязательные могут быть не определены.

Связь — это отношение одной сущности к другой или к самой себе. Если любой экземпляр одной сущности связан хотя бы с одним экземпляром другой сущности, то связь является обязательной. Необязательная связь представляет собой условное отношение между сущностями. Каждая сущность может обладать любым количеством связей с другими сущностями модели. Различают три типа отношений «один-к-одному»; «один-ко-многим»; «многие-ко-многим».

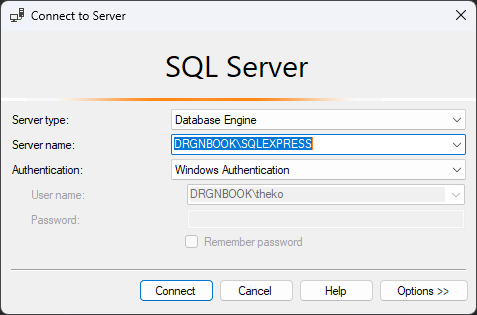


*Рис. 2 - Диаграмма ER*

Основные сущности для данной диаграммы это: Пользователь и Запрос. Второстепенными сущностями являются Роль, Статус и Устройство. Сущность Пользователь обращается к Роль для назначения ID Роли. Запрос содержит в себе данные клиента, а также информацию о статусе заказа и информацию об устройстве в отдельных таблицах. Каждый запрос содержит уникальный идентификатор (личный номер)

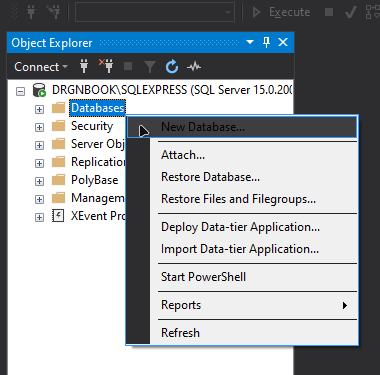
# 2.2 Разработка базы данных

Что бы создать базу данных, нужно запустить “Microsoft SQL Server Management Studio” и выполнить подключение к серверу.



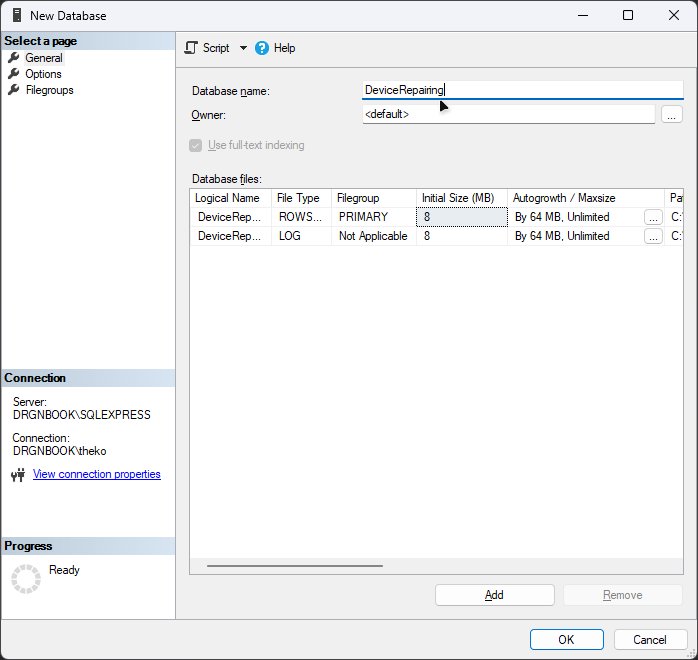
*Рис. 3 - Подключение к серверу*

Далее необходимо нажать правой кнопкой мыши на папку с базами данных и выбрать пункт «Создать базу данных»



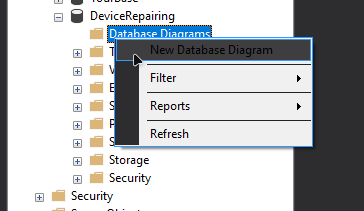
*Рис. 4 - Создание базы данных*

Далее вводим её имя и нажимаем “OK”.



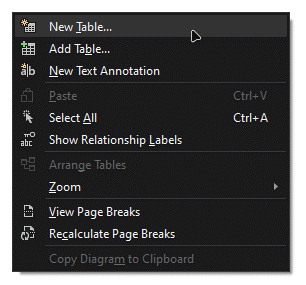
*Рис. 5 - Создание базы данных*

В обозревателе объектов появилась созданная нами база данных, открываем её и нажимаем на “Диаграммы баз данных” и выбираем “Создать диаграмму базы данных”.



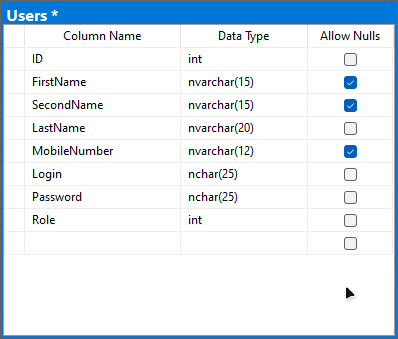
*Рис. 6 - Создание диаграммы*

Для того что бы создавать таблицы нашей базы данных, нужно нажать правой кнопкой мыши по пустому месту нашей диаграммы и выбрать пункт “Создать таблицу” и вводим название нашей таблицы.



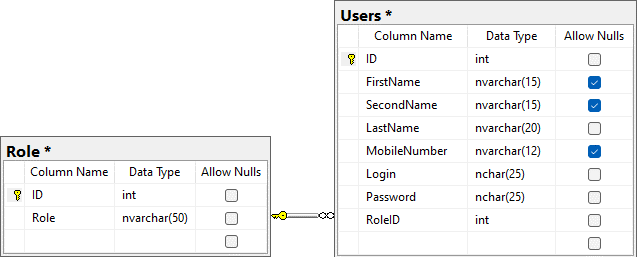
*Рис. 7 - Создание таблиц*

Появилась таблица, в которую можно вписать необходимые атрибуты и уникальные идентификаторы. Вводим атрибуты и присваиваем им типы данных.



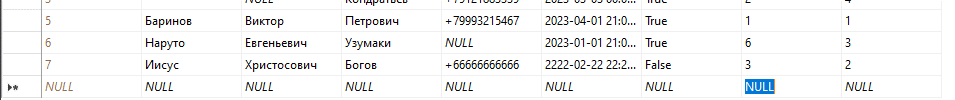
*Рис. 8 - Типы данных и атрибуты*

После создания и заполнения таблиц, соединяем их связями. Для этого зажимаем ключик и ведём к нужному полю в другой таблице.



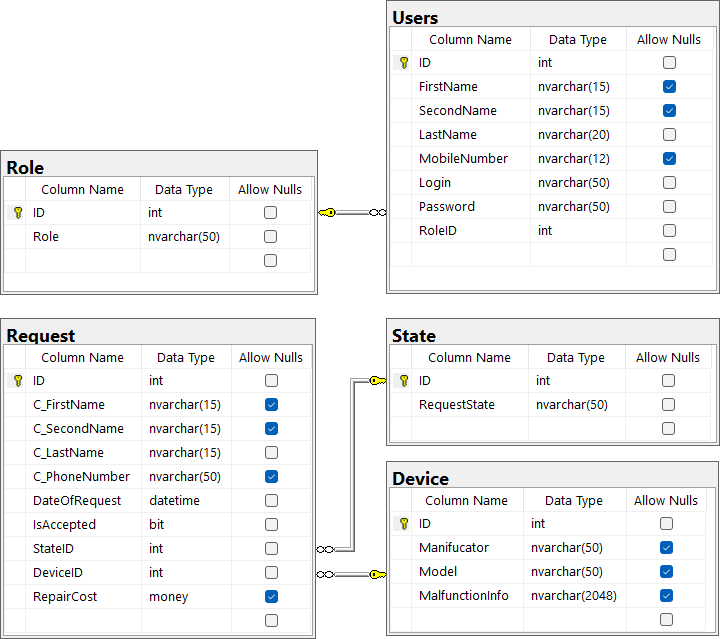
*Рис. 9 - Связь*

Последним шагом будет сохранить диаграмму и выбрать таблицу во вкладке “таблицы”, после чего нажать «Редактировать первые 200 строк».



*Рис. 10 - Заполнение таблиц*

Финальная база данных ИС выглядит следующим образом:



*Рис. 11 - Диаграмма базы данных ИС*

# 2.3 Словарь данных

Словарь данных (Data Dictionary) – сервис, который предоставляет описание данных в бизнес-терминах, и дополнительно может содержать другие сведения о данных, например информацию о типах форматов данных, детализацию структур данных и нормативно-справочной информации, ограничений по безопасности. Таким образом, словарь данных является одним из способов ведения метаданных.

Словари данных связаны с хранилищем данных (Data Warehouses) или другими информационными ресурсами, описывая возможности их использования. Словарь данных помогает организовать ведение метаданных с более высоким качеством по сравнению с неунифицированным документированием, благодаря дисциплинированному и систематическому подходу к управлению определениями и семантикой (смысловым содержанием) информационного пространства организации.

Несмотря на то, что инструменты типа “Словарь данных” имеют различный вид и формат, все они предоставляют компоненты и функции, необходимые для создания соответствующих описаний реляционных баз данных. Международная организация по стандартизации (ISO) выделяет 3 категории для элементов словаря данных:

1. Бизнес-концепты (Business Concepts): бизнес-компоненты с   семантическим (смысловым) значением: связанные с ними бизнес-ассоциации и функциональные роли. К примеру, организация должна определить термин “клиент”, описать характеристики в виде атрибутов и назначить ответственных за их ведение.
2. Типы данных (Data Types): технические характеристики допустимых значений определенного бизнес-элемента. Например, формат данных числовой, текстовый и др.

Пользователи баз данных и разработчики [приложений](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) могут получить выгоду от единого стандартизированного документа словаря данных, который перечисляет организацию, содержимое, соглашения по одной или более баз данных. Это обычно включает в себя имена и описания различных [таблиц](https://ru.wikipedia.org/wiki/%D0%A2%D0%B0%D0%B1%D0%BB%D0%B8%D1%86%D0%B0_(%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)) и [полей](https://ru.wikipedia.org/w/index.php?title=%D0%A1%D1%82%D0%BE%D0%BB%D0%B1%D0%B5%D1%86_(%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)&action=edit&redlink=1) в каждой базе данных, дополнительные детали такие, как [тип](https://ru.wikipedia.org/wiki/%D0%A2%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) и длина каждого [элемента данных](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85). Не существует универсального стандарта, описывающего уровень детализации в подобном документе, но есть основное описание [метаданных](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%B0%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D0%B5) о структуре базы данных, а не о самих данных. Документ словаря данных также может включать в себя дополнительную информацию, описывающую кодирование элементов данных.

Таблица 1 – Словарь данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Dictionary | | | | |
| **Role** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **ID** | int | Y | Уникальный номер роли |
|  | **Role** | nvarchar(50) | Y | Название роли |
| **Users** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **ID** | int | Y | Уникальный номер пользователя |
|  | **FirstName** | nvarchar(15) | N | Имя пользователя |
|  | **SecondName** | nvarchar(15) | N | Отчество пользователя |
|  | **LastName** | nvarchar(20) | Y | Фамилия пользователя |
|  | **MobileNumber** | nvarchar(12) | N | Номер телефона |
|  | **Login** | nvarchar(50) | Y | Логин |
|  | **Password** | nvarchar(50) | Y | Пароль |
| FK | **RoleID** | int | Y | Роль пользователя |
| **State** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **ID** | int | Y | Уникальный номер состояния |
|  | **RequestState** | nvarchar(50) | Y | Состояние заявки |
| **Device** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **ID** | int | Y | Уникальный номер устройства |
|  | **Manifucator** | nvarchar(50) | N | Производитель устройства |
|  | **Model** | nvarchar(50) | N | Модель устройства |
|  | **MalfunctionInfo** | nvarchar(2048) | N | Информация о поломке |
| **Request** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **ID** | int | Y | Уникальный номер заявки |
|  | **C\_FirstName** | nvarchar(15) | N | Имя клиента |
|  | **C\_SecondName** | nvarchar(15) | N | Отчество клиента |
|  | **C\_LastName** | nvarchar(15) | Y | Фамилия клиента |
|  | **C\_PhoneNumber** | nvarchar(50) | N | Номер телефона для связи |
|  | **DateOfRequest** | datetime | Y | Дата открытия заявки |
|  | **IsAccepted** | bit | Y | Статус заявки |
| FK | **StateID** | int | Y | Состояние заявки |
| FK | **DeviceID** | int | Y | Устройство, затрагиваемое в заявке |
|  | **RepairCost** | money | N | Стоимость ремонта устройства |

# ГЛАВА 3. РАЗРАБОТКА БАЗЫ ДАННЫХ

# 3.1 Введение в безопасность SQL Server

Введение в безопасность SQL Server включает в себя ряд важных мер, которые необходимо предпринять для защиты данных и обеспечения безопасности операций. Ниже приведены основные аспекты, которые обычно рассматриваются при обсуждении безопасности SQL Server.

1. Аутентификация и авторизация: SQL Server предоставляет средства аутентификации и авторизации для управления доступом к данным и функциональным возможностям сервера. Это включает управление пользователями, ролями, уровнями доступа и создание безопасных паролей.

2. Шифрование данных: SQL Server поддерживает шифрование данных в хранилище, используя различные методы, такие как прозрачное шифрование столбцов, Transparent Data Encryption (TDE) и шифрование резервных копий.

3. Аудит и мониторинг: SQL Server предоставляет средства для мониторинга и аудита доступа к данным и операциям с сервером. Это включает возможности отслеживания изменений, журналирования событий и анализа безопасности.

4. Управление ролями и полномочиями: В SQL Server можно определять различные роли и полномочия для пользователей и групп, что позволяет более точно управлять доступом к различным частям базы данных.

5. Защита от инъекций: SQL Server обеспечивает механизмы защиты от различных форм атак, таких как SQL-инъекции, предотвращая возможные уязвимости.

6. Обновление и патчи: Регулярное обновление и установка патчей являются важными аспектами обеспечения безопасности SQL Server, поскольку это помогает в устранении обнаруженных уязвимостей.

Введение в безопасность SQL Server состоит из умения применять вышеперечисленные меры, а также из понимания угроз безопасности и способов их предотвращения.

Для создания резервной копии базы данных необходимо воспользоваться следующим SQL-запросом:

****

Для восстановления резервной копии базы данных необходимо использовать следующий запрос:

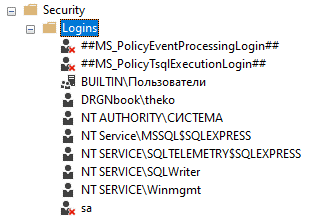
****

# 3.2 Управление безопасностью уровня сервера

Для управления безопасностью уровня сервера необходимо создать аккаунты, которые позволят ограничить пользователей в доступе к базам данных и действиям их редактирования.

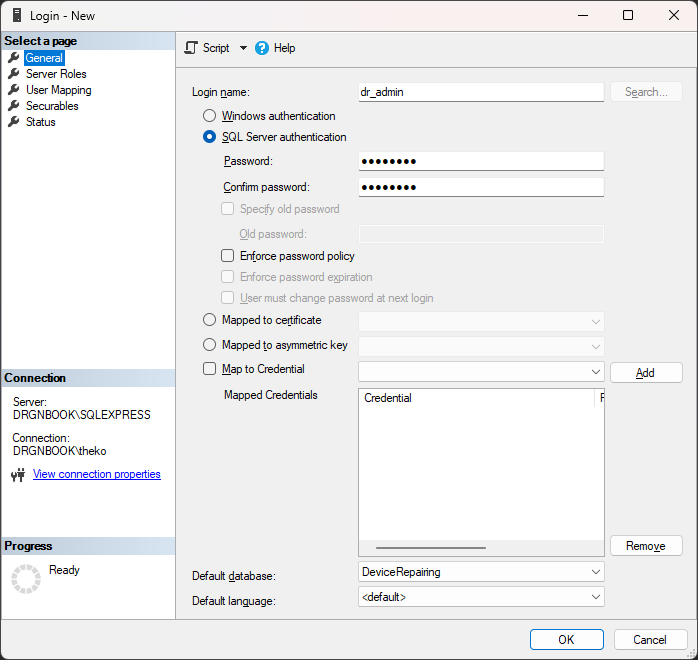
Перед созданием пользователя базы данных необходимо создать логин для входа этого пользователя на сервер, на котором расположена база данных. Для этого необходимо выполнить следующие шаги:

1. Запустите SQL Server Management Studio и подключитесь к серверу баз данных, для которого вы хотите создать логин;
2. в обозревателе объектов раскройте папку "Безопасность" и перейдите к папке "Логины";



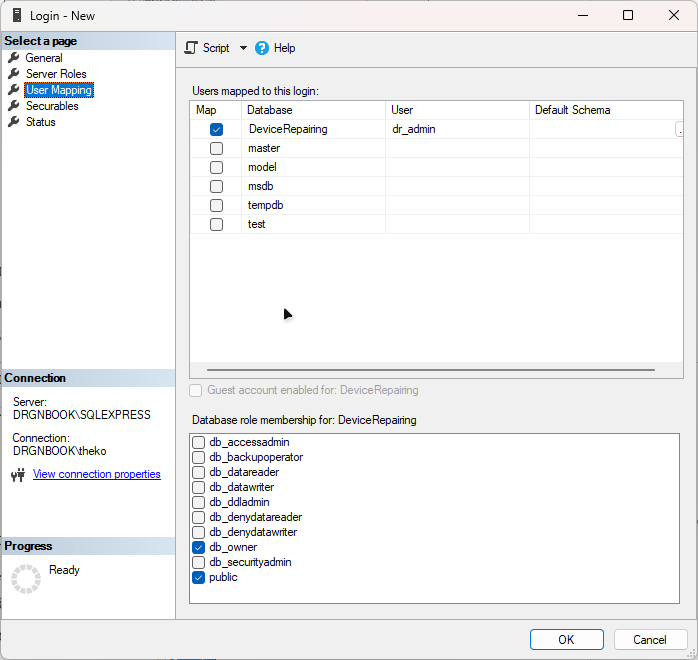
*Рис. 12 - Логины SQL-сервера*

1. Щелкните правой кнопкой мыши по папке "Логины" и выберите "Новый логин": это откроет окно, в котором вы сможете указать имя нового логина, тип аутентификации (Windows или SQL Server), а также другие настройки, такие как разрешения и роль;

**

*Рис. 13 - Создание нового логина*

1. В появившемся окне укажите имя логина пользователя, его тип аутентификации (Windows или SQL Server), а также другие необходимые параметры;
2. Выберите необходимые разрешения и роль для нового пользователя, учитывая требования безопасности и цели использования этого пользователя.
3. Чтобы не пришлось отдельно создавать еще и пользователя для требуемой базы данных, во вкладке «Назначение Пользователей» необходимо выбрать базу данных, для который требуется пользователь. В таком случае он создастся автоматически и будет готов к работе.;



*Рис. 14 - Настройка нового логина*

1. После заполнения всех необходимых данных нажмите кнопку "ОК" для создания нового логина.

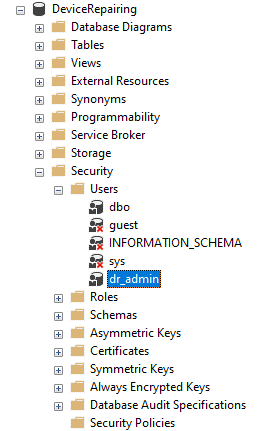
Этот процесс позволяет создавать пользователей глобально в SQL Server Management Studio.

# 3.3 Управление участниками уровня базы данных

Для управления участниками уровня базы данных необходимо создать пользователей или группу пользователей в конкретной базе данных и дать им определённые возможности, в зависимости от их роли в базе данных.

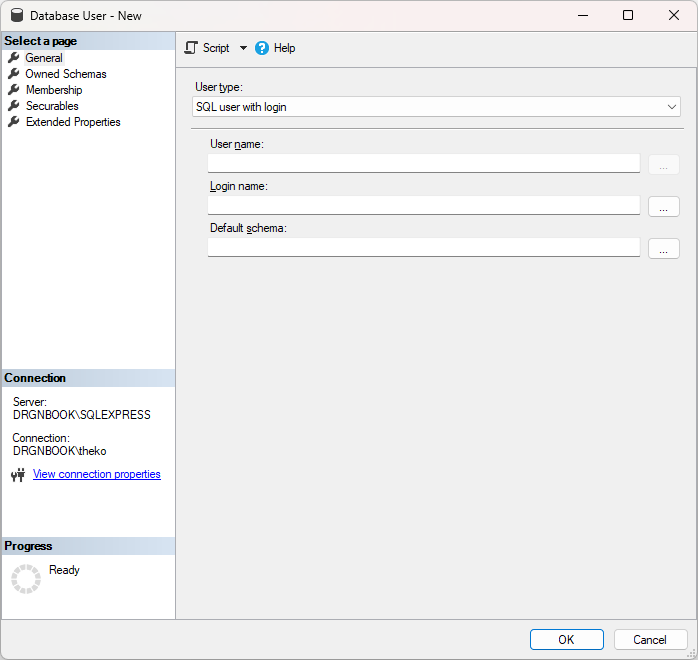
Для создания пользователей локально для базы данных в программе SQL Server Management Studio (SSMS) необходимо выполнить следующие шаги:

1. В обозревателе объектов найдите свою базу данных, щелкните на нее правой кнопкой мыши и выберите "Свойства".



*Рис. 15 - Добавление нового пользователя локально для базы данных*

1. В окне "Свойства базы данных" выберите папку "Безопасность".
2. В папке "Безопасность" выберите папку "Пользователи".
3. Щелкните правой кнопкой мыши по папке "Пользователи" и выберите "Новый пользователь": это откроет всплывающее окно, в котором вы сможете указать имя нового пользователя и его параметры.



*Рис. 16 - Окно создания пользователя*

1. Укажите имя нового пользователя, его тип аутентификации (Windows или SQL Server), а также другие параметры, такие как разрешения и роль.
2. Выберите необходимые разрешения и роль для нового пользователя, учитывая требования безопасности и цели использования этого пользователя.
3. После заполнения всех необходимых данных нажмите кнопку "ОК" для создания нового пользователя в вашей базе данных.

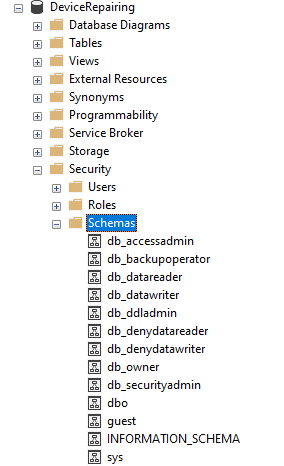
Этот процесс позволяет создавать пользователей локально для конкретной базы данных в SQL Server Management Studio, обеспечивая более тесное управление доступом пользователей к базе данных и ее объектам.

# 3.4 Управление разрешениями уровня базы данных

Управление разрешениями уровня базы данных осуществляется путём выдачи разрешений пользователям на какие – либо действия над таблицами в базе данных.

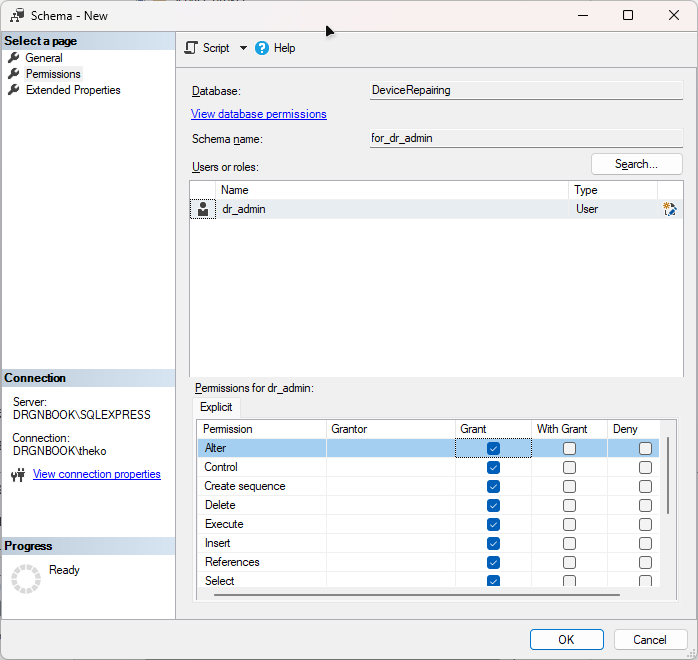
Для создания разрешений для локальных пользователей базы данных в SQL Server Management Studio (SSMS) выполните следующие шаги:

1. В обозревателе объектов найдите вашу базу данных;
2. Выберите папку "Безопасность".
3. В зависимости от того, какие разрешения вы хотите предоставить, выберите либо "Роли базы данных", либо "Схемы", и выберите нужную папку.



*Рис. 17 - Схемы БД DRepairing*

1. Щелкните правой кнопкой мыши по выбранной папке и выберите "Новая роль" или "Новая схема";
2. Укажите имя новой роли или схемы, затем установите подходящие разрешения для этой роли или схемы, такие как SELECT, INSERT, UPDATE, DELETE и т.д.
3. Добавьте локальных пользователей в созданную роль или схему: после создания роли или схемы, добавьте нужных локальных пользователей базы данных в эту роль или схему, чтобы предоставить им соответствующие разрешения.



*Рис. 18 - Настройка новой схемы*

Этот процесс позволяет создавать и управлять разрешениями для локальных пользователей базы данных в SQL Server Management Studio, обеспечивая более гранулированный контроль над доступом к данным и объектам базы данных.

# 3.5 Шифрование данных баз данных

Шифрование данных в базе данных используется для защиты данных, которые могут украсть.

Для создания шифрования данных в базе данных с помощью SQL Server Management Studio (SSMS) следует выполнить следующие шаги:

1. Создание мастера ключа: В объектном обозревателе базы данных перейдите в "Безопасность" -> "Ключи" -> "Мастер ключ" и создайте новый мастер ключ.

2. Создание сертификата: В папке "Безопасность" выберите "Сертификаты" и создайте новый сертификат для шифрования данных.

3. Создание ключа для шифрования данных: после создания сертификата откройте папку "Ключи" и создайте новый ключ для шифрования данных.

4. Шифрование колонки или данных: после создания ключа и сертификата выберите конкретную колонку или данные, которые требуется зашифровать, и примените шифрование, используя созданный ключ.

5. Настройка доступа к ключам и сертификатам: убедитесь, что только авторизованные пользователи имеют доступ к созданным ключам и сертификатам, чтобы обеспечить безопасное хранение и расшифровку зашифрованных данных.

Эти шаги позволяют создать и применить шифрование данных в базе данных с помощью SQL Server Management Studio, обеспечивая защиту конфиденциальных данных от несанкционированного доступа.

# ЗАКЛЮЧЕНИЕ

При выполнении данной работы, а именно конструирования автоматизированной системы «Ремонт мобильных устройств», была разработана диаграмма Entity Relationship, была спроектирована база данных в приложении Microsoft SQL Server Management Studio 18. Данная автоматизированная информационная система была разработана для всех сотрудников системы. С её помощью можно быстро регистрировать новых сотрудников, добавлять заявки и формировать отчёты.

Экономический эффект от внедрения автоматизированной информационной системы «Ремонт мобильных устройств» ожидается за счет сокращения затрат времени на выполняемые сотрудниками задачи, исключения ошибок при формировании отчетов, увеличения времени на анализ и повышения продуктивности сотрудников. При ручном создании отчетов могут быть допущены ошибки, правильно составленный алгоритм разрабатываемой программы экономит время и исключает ошибки.

Поставленные цели и задачи при проектировании информационной системы были выполнены в полной мере.

Ссылка на репозиторий GitHub: <https://github.com/thekottv/DRepairing>

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Грин Дженнифер - Head First: изучаем С#/ Грин Дженнифер, Стиллмен Эндрю – Санкт-Петербург, 2022.
2. Документация по C#. Начало работы, руководства, справочные материалы. // Microsoft Learn. – Режим доступа:  <https://learn.microsoft.com/ru-ru/dotnet/csharp/>
3. Особые исключения в .NET и как их готовить // Хабр – Режим доступа: [https://habr.com/ru/companies/jugru/articles/426045](https://habr.com/ru/companies/jugru/articles/426045/)
4. Помощь и объяснение с примерами работы с базами данных с помощью Microsoft SQL Server. – Режим доступа: [https://en.wikipedia.org/wiki/SQLServer](https://en.wikipedia.org/wiki/Microsoft_SQL_Server%20)
5. Уильям Р. Станек. "SQL Server 2019. Базовый курс": книга по программированию/ Уильям Р. Станек. - Питер, 2020. – С. 230-340.
6. Newest 'c#' Questions // StackOverflow. – Режим доступа: [https://stackoverflow.com/questions](https://stackoverflow.com/questions/tagged/c%23)
7. StackOverflow exception. Интерфейсы // C# - Киберфорум – Режим доступа: [https://www.cyberforum.ru/csharp](https://www.cyberforum.ru/csharp-beginners/thread2225109.html?ysclid=lgw4535h24160143326)
8. SQL Server Management Studio (SSMS) // работа с базами данных от “Майкрософт”. – Режим доступа: [https://learn.microsoft.com/en-us/sql/ssms](https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16%20)
9. Using the DataContext - <https://wpf-tutorial.com/data-binding/using-the-datacontext/>
10. W3schools // Подробные учебные материалы по C# и SQL. – Режим доступа: <https://www.w3schools.com/>