**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 базы данных**

**«Администратор гостиницы»**

Выполнил студент гр. ИСП-20 /\_\_\_\_\_\_\_\_\_\_\_\_\_/Родин Олег Альбертович/

(подпись) (Ф.И.О.)

**ОЦЕНКА:** \_\_\_\_\_\_\_\_\_\_\_\_\_

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

**ПРОВЕРИЛ**

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

(подпись) (Ф.И.О.)

Воркута

2023

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc153006255)

[ГЛАВА 1. ВЫБОР ИНСТРУМЕНТАРИЯ 5](#_Toc153006256)

[1.1СУБД SQL Server 5](#_Toc153006257)

[1.2 Microsoft SQL Server Management Studio 6](#_Toc153006258)

[1.3 Entity Framework 7](#_Toc153006259)

[ГЛАВА 2. ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ 9](#_Toc153006260)

[2.1 Разработка диаграммы ERD 9](#_Toc153006261)

[2.2 Разработка базы данных 11](#_Toc153006262)

[2.3 Словарь данных 16](#_Toc153006263)

[ГЛАВА 3. РАЗРАБОТКА БАЗЫ ДАННЫХ 20](#_Toc153006264)

[3.1 Введение в безопасность sql server 20](#_Toc153006265)

[3.2 Управление безопасностью уровня сервера 21](#_Toc153006266)

[3.3 Управление участниками уровня базы данных 22](#_Toc153006267)

[3.4 Управление разрешениями уровня базы данных 23](#_Toc153006268)

[3.5 Шифрование данных баз данных 23](#_Toc153006269)

[ЗАКЛЮЧЕНИЕ 25](#_Toc153006270)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 26](#_Toc153006271)

ВВЕДЕНИЕ

Сегодня гостиничный бизнес - одна из наиболее перспективных и быстроразвивающихся отраслей, приносящих по всему миру многомиллионные прибыли.

С каждым годом растет количество гостиниц, как в нашей стране, так и за рубежом, поэтому тема разработки информационной системы «Администратор гостиницы» в настоящее время является актуальной, так как гостиничный бизнес активно развивается, разрастаются сети отелей, создавая на рынке конкуренцию.

Информационная система «Администратор гостиницы» даёт преимущество перед конкурентами, сокращает время на выполнение поставленных задач. Да и сейчас трудно обойтись без информационной систем, тем более в современном мире, с его сумасшедшим ритмом.

Объект: база данных «Администратор гостиницы».

Предмет: автоматизация процесса работы гостиницы.

Цель курсовой работы - разработать информационную систему для автоматизации бизнес-процесса заселения гостей в номера.

Повысить эффективность за счет увеличения количества и повышения качества информации и работы сотрудников.

Для достижения цели продукта нужно выполнить следующие задачи:

1. составить техническое задание к информационной системе;
2. спроектировать базу данных информационной системы;
3. разработать информационную систему;
4. оптимизировать временные затраты сотрудников путём автоматизации.

ГЛАВА 1. ВЫБОР ИНСТРУМЕНТАРИЯ

* 1. СУБД SQL Server

Система управления реляционными базами данных (РСУБД), разработанная корпорацией Microsoft. Основной используемый язык запросов - Transact-SQL, создан совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов 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.

Чтобы упростить работу с такими хранилищами данных и повысить эффективность их применения, создаются специализированные системы управления. Одной из наиболее популярных является разработка от Microsoft – SQL Server. Первый релиз платформы опубликован еще в 1989 году, а последняя версия выпущена в 2019 году (проект продолжает развиваться).

Преимущества решения:

* тесная интеграция с операционной системой Windows;
* высокая производительность, отказоустойчивость;
* поддержка многопользовательской среды;
* расширенные функции резервирования данных;
* работа с удаленным подключением.

Каждый выпуск включает в себя несколько специализированных редакций. Это снижает сложность внедрения и затраты на процесс разработки собственных решений, адаптированных для «узких» задач. При написании программного кода активно используется интеграция с продуктами Microsoft, например, с платформой Visual Studio.

Появление таких продуктов позволило объединить разное понимание БД (баз данных) со стороны пользователей и системных администраторов. Неискушенные в технических деталях люди «видят» таблицы как некий перечень данных с колонками и строками. Системный подход включает файлы с табличными данными, связанными друг с другом.

Функции базы данных:

* постоянное хранение информации;
* поиск по ключевым критериям;
* чтение и редактирование по запросу.

Клиентами БД являются прикладные программы, их интерфейс, различные интерактивные модули сайтов вроде калькуляторов и онлайн-редакторов. Но есть еще один компонент системы – СУБД. Он предназначен для ручного доступа к информации и позволяет извлекать данные на диск, работать с ними в памяти сервера, в том числе с применением структурированного языка SQL.

Всего различают три типа БД – клиент-серверные, файл-серверные и встраиваемые. MS SQL Server относится к первой категории. Плюс система является реляционной, т.е. адаптированной для хранения данных без избыточности, с минимальными рисками появления аномалий и нарушения целостности внутренних таблиц.

1.2 Microsoft SQL Server Management Studio

SQL Server Management Studio (SSMS) — это интегрированная среда для управления любой инфраструктурой SQL. Используйте SSMS для доступа, настройки, администрирования и разработки всех компонентов SQL Server.

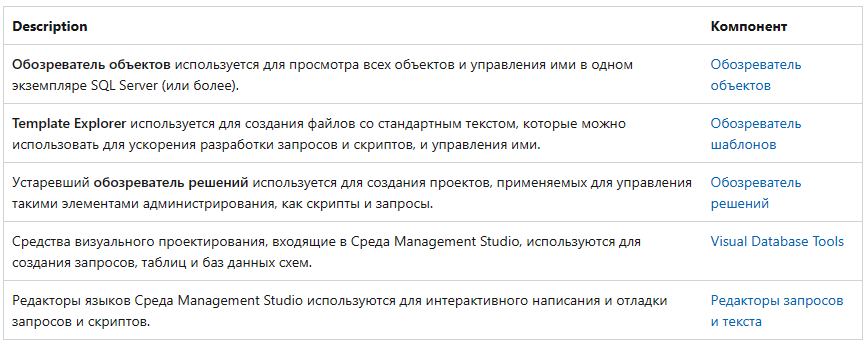
Бесплатная графическая среда, включающая набор инструментов для разработки сценариев на T-SQL и управления инфраструктурой Microsoft SQL Server.

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.



*Рисунок 1.1. Описание компонентов*

1.3 Entity Framework

Entity Framework — это решение для работы с базами данных, которое используется в программировании на языках семейства .NET. Оно позволяет взаимодействовать с СУБД с помощью сущностей (entity), а не таблиц. Также код с использованием EF пишется гораздо быстрее.

Например, работая с базами данных напрямую, разработчик должен беспокоиться о подключении, подготовке SQL и параметров, отправке запросов и транзакций. На Entity Framework всё это делается автоматически — программист же работает непосредственно с сущностями и только говорит EF, что нужно сохранить изменения.

Набор технологий в ADO.NET, которые поддерживают разработку программных приложений, ориентированных на данные.

Архитекторам и разработчикам приложений, ориентированных на обработку данных, приходится учитывать необходимость достижения двух совершенно различных целей. Они должны моделировать сущности, связи и логику решаемых бизнес-задач, а также работать с ядрами СУБД, используемыми для сохранения и получения данных. Данные могут распределяться по нескольким системам хранения данных, в каждой из которых применяются свои протоколы, но даже в приложениях, работающих с одной системой хранения данных, необходимо поддерживать баланс между требованиями системы хранения данных и требованиями написания эффективного и удобного для обслуживания кода приложения.

Платформа Entity Framework позволяет работать с данными в форме специфических для домена объектов и свойств (например, с клиентами и их адресами) без необходимости учитывать формат базовых таблиц и столбцов базы данных, где хранятся эти данные.

Entity Framework дает разработчикам возможность работать с данными на более высоком уровне абстракции, создавать и сопровождать приложения, ориентированные на работу с данными, одновременно с этим сокращая объем кода по сравнению с традиционными приложениями. поскольку

Entity Framework является компонентом платформа .NET Framework, Entity Framework приложения могут работать на любом компьютере, на котором установлена платформа .NET Framework с пакетом обновления 1 (SP1) версии 3,5.

ГЛАВА 2. ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ

2.1 Разработка диаграммы ERD

Характеристика диаграмм «сущность— связь». Данная 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— Relati-onship) 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. Структурой данных называют совокупность правил и ограничений, которые отражают связи, существующие между отдельными частями (элементами) данных.

Каждая сущность должна иметь:

* уникальное имя;
* один или несколько атрибутов, которые либо принадлежат сущности, либо наследуются через связь;
* один или несколько атрибутов, которые однозначно идентифицируют каждый экземпляр сущности.

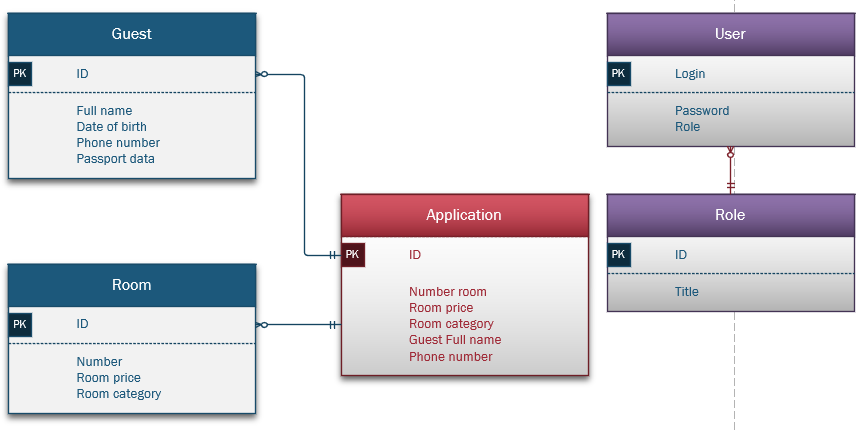
Атрибут — любая характеристика сущности, значимая для рассматриваемой предметной области и предназначенная для квалификации, идентификации, классификации, количественной характеристики или выражения состояния сущности. Атрибут, таким образом, представляет собой некоторый тип характеристик или свойств,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

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

Первичный ключ — это атрибут или совокупность атрибутов и (или) связей, предназначенная для уникальной идентификации каждого экземпляра сущности (совокупность признаков, позволяющих идентифицировать объект). Ключевые атрибуты помещают в начало списка и помечают символом «#».

Описательные атрибуты могут быть обязательными или необязательными.

Обязательные атрибуты для каждой сущности всегда имеют конкретное значение, необязательные могут быть не определены. Обязательные и необязательные описательные атрибуты помечают символами «\*» и «°» соответственно.

Связь — это отношение одной сущности к другой или к самой себе. Каждая связь может иметь одну из двух модальностей связей. 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.

*Рисунок 2.1. Диаграмма ERD*

Связь может иметь разную модальность с разных концов. Каждая сущность может обладать любым количеством связей с другими сущностями модели. Различают три типа отношений «один-к-одному»; «один-ко-многим»; «многие-ко-многим».

Основными сущностями для решения указанной задачи являются: Клиент (Guest - гость), пользователь (User), Роль (Role), номер (Room), заявка (Application). Сразу возникает очевидная связь между сущностями — «Гостем, который может оформлять много заявок», и «связь между заявкой и номером». Заявка содержит ID (уникальный ключ) гостя, а также номер комнаты, его цену, категорию, телефон гостя и его полное имя. Отношение между ними относится к типу «многие-к-одному». Также есть связь между двумя сущностями, пользователем и его ролью, пользователей много, но роль у них всего одна.

Проанализируем атрибуты сущности.

Гость имеет ФИО, телефон, паспортные данные, у каждого пользователя есть ID отличающий его от других клиентов. Каждая заявка имеет номер, каждая комната (номер) имеет свой номер. Администраторы могут получают данные из системы и выводить их в виде отчёта.

2.2 Разработка базы данных

SQL Server Management Studio — это набор административных средств для управления компонентами, относящимися к SQL Server. Эта интегрированная среда позволяет пользователям выполнять разнообразные задачи, например резервное копирование данных, редактирование запросов и автоматизацию общих функций в одном интерфейсе.

Среда SQL Server Management Studio включает в себя следующие средства:

Редактор кода — богатый возможностями редактор скриптов для написания и редактирования скриптов. Среда SQL Server Management Studio предоставляет четыре версии редактора кода: редактор запросов ядра СУБД для скриптов Transact-SQL, редактор запросов многомерных выражений, редактор запросов расширения интеллектуального анализа данных и редактор запросов XML/A.

Обозреватель объектов для размещения, изменения, создания скрипта или выполнения объектов, принадлежащих экземплярам SQL Server.

Обозреватель шаблонов для размещения и написания сценариев шаблонов.

Обозреватель решений для организации и хранения связанных скриптов как части проекта.

Окно свойств для отображения текущих свойств выбранных объектов.

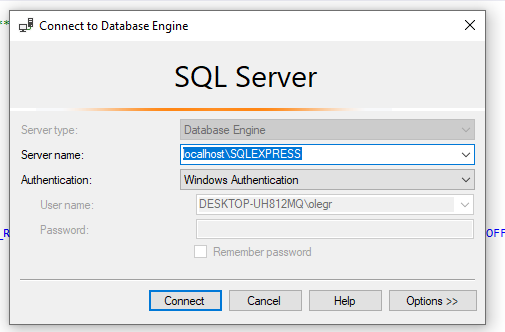
Среда SQL Server Management Studio обеспечивает эффективность рабочих процессов, предоставляя:

Отключенный доступ. Можно писать и изменять скрипты, не соединяясь с экземпляром SQL Server.

Создание сценариев из любого диалогового окна. Можно создать скрипт из любого диалогового окна, а также читать, изменять, сохранять и многократно использовать скрипты после создания.

Немодальные диалоговые окна. При обращении к диалоговому окну интерфейса можно просмотреть другие ресурсы в среде SQL Server Management Studio, не закрывая диалоговое окно.

Разработка базы данных начинается с ранее разработанной ERD диаграммой. Откроем Microsoft SQL Server Management Studio и подключаемся к нашему локальному серверу.



*Рисунок 2.2. Окно подключения к локальному серверу*

После чего у нас станет доступна таблица Object Explorer в которой будут показаны файлы и базы данных на локальном сервере. Создадим базу данных, нажимаем ПКМ по папке «Databases» и выбираем «New Database».

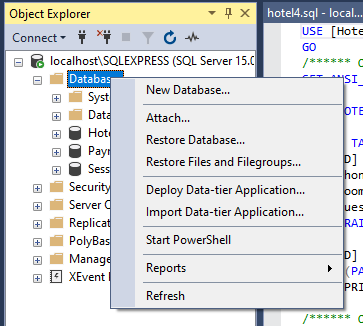


Рисунок 2.3. Создание базы данных

Далее нам выводиться окно, в котором мы даём название для базы данных и

н нажимаем на кнопку OK. В Object Explorer появиться созданная база данных, развернём её, нажмём ПКМ по папке Database Diagrams и нажмём на New Database Diagram.

После чего нам выведется поле для создания диаграммы базы данных, на котором мы можем создавать таблицы с название столбцов и их типом данных.

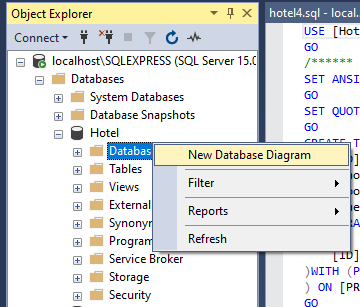


Рисунок 2.4. Обозреватель базы данных

Для создания таблицы данных необходимо нажать ПКМ в любом месте диаграммы,

выбираем «New Table» и даём название для нашей таблицы. Для информационной системы необходимо создать пять таблиц: Клиент (Guest - гость), пользователь (User), Роль (Role), номер (Room), заявка (Application). После чего заполним таблицы необходимыми полями и типами данных из ERD таблицы.

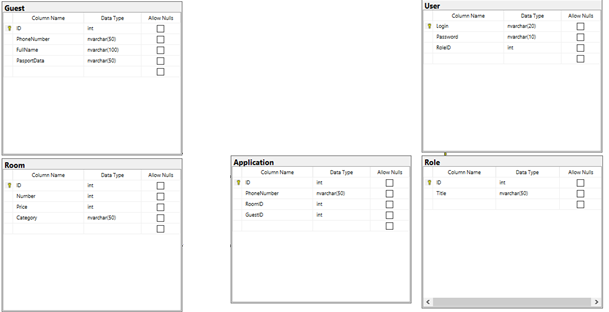


Рисунок 2.5. Все таблицы

Далее создаём связи как в ERD сущностях и после сохраняем таблицы нажав сочетание клавиш «Ctrl + S» и в высветившимся окне пишем название нашей диаграммы.

Созданные нами таблицы можно увидеть в Object Explorer кликнув на название базы данных и перейдя в папку Tables. Зажав уникальный ключ, тянем к полу в нужной таблице и создаётся связь

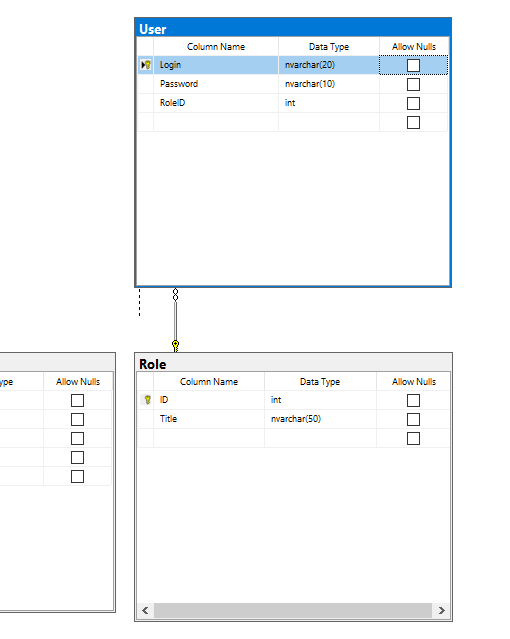


Рисунок 2.6. Создание связи

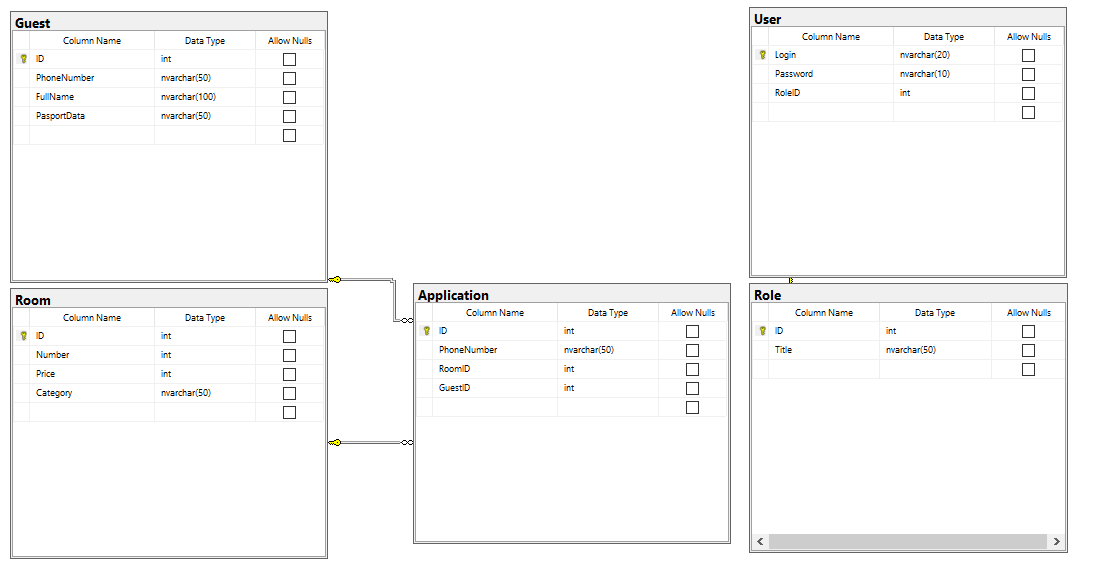
После создания связей наша таблица полностью готова.

Рисунок 2.7. База данных

В данной работе используется 5 таблицы базы данных. Gust содержит уникальный ключ, номер телефона, ФИО, паспортные данные, Room также как и гость и другие таблицы имеют уникальный ключ, у комнаты есть номер, цена, категория. User имеет логин пароль и роль, и таблица напрямую связанна с таблицей Role откуда он берёт роль каждого пользователя. Таблица Application (заявка) имеет связь с другими таблица и берёт из них данный для формирования данных в себе.

2.3 Словарь данных

Словарь данных содержит информацию об источниках, взаимосвязях между данными, их описания, сведения о характере использования. Словарь данных можно рассматривать как вспомогательную базу данных, в которой хранится информация об основной базе данных.

Пользователи баз данных и разработчики приложений 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 и полей в каждой базе данных, дополнительные детали такие, как тип и длина каждого элемента данных. Не существует универсального стандарта, описывающего уровень детализации в подобном документе, но есть основное описание метаданных о структуре базы данных, а не о самих данных.

Одним из преимуществ хорошо спроектированного словаря данных является то, что он помогает упорядочить структуру базы данных или большого комплекса баз данных.

Таблица 1. Пользователь

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Dictionary** | | | | |
| **User** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | Login | NVARCHAR (20) | Y | Запрашивает имя |
|  | Password | NVARCHAR (10) | Y | Запрашивает пароль |
| FK | RoleID | int | Y | Указывает роль пользователя |

Таблица 2. Роль

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Dictionary** | | | | |
| **Role** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | ID | int | Y | Уникальный номер по списку |
|  | Title | NVARCHAR (50) | Y | Описание уникального номера |

Таблица 3. Заявка

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Dictionary** | | | | |
| **Application** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | ID | int | Y | Уникальный номер по списку |
|  | PhoneNumber | NVARCHAR (50) | Y | Номер телефона в заявке |
|  | RoomID | int | Y | Уникальный номер комнаты по списку в заявке |
|  | GuestID | int | Y | Уникальный номер гостя по списку в заявке |

Таблица 4. Гость

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Dictionary** | | | | |
| **Guest** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | ID | int | Y | Уникальный номер по списку |
|  | PhoneNumber | NVARCHAR (50) | Y | Номер телефона гостя |
|  | FullName | NVARCHAR (100) | Y | Полное ФИО гостя |
|  | PasportDate | NVARCHAR (50) | Y | Паспортные данные гостя |

Таблица 5. Номера

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Dictionary** | | | | |
| **Room** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | ID | int | Y | Уникальный номер по списку |
|  | Number | int | Y | Номер комнаты в гостинице |
|  | Price | int | Y | Цена номера в гостинице |
|  | Category | NVARCHAR (50) | Y | Категория номера в гостинице |

ГЛАВА 3. РАЗРАБОТКА БАЗЫ ДАННЫХ

3.1 Введение в безопасность sql server

Многоуровневая технология безопасности предусматривает решение "защита в глубину", используя многочисленные возможности защиты, направленные на различные области безопасности. Возможности безопасности, доступные в SQL Server 2018, и улучшенные в последующих выпусках, помогают противостоять угрозам безопасности и обеспечивать надежную защиту приложений баз данных. Для обеспечения безопасности используем функцию “Backup” для создания файла восстановления.

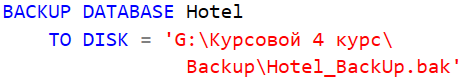


Рисунок 3.1. Код для создания файла восстановления

Создали файл резервного восстановления базы данных.

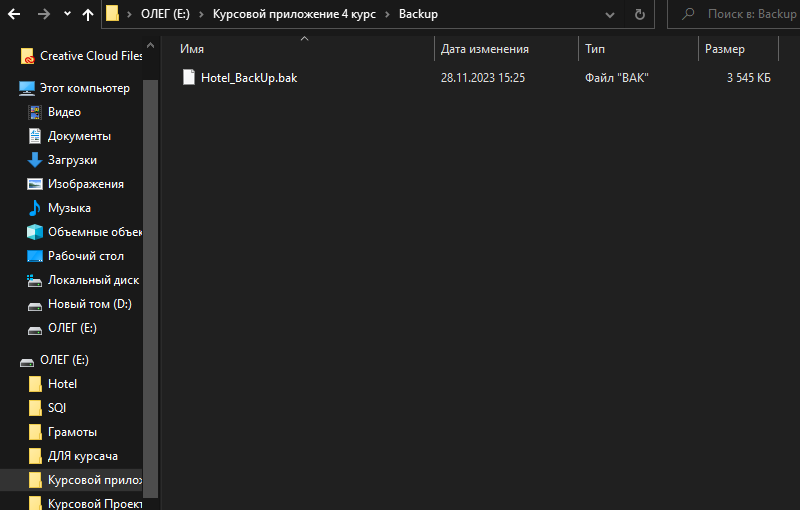


Рисунок 3.2. Файл восстановления

После создания файла его можно будет восстановить функцией “restore”.

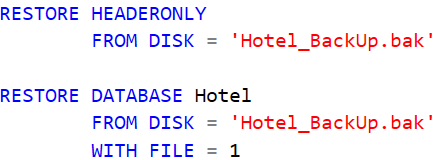


Рисунок 3.3. Код для восстановления базы данных

3.2 Управление безопасностью уровня сервера

Создание пользователя, присвоение ему пароля.

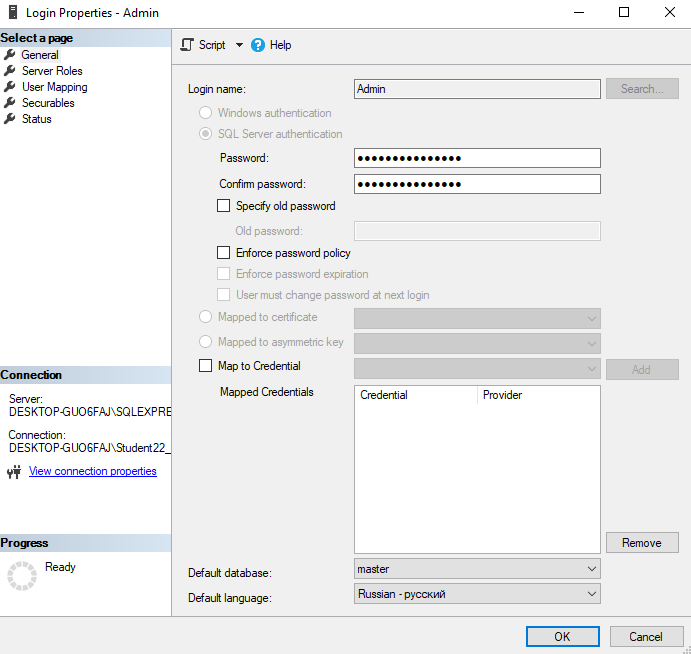


Рисунок 3.4. Добавление пользователя

Подключение пользователя к базе данных Hotel.

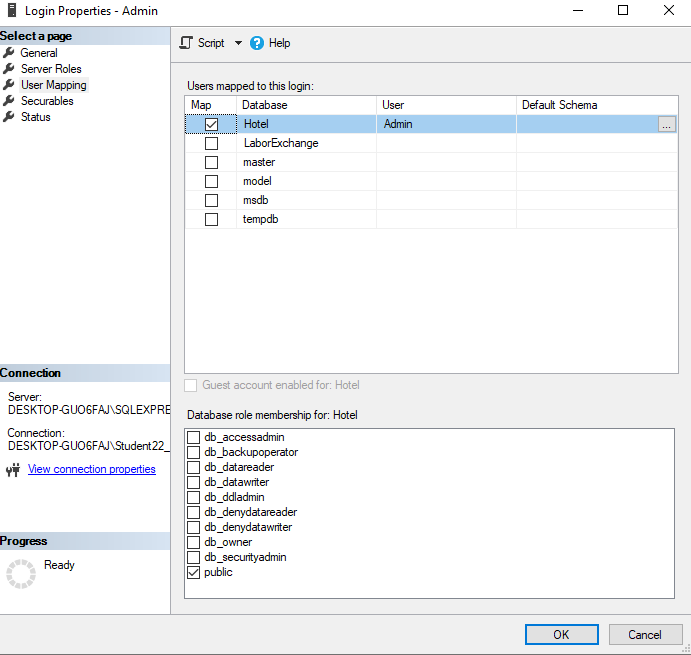


Рисунок 3.5. Разграничение прав доступа к базе данных

Создали двух новый пользователей в системе.

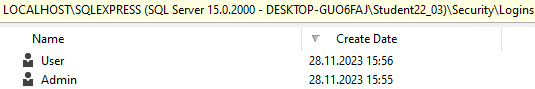


Рисунок 3.6 Созданные пользователи на сервере

3.3 Управление участниками уровня базы данных

Создали два пользователя на уровне сервера, для дальнейшего подключение их к базе данных.

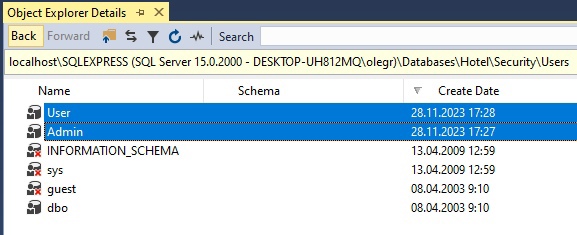


Рисунок 3.7 Созданные пользователи в базе данных Hotel

3.4 Управление разрешениями уровня базы данных

SQL Server использует системную базу данных для поддержания внутренних метаданных. Администратор базы данных сервера должен данных уметь управлять ими. В базе данных выдали права доступа к таблицам и права пользования функциями в данных таблицах.

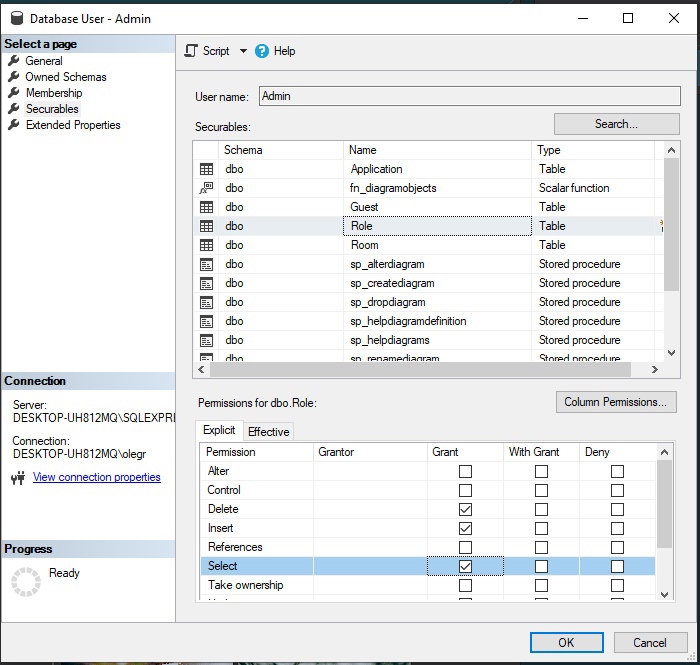


Рисунок 3.8 Выдача прав пользователю

3.5 Шифрование данных баз данных

Шифрование базы данных — использование технологии шифрования для преобразования информации, хранящейся в базе данных (БД), в шифротекст, что делает её прочтение невозможным для лиц, не обладающих ключами шифрования.

Данные шифруются перед записью на диск и дешифруются во время чтения в память, что решает проблему защиты «неактивных» данных, но не обеспечивает сохранность информации при передаче по каналам связи или во время использования.

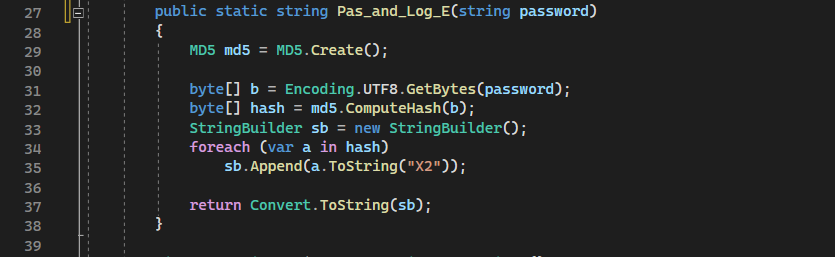


Рисунок 3.9 Шифрование поля логин и пароль в приложении

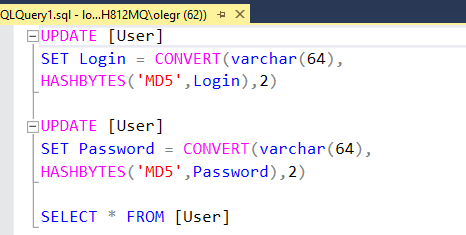


Рисунок 3.10 Шифрование поля логин и пароль в базе данных

ЗАКЛЮЧЕНИЕ

Информационные системы являются актуальными на сегодняшний день - одна из наиболее перспективных и быстроразвивающихся отраслей, приносящих по всему миру многомиллионные прибыли. С каждым годом растет количество информационных систем, как в нашей стране, так и за рубежом.

При выполнении данной работы, а именно конструировании автоматизированной системы “Администратор гостиницы”, были спроектированы диаграммы: Entity-Relationship Diagram, и разработана база данных, а также разработан интерфейс приложения и произведено программирование ИС. Данная автоматизированная информационная система была разработана для работников гостиницы. С её помощью можно быстро оформлять клиентов гостиницы, выполнять отчёты по нужным данным: ФИО, номер телефона, паспортные данные, номер проживания и категорию номера.

Экономический эффект от внедрения автоматизированной информационной системы «Администратор гостиницы» ожидается за счет сокращения времени на выполняемые работы, исключения ошибок при формировании отчетов, увеличения времени на анализ. При ручном создании отчетов человеком могут быть допущены ошибки; правильно составленный алгоритм разрабатываемой программы ошибки исключает.

При выполнении работы все поставленные задачи были выполнены.

На данной стадии разработки было положено начало и учтены все нюансы при разработке, чтобы в дальнейшем доработать эти недочёты.

Практическая ценность курсового проекта заключается в том, что ИС позволяют получить наиболее достоверную информацию, а также повысить качество выполняем работы сотрудниками.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. .NET Framework – Режим доступа: https://ru.wikipedia.org/wiki/.NET\_Framework

2.C Sharp – Режим доступа: https://ru.wikipedia.org/wiki/C\_Sharp.

3.Entity Framework: как быстрее написать код для работы с базой данных – Режим доступа: https://skillbox.ru/media/code/entity\_framework/.

4.Microsoft SQL Server – Режим доступа: https://ru.wikipedia.org/wiki/Microsoft\_SQL\_Server.

5.SQL Server Management Studio – Режим доступа: https://ru.wikipedia.org/wiki/SQL\_Server\_Management\_Studio.

6.Документация по C# – Режим доступа: <https://learn.microsoft.com/ru->ru/dotnet/csharp/.

7.Обзор Entity Framework – Режим доступа: <https://learn.microsoft.com/ru->ru/dotnet/framework/data/adonet/ef/overview.

8.Обзор платформы .NET – Режим доступа: <https://learn.microsoft.com/ru->ru/dotnet/framework/get-started/overview.

9.Обзор функционала SQL Server Management Studio (SSMS) – Режим доступа: https://info-comp.ru/ssms-feature-overview?ysclid=lenvbogeq8366784796.

10.Общие сведения о WPF – Режим доступа: https://learn.microsoft.com/ru-ru/dotnet/desktop/wpf/introduction-to-wpf?view=netframeworkdesktop-4.8.

11.Руководство по WPF – Режим доступа: https://metanit.com/sharp/wpf/?ysclid=lenvi5n1bh823160038.

12.Руководство по классическим приложениям (WPF .NET) – Режим доступа: https://learn.microsoft.com/ru-ru/dotnet/desktop/wpf/overview/?view=netdesktop-6.0.

13.Что такое MS SQL Server и чем она отличается от других СУБД – Режим доступа: https://timeweb.com/ru/community/articles/osnovy-raboty-s-mssqlserver.

14.Язык С#: где используют, что пишут, как появился и чем хорош – Режим доступа: https://skillbox.ru/media/code/yazyk-s-gde-ispolzuyut-chto-pishut-kak-poyavilsya-i-chem-khorosh/?ysclid=lenvmtelk1292224169.