МИНИСТЕРСТВО ОБРАЗОВАНИЯ КИРОВСКОЙ ОБЛАСТИ

Кировское областное государственное профессиональное образовательное

бюджетное учреждение

«Слободской колледж педагогики и социальных отношений»

**ОТЧЕТ**

**по учебной практике**

**ПМ 02. Разработка и администрирование баз данных**

**Тема: «Разработка базы данных «Рыбные хозяйства»**

Студента

Ермакова Павла Андреевича

Группа 21П-1

Специальность 09.02.07Информационные системы и программирование

Руководитель практики от колледжа

Калинин А.О/Махнев Александр Анатольевич

Подпись расшифровка

2024 год

**Содержание**

1. Выполнение анализа и предварительной обработки информации
2. Выделение объектов и атрибутов в соответствии с заданием
3. Проектирование и нормализация БД в полном соответствии с поставленной задачей и применением CASE-средств
4. Выполнение построения БД в предложенной СУБД и заполнение всех таблиц с помощью соответствующих средств
5. Реализация уровней доступа для различных категорий пользователей
6. Создание запросов и отчетов в соответствии с заданием
7. Создание и обоснование групп пользователей, принципов регистрации и системы паролей
8. Выполнение резервного копирования БД и восстановление состояния БД на заданную дату.
9. Заключение.
10. Приложения

# **Выполнение анализа и предварительной обработки информации**

В ходе учебной практики необходимо разработать базу данных и модули системы защиты объекта критической информационной инфраструктуры (КИИ).

Любой объект КИИ предполагает пропускной режим на предприятие, независимо от периметра контролируемой и опасной зон. Модуль “ХранительПРО” направлен на организацию пропускного режима и включает в себя:

1. веб-сервис для заказа пропуска на предприятие для гостей и экскурсионных групп;
2. терминал сотрудника общего отдела для формальной проверки поданной заявки;
3. терминал сотрудника охраны для реализации пропускного режима;
4. терминал сотрудника подразделения для учета посещений.

В программе MS Visio 2019 я построила диаграмму вариантов использования.

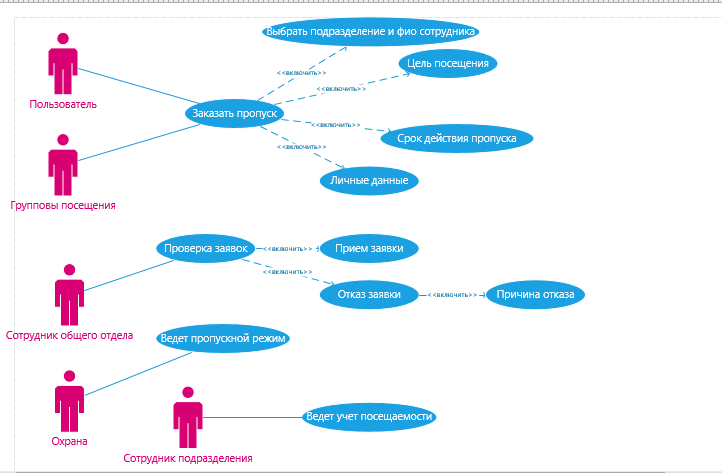


Рисунок - Диаграмма вариантов использования

# **Выделение объектов и атрибутов в соответствии с заданием**

Для базы данных «ХранительПРО» выделены следующие объекты:

- User

- Группа

- Групповое посещение

- Данные пользователя

- Данные сотрудника

- Индивидуальное посещение

- Информация для пропуска

- Код сотрудника

- Назначение

- Отдел

- Подразделение

- Пользователь

- Прикрепляемые документы

- Сотрудник

- Список посетителей

- Статус

- Тип посещения

Сущность user содержит следующие атрибуты:

- id

- login

- Password

Сущность Группа Содержит:

- ID

- Группы

Сущность Групповое посещение содержит следующие атрибуты:

- ID

- Фамилия

- Имя

- Отчество

- Телефон

- E-mail

- Организация

- Примечание

- Дата рождения

- Серия

- Номер

Сущность Данные пользователя содержит следующие атрибуты:

- id

- ФИО

- Номер телефона

- E-mail

- Дата рождения

- Данные паспорта

Сущность Данные сотрудника содержит следующие атрибуты:

- id

- Фамилия

- Имя

- Отчество

Сущность Индивидуальное посещение содержит следующие атрибуты:

- ID

- Фамилия

- Имя

- Отчество

- Телефон

- E-mail

- Организация

- Примечание

- Дата рождения

- Серия

- Номер

- фото паспорта

Сущность Информация для пропуска содержит следующие атрибуты:

- id

- Дата (с)

- Дата (до)

- Цель посещения

Сущность Код сотрудника содержит следующие атрибуты:

- id

- Код сотруника

- ФИО

Сущность Назначение содержит следующие атрибуты:

- id

- Дата

- № кода сотрудника

Сущность отдел содержит следующие атрибуты:

- id

- Отдел

Сущность Подразделение содержит следующие атрибуты:

- id

Подразделение

Сущность Пользователь содержит следующие атрибуты:

- id

- № пользователя

- № Паспортных данных

- № учетных данных

- № назначение

- № группы

Сущность Прикрепляемые документы содержит следующие атрибуты:

- id

- Прикрепляемые документы

Сущность Принимающая сторона содержит следующие атрибуты:

- id

- № подзраделения

- ФИО

Сущность Сотрудник содержит следующие атрибуты:

- id

- № сотрудника

- № подразделения

- № отдела

- № кода сотрудника

Сущность Список посетителя содержит следующие атрибуты:

- id

- ФИО

- Номер телефона

- Фотография

Сущность Статус содержит следующие атрибуты:

- id

- Тип регистрации

- № отдела

- Статус

Сущность Тип посещения содержит следующие атрибуты:

- id

- Тип посещения

# **Проектирование и нормализация БД в полном соответствии с поставленной задачей и применением CASE-средств**

Построение в программе MS Visio 2019 концептуальной модели БД.

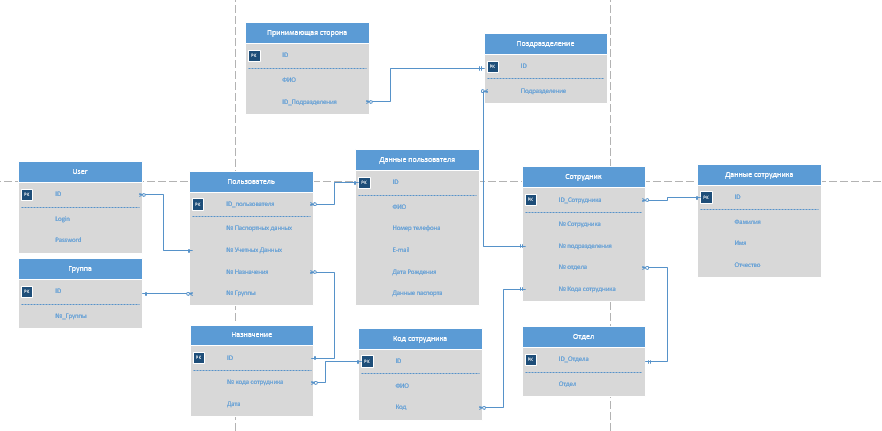


Рисунок - ERD диаграмма

# **Выполнение построения БД в предложенной СУБД и заполнение всех таблиц с помощью соответствующих средств**

Была спроектирована база данных в SQL Server Management Studio (SSMS).

Рисунок - Диаграмма БД

4.1 Словарь БД «Учет успеваемости учащихся 5-9 классов»

Таблица “Принимающая сторона”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | ФИО | Nvarchar(100) | Y |
|  | ID\_Подразделения | int | Y |

Таблица “Подразделение”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | Подразделение | Nvarchar(100) | Y |

Таблица “User”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | Login | Nvarchar(100) | Y |
|  | Password | Nvarchar(100) | Y |

Таблица “Пользователь”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | № Паспортных данных | int | Y |
|  | № Учетных данных | int | Y |
|  | № Назначения | Int | Y |
|  | № Группы | int | Y |

Таблица “Данные пользователя”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | ФИО | Nvarchar(100) | Y |
|  | Телефон | Nvarchar(50) | Y |
|  | Email | Nvarchar(100) | Y |
|  | Дата рождения | Nvarchar(100) | Y |
|  | Данные паспорта | int | Y |

Таблица “Группа”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | № Группы | Int | Y |

Таблица “Назначения”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | № кода сотрудника | int | Y |
|  | Дата | Date | Y |

Таблица “Сотрудник”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | № сотрудника | int | Y |
|  | № подразделения | int | Y |
|  | № отдела | int | Y |
|  | № кода сотрудника | int | Y |

Таблица “Данные сотрудника”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | Фамилия | Nvarchar(50) | Y |
|  | Имя | Nvarchar(50) | Y |
|  | Отчество | Nvarchar(50) | Y |

Таблица “Отдел”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | Отдел | Nvarchar(50) | Y |

Таблица “Код сотрудника”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Data type | Required? |
| PK | ID | int | Y |
|  | ФИО | Nvarchar(100) | Y |
|  | Код | Nvarchar(50) | Y |

# **Реализация уровней доступа для различных категорий пользователей**

Реализовано 2 уровня доступа: Групповое посещение и личное

Посетитель может посмотреть список его заявок или заявок оформленных на него. Также имеет возможность оформить личную или групповую заявку, заполнив все обязательные поля.

Сотрудник имеет возможность работать с заявками.

# **Создание запросов и отчетов в соответствии с заданием**

Запрос для авторизации пользователей

CREATE PROCEDURE AuthenticateUser

@Login NVARCHAR(50),

@Password NVARCHAR(MAX)

AS

BEGIN

SET NOCOUNT ON;

DECLARE @UserID INT;

-- Проверяем существование пользователя с указанным логином и паролем

SELECT @UserID = ID

FROM [dbo].[User]

WHERE Login = @Login AND Password = @Password;

-- Если пользователь найден, возвращаем его ID

IF @UserID IS NOT NULL

BEGIN

SELECT 'Успех' AS Status, @UserID AS UserID;

END

ELSE

BEGIN

-- Если пользователь не найден, возвращаем сообщение об ошибке

SELECT 'Ошибка: Неверный логин или пароль' AS Status;

END

END;GO

Создание хранимой процедуры для авторизации пользователей.

CREATE PROCEDURE AuthenticateUser

@Login NVARCHAR(50),

@Password NVARCHAR(MAX)

AS

BEGIN

SET NOCOUNT ON;

DECLARE @UserID INT;

-- Проверяем существование пользователя с указанным логином и паролем

SELECT @UserID = ID

FROM [dbo].[User]

WHERE Login = @Login AND Password = @Password;

-- Если пользователь найден, возвращаем его ID

IF @UserID IS NOT NULL

BEGIN

SELECT 'Успех' AS Status, @UserID AS UserID;

END

ELSE

BEGIN

-- Если пользователь не найден, возвращаем сообщение об ошибке

SELECT 'Ошибка: Неверный логин или пароль' AS Status;

END

END;

Хранимая процедура для регистрации пользователей

CREATE PROCEDURE RegisterUser

@Login NVARCHAR(50),

@Password NVARCHAR(MAX),

@FullName NVARCHAR(50),

@PhoneNumber NVARCHAR(50),

@Email NVARCHAR(MAX),

@DateOfBirth NVARCHAR(50),

@PassportData NVARCHAR(50),

@EmployeeCode NVARCHAR(50),

@DepartmentID INT,

@RoleID INT -- Предполагается, что есть роль пользователя (например, администратор, обычный пользователь)

AS

BEGIN

SET NOCOUNT ON;

DECLARE @UserID INT;

-- Вставка данных пользователя

INSERT INTO [ХранительПРО].[dbo].[Данные пользователя] ([ФИО], [Номер телефона], [E-mail], [Дата рождения], [Данные паспорта])

VALUES (@FullName, @PhoneNumber, @Email, @DateOfBirth, @PassportData);

SET @UserID = SCOPE\_IDENTITY(); -- Получаем ID только что вставленного пользователя

-- Вставка учетных данных

INSERT INTO [ХранительПРО].[dbo].[User] ([Login], [Password])

VALUES (@Login, @Password);

DECLARE @UserAccountID INT;

SET @UserAccountID = SCOPE\_IDENTITY(); -- Получаем ID только что вставленных учетных данных

-- Связываем данные пользователя с учетной записью

UPDATE [ХранительПРО].[dbo].[Данные пользователя]

SET [№ пользователя] = @UserID

WHERE [ID] = @UserID;

-- Вставка данных сотрудника, если пользователь является сотрудником

IF @EmployeeCode IS NOT NULL

BEGIN

INSERT INTO [ХранительПРО].[dbo].[Данные сотрудника] ([Код сотрудника])

VALUES (@EmployeeCode);

DECLARE @EmployeeID INT;

SET @EmployeeID = SCOPE\_IDENTITY(); -- Получаем ID только что вставленного сотрудника

-- Связываем данные сотрудника с данными пользователя

UPDATE [ХранительПРО].[dbo].[Данные сотрудника]

SET [№ пользователя] = @UserID

WHERE [ID] = @EmployeeID;

-- Связываем данные сотрудника с отделом и подразделением

UPDATE [ХранительПРО].[dbo].[Сотрудник]

SET [№ отдела] = @DepartmentID,

[№ подразделения] = @DepartmentID

WHERE [№ сотрудника] = @EmployeeID;

END;

-- Вставка записи о пользователе

INSERT INTO [ХранительПРО].[dbo].[Пользователь] ([№ пользователя], [№ учетных данных], [№ группы])

VALUES (@UserID, @UserAccountID, @RoleID);

END;

Хранимая процедура для добавления записей в связанные таблицы (например, посещения)

CREATE PROCEDURE AddVisit

@VisitType NVARCHAR(50),

@UserID INT,

@VisitDate DATE,

@AdditionalInfo NVARCHAR(50) = NULL

AS

BEGIN

SET NOCOUNT ON;

-- Вставка информации о посещении

IF @VisitType = 'Individual' -- Индивидуальное посещение

BEGIN

INSERT INTO [ХранительПРО].[dbo].[Индивидуальное посещение] ([Фамилия], [Имя], [Отчество], [Телефон], [E-mail], [Дата рождения], [Организация], [Примечение], [Дата], [Серия], [Номер])

SELECT [Фамилия], [Имя], [Отчество], [Номер телефона], [E-mail], [Дата рождения], [Организация], @AdditionalInfo, @VisitDate, [Серия], [Номер]

FROM [ХранительПРО].[dbo].[Данные пользователя]

WHERE [ID] = @UserID;

END

ELSE IF @VisitType = 'Group' -- Групповое посещение

BEGIN

INSERT INTO [ХранительПРО].[dbo].[Групповое посещение] ([Фамилия], [Имя], [Отчество], [Телефон], [E-mail], [Организация], [Примечение], [Дата рождения], [Серия], [Номер])

SELECT [Фамилия], [Имя], [Отчество], [Телефон], [E-mail], [Организация], @AdditionalInfo, [Дата рождения], [Серия], [Номер]

FROM [ХранительПРО].[dbo].[Данные пользователя]

WHERE [ID] = @UserID;

END;

END;

# **Создание и обоснование групп пользователей, принципов регистрации и системы паролей.**

Регистрация пользователя происходит с помощью заполнения всех обязательных полей.

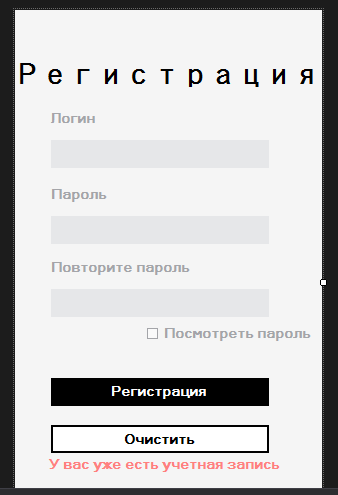


Рисунок - регистрация посетителя

Авторизация пользователя происходит с помощью заполнения всех обязательных полей.

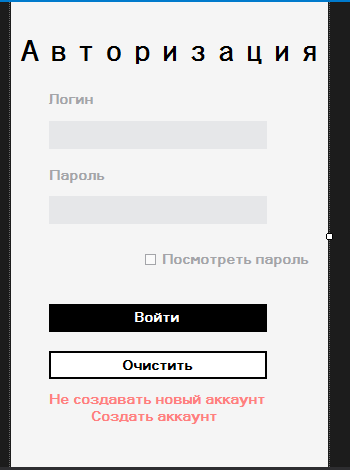


Рисунок - авторизация посетителя

# **Выполнение резервного копирования БД и восстановление состояния БД на заданную дату.**

Выполнение резервного копирования через графический интерфейс

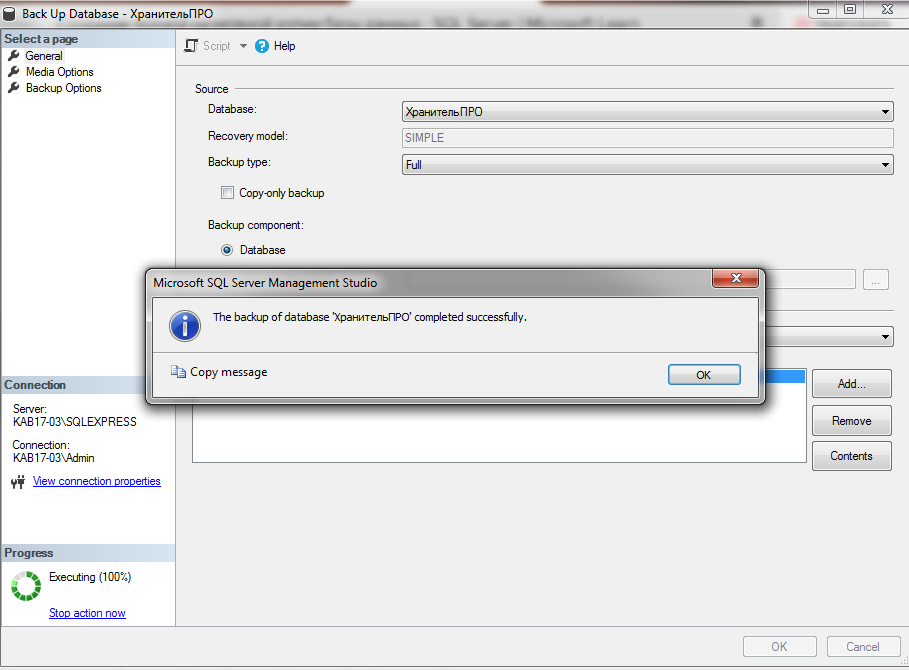


Рисунок - Резервное копирование

# **Заключение**

В ходе учебной практики по модулю ПМ.11. Разработка, администрирование и защита баз данных была создана база данных «ХранительПРО», которая хранит информацию о посетителях, сотрудниках и заявках. Также были созданы вспомогательные таблицы.

Я создал запросы, хранимые процедуры.

Написал приложение для авторизации пользователей.

Для выполнения заданий были использованы все знания, полученные за годы обучения в колледже.