МИНИСТЕРСТВО ОБРАЗОВАНИЯ КИРОВСКОЙ ОБЛАСТИ

Кировское областное государственное профессиональное образовательное

бюджетное учреждение

«Слободской колледж педагогики и социальных отношений»

**ОТЧЕТ**

**по производственной практике**

**ПМ.11. Разработка, администрирование и защита баз данных**

**Тема: «Разработка базы данных для учета заявок на ремонт компьютерной техники в сервисном центре»**

Студент

Чемоданов Павел Юрьевич

Группа 20П-1

Специальность 09.02.07 Информационные системы и программирование

Руководитель практики от колледжа:

*Махнев Александр Анатольевич*

Руководитель практики от организации:

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* *Павленко Сергей Михайлович*

подпись

УТВЕРЖДАЮ:

Директор:

Устюжанинов Вадим Борисович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
 Наименование организации

ООО "ЭЛЕКТРОН-СОФТ"

\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись расшифровка

М. П.

2024-2025 уч. 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 к отчету: диск со скриптами БД (1 - пустая, 2 – заполненная демонстрационными данными), резервные копии БД, файлы БД, проект приложения для работы с БД (интерфейс, операции CRUD), отчет в электронном виде, презентация для выступления и др. материалы.
11. **Выполнение анализа и предварительной обработки информации**

**1.1 Цель системы**

Разработать базу данных, которая позволит эффективно учитывать и управлять заявками на ремонт компьютерной техники, включая данные клиентов, оборудования, мастеров, статусы заявок, оплаты и сроки.

**1.2 Основные процессы в сервисном центре**

* Клиент обращается в сервис с заявкой на ремонт техники.
* Оператор или мастер создает заявку, фиксирует данные клиента и техники.
* Назначается мастер, который будет выполнять ремонт.
* Статус заявки меняется по мере выполнения (новая, в работе, завершена, отказ).
* Фиксируется стоимость и оплата ремонта.
* Клиент получает уведомление о готовности.

**1.3 Какие данные нужно хранить**

| **Категория** | **Пример данных** |
| --- | --- |
| Клиенты | ФИО, телефон, email, адрес |
| Техника | Тип (ноутбук, ПК, принтер), модель, серийный номер, дата покупки |
| Заявки | Номер заявки, дата создания, описание проблемы, статус, дата закрытия |
| Мастера | ФИО, специализация, телефон |
| Статусы | Наименование статуса заявок (принята, в работе, завершена и т.д.) |
| Оплата | Сумма, дата оплаты, способ оплаты |

1. **Выделение объектов и атрибутов в соответствии с заданием**

**1. Клиент**

* **ID\_Клиента** (int, PK, автонумерация) — уникальный идентификатор клиента
* **ФИО** (nvarchar(100)) — полное имя клиента
* **Телефон** (nvarchar(20)) — контактный номер телефона
* **Email** (nvarchar(50), nullable) — электронная почта
* **Адрес** (nvarchar(200), nullable) — адрес клиента

**2. Техника**

* **ID\_Техники** (int, PK, автонумерация) — уникальный идентификатор техники
* **ID\_Клиента** (int, FK) — ссылка на владельца (клиента)
* **Тип\_техники** (nvarchar(50)) — например, "Ноутбук", "ПК", "Принтер"
* **Модель** (nvarchar(100)) — модель техники
* **Серийный\_номер** (nvarchar(50), nullable) — серийный номер техники
* **Дата\_покупки** (date, nullable) — дата покупки (если известна)

**3. Мастер**

* **ID\_Мастера** (int, PK, автонумерация) — уникальный идентификатор мастера
* **ФИО** (nvarchar(100)) — имя мастера
* **Телефон** (nvarchar(20), nullable) — контактный номер
* **Специализация** (nvarchar(100)) — направления работы мастера

**4. Статус заявки**

* **ID\_Статуса** (int, PK) — уникальный идентификатор статуса
* **Название\_статуса** (nvarchar(50)) — например, "Принята", "В работе", "Завершена", "Отказ"

**5. Заявка**

* **ID\_Заявки** (int, PK, автонумерация) — уникальный номер заявки
* **ID\_Клиента** (int, FK) — клиент, сделавший заявку
* **ID\_Техники** (int, FK) — техника, относящаяся к заявке
* **ID\_Мастера** (int, FK, nullable) — мастер, назначенный на ремонт
* **ID\_Статуса** (int, FK) — текущий статус заявки
* **Дата\_создания** (datetime) — дата и время создания заявки
* **Описание\_проблемы** (nvarchar(max)) — описание неисправности
* **Стоимость** (decimal(10,2), nullable) — цена ремонта
* **Дата\_завершения** (datetime, nullable) — дата завершения ремонта

**6. Оплата**

* **ID\_Оплаты** (int, PK, автонумерация) — уникальный идентификатор оплаты
* **ID\_Заявки** (int, FK) — заявка, по которой произведена оплата
* **Сумма** (decimal(10,2)) — сумма оплаты
* **Дата\_оплаты** (datetime) — дата оплаты
* **Способ\_оплаты** (nvarchar(50)) — например, "Наличные", "Банк.картой", "Перевод"

**2.2 Важные связи (FK)**

* Техника связана с Клиентом по ID\_Клиента (один клиент — много устройств)
* Заявка связана с Клиентом, Техникой, Мастером и Статусом
* Оплата связана с Заявкой (может быть несколько оплат по одной заявке)

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

Спроектирована ER-диаграмма, отражающая связи между таблицами (Рисунок 1):

**Клиент — Техника**: Один клиент может иметь много устройств

**Техника — Заявка**: Одно устройство может иметь много заявок на

**Мастер — Заявка**: Один мастер может выполнять много заявок

**Статус — Заявка**: Один статус может относиться ко многим заявкам

**Заявка — Оплата**: По одной заявке может быть несколько оплат

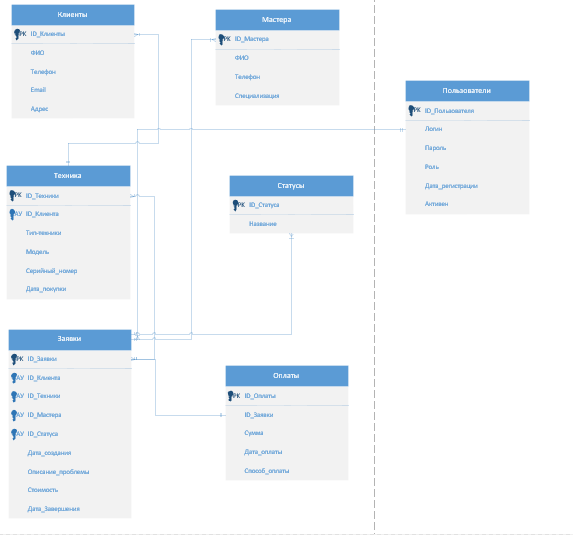


Рисунок 1 – ER-диаграмма

Проведена нормализация до 3НФ для устранения избыточности и обеспечения целостности данных.

1. **Построение БД в SQL Server**

База данных реализована в СУБД SQL Server. Созданы таблицы(Рисунок 2):

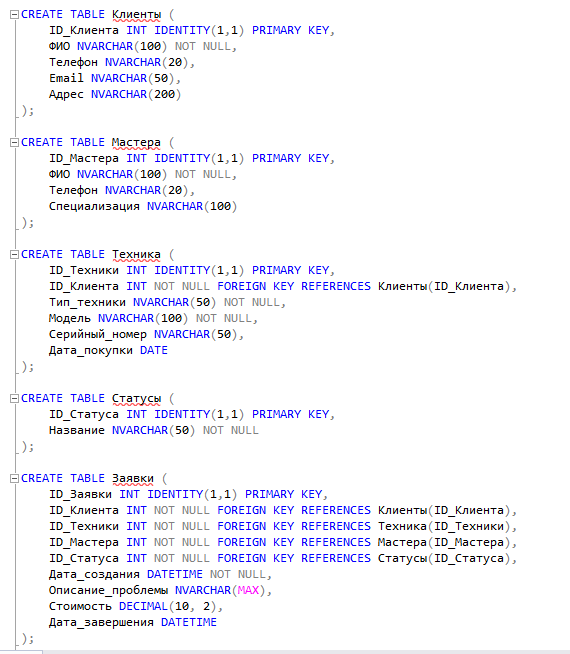


Рисунок 2 – Запрос на создание таблиц

Заполнены тестовыми данными (Рисунок 3)

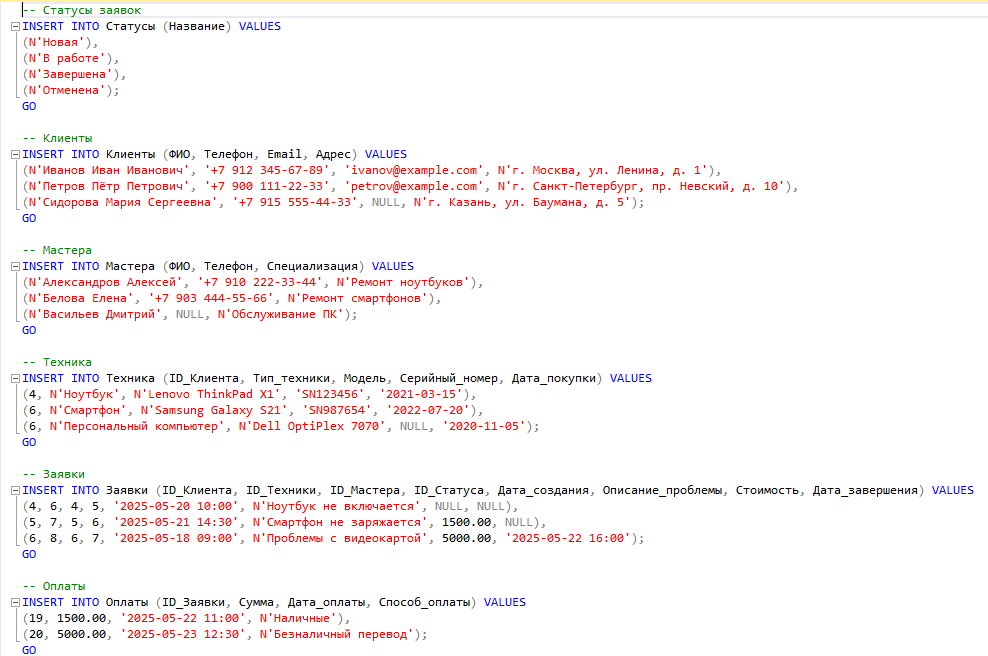


Рисунок 3 – Запрос на добавление данных

1. **Реализация уровней доступа**

Настроены роли (Рисунок 4):

**Администратор** — управляет системой, пользователями, доступом, настройками.

**Мастер** — работает с заявками, фиксирует ремонт, ставит статусы.

**Клиент** — создает заявки, отслеживает статус своих заказов.

**Оператор/Менеджер** — контролирует процесс ремонта и отчёты.

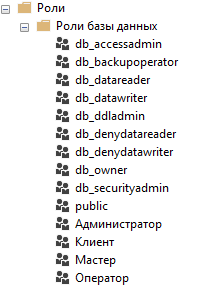


Рисунок 4 – Настройка ролей

1. **Создание запросов и отчетов**

Разработаны SQL-запросы для:

Отчёт по количеству заявок по статусам (Рисунок 5)

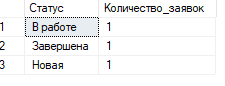


Рисунок 5 – Отчёт по количество заявок по статусам

Отчёт по выгрузке всех клиентов с количеством их заявок (Рисунок 6)

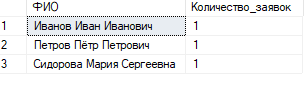


Рисунок 6 – Отчёт по выгрузке всех клиентов с количеством их заявок

Запрос на вывод всех заявок с клиентами и статусами (Рисунок 7)

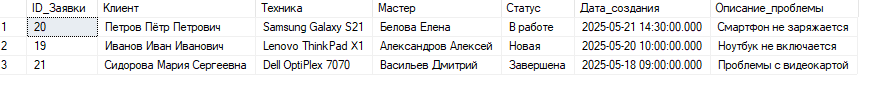


Рисунок 7 – Запрос на вывод всех заявок с клиентами и статусами

1. **Группы пользователей и безопасность**

Реализована система аутентификации:

* При регистрации пароль хэшируется и сохраняется.
* При входе введённый пароль хэшируется и сравнивается с сохранённым хэшем.

1. **Резервное копирование**

Настроено регулярное резервное копирование (Рисунок 8)

BACKUP DATABASE [СервисЦентрРемонта]

TO DISK = 'C:\Program Files\Microsoft SQL Server\MSSQL16.KAMI0YASU\MSSQL\Backup\СервисЦентрРемонта.bak'

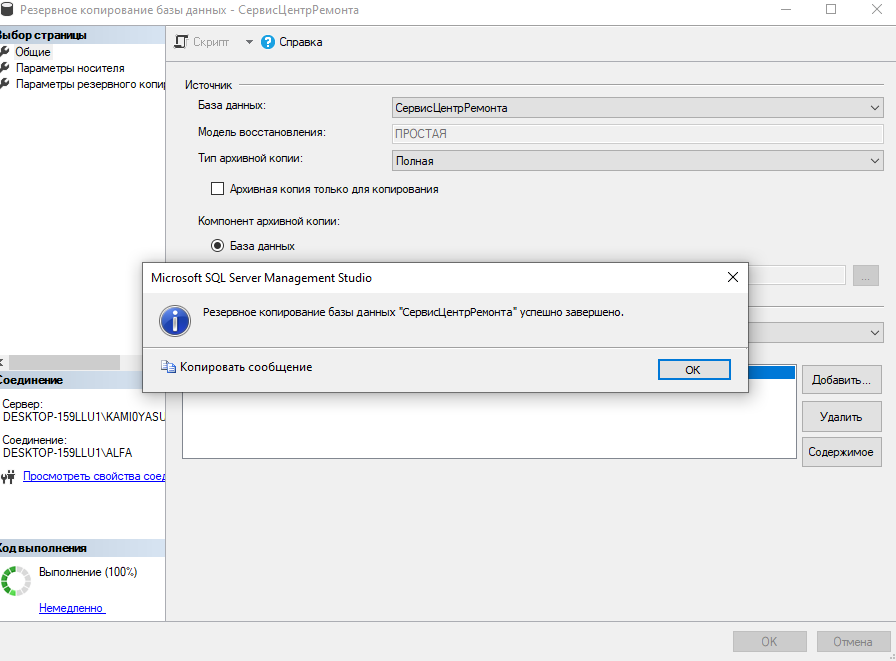


Рисунок 8 – Резервное копирование

Проверено восстановление из резервной копии (Рисунок 9)

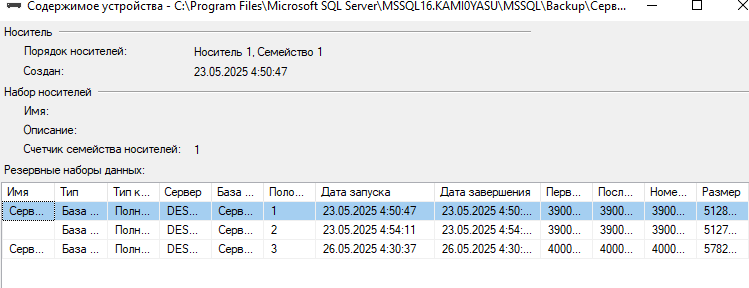


Рисунок 9 – Проверка восстановления резервной копии

1. **Заключение**

В ходе производственной практики спроектирована и реализована база данных для учёта заявок в сервисном центре. Проект охватывает работу с SQL Server, нормализацию, разграничение прав, резервное копирование и интеграцию с WPF-приложением. База данных соответствует требованиям и готова к внедрению.

1. **Приложения**

Ссылка на Github - https://github.com/kaminomi1/PM\_11

Файлы SQL-скриптов.

Резервные копии БД.

ER-диаграмма (в виде изображения или проекта CASE-средства).