|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

**ТЕХНИЧЕСКИЙ ПРОЕКТ**

на разработку

\_\_информационной системы для ювелирного магазина

2024

**Содержание**

[1. Требования к системе 2](#_Toc181200019)

[1.1. Функциональные требования 2](#_Toc181200020)

[1.2. Нефункциональными требованиями к ИС являются: 3](#_Toc181200021)

[1.3. Интерфейсные требования 3](#_Toc181200022)

[1.3.1. Пользовательский интерфейс и взаимодействие 3](#_Toc181200023)

[1.3.2. Поддержка мультиязычности 3](#_Toc181200024)

[2. Уточнение структуры данных 4](#_Toc181200025)

[2.1. Выделение сущностей 4](#_Toc181200026)

[2.2. Построение логической ER-диаграммы 4](#_Toc181200027)

[2.3. Построение диаграммы потоков данных 5](#_Toc181200028)

[3. Укрупненный сценарий использования 8](#_Toc181200029)

[4. Диаграмма вариантов использования 10](#_Toc181200030)

[4.1. Начать процесс продажи 10](#_Toc181200031)

[4.2. Редактирование продажи 10](#_Toc181200032)

[4.3. Процесс формирования отчета 11](#_Toc181200033)

[5. Уточнение структуры системы 12](#_Toc181200034)

[6. Пользовательские интерфейсы 13](#_Toc181200035)

[8. Определение языка разработки 26](#_Toc181200036)

[9. Определение системы управления базами данных 27](#_Toc181200037)

[10. Определение библиотеки для создания пользовательского интерфейса 28](#_Toc181200038)

1. Требования к системе

Основной процесс в виде диаграммы IDEF0 показан на рисунке 1.

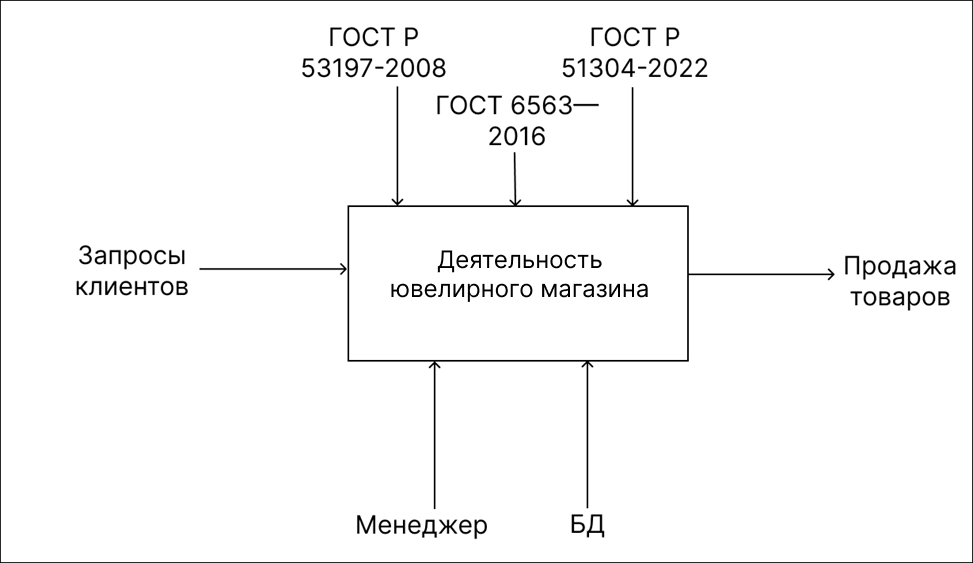


Рисунок 1 – Диаграмма IDEF0

* 1. Функциональные требования

Система реализации продукции ювелирной мастерской должна предоставлять возможность:

* Менеджеру иметь доступ к полному, актуальному списку товаров, имеющихся в ювелирной мастерской, с информацией о них.
* Менеджеру иметь доступ к полному, актуальному списку поставщиков, обслуживающих ювелирную мастерскую, с информацией о них.
* Менеджеру иметь доступ к полному, актуальному списку по продажам в ювелирной мастерской, с информацией о них.
* Менеджеру возможность формирования отчета по продажам в ювелирной мастерской.
* Менеджеру управлять списком товаров, иметь функции изменения, добавления и удаления товаров из списка.
* Менеджеру управлять списком категорий, иметь функции изменения, добавления и удаления категорий из списка.
* Менеджеру управлять списком продаж, иметь функции изменения, добавления и удаления продаж из списка.
* Менеджеру управлять списком поставщиков, иметь функции изменения, добавления и удаления поставщиков из списка.

Создаваемая система будет работать с БД, которая содержит всю информацию о товарах, категорий товаров, поставщиков и продажах.

## Нефункциональными требованиями к ИС являются:

* требования к производительности:

Система должна обеспечивать быстрый доступ к товарам, а также быструю обработку запросов на оформление продажи и создания отчета.

* требования к надежности:

1. ИС должна быть стабильной и надежной, чтобы избежать потери данных или сбоев в работе.;
2. Каждый элемент интерфейса должен быть рабочим, срабатывать при нажатии, что бы не возникало проблем с использованием;
3. интерфейс должен четко отображать все компоненты приложения на экране.

1.3. **Интерфейсные требования**

1.3.1. Пользовательский интерфейс и взаимодействие

Интерфейс должен быть простым с понятными элементами управления и навигацией. Все данные должны быть представлены в удобной для восприятия форме (графики, таблицы, списки).

1.3.2. Поддержка мультиязычности

Интерфейс должен поддерживать только русский язык, мультиязычность не предусмотрена.

1. Уточнение структуры данных

В ходе уточнения данных были выполнены следующие задачи, показанные в Таблице 2.

Таблица 2 – Задачи для уточнения структуры данных.

|  |  |  |  |
| --- | --- | --- | --- |
| **№ задачи** | **Название** | **Содержание** | **Результат** |
| 1 | Выделение сущностей | Выделение основных сущностей программы, а также их описание | Готовая таблица с сущностями и их описанием |
| 2 | Построение логической ER-диаграммы | Схематическое представление модели базы данных в виде логической ER-диаграммы | Готовая логическая ER-диаграмма |
| 3 | Построение диаграммы потоков данных | Определить, как будут передвигаться данные в программе; построить DFD диаграмму | Готовая диаграмма потоков данных (DFD) |

* 1. Выделение сущностей

Краткое описание основных сущностей показано в Таблице 3.

Таблица 3 – Сущности и их описание

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Название сущности** | **Описание сущности** |
| 1 | Товары | Единицы продукции, доступные для продажи, содержащие информацию о названии, цене и характеристиках. |
| 2 | Категории товаров | Группы товаров, объединенные по общим признакам, например, по типу или назначению. |
| 3 | Поставщики | Компании или лица, предоставляющие товары для пополнения ассортимента. |
| 4 | Продажи | Операции, фиксирующие данные о проданных товарах, покупателях и сумме продаж. |
| 5 | Материал | Основные материалы или компоненты, из которых изготовлены товары. |

* 1. Построение логической ER-диаграммы

Логическая ER-диаграмма показана на рисунке 2.

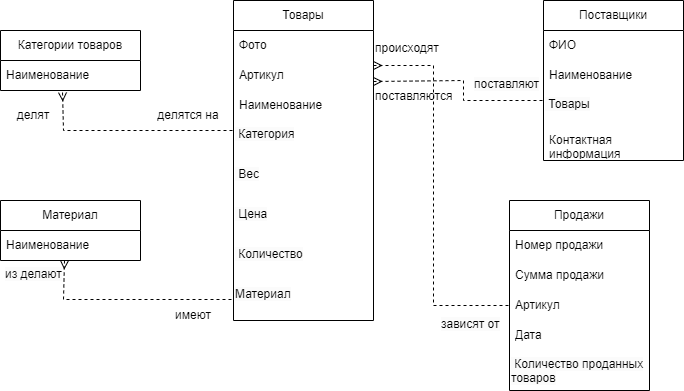


Рисунок 2 – Логическая модель базы данных

* 1. Построение диаграммы потоков данных

DFD (Data Flow Diagram) – это графическое представление потоков данных в системе, показывающее, как информация перемещается через различные процессы. DFD используется для моделирования и анализа информационных систем, позволяя представить их структуру, функции и потоки данных между различными компонентами.

На рисунке 3 представлена DFD-диаграмма для ювелирного магазина.

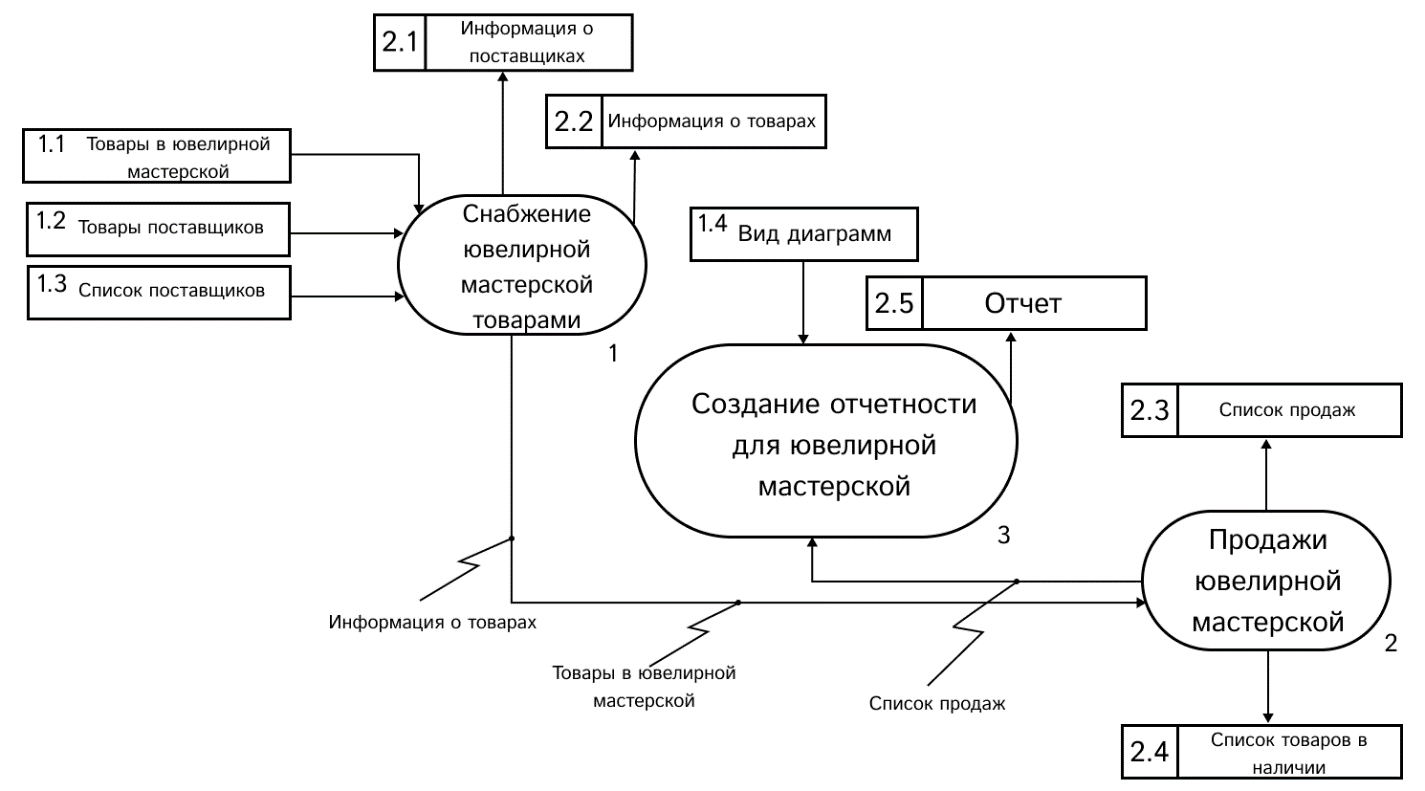


Рисунок 3 – DFD диаграмма

На данной диаграмме присутствую 3 блока работы: «Снабжение ювелирной мастерской товарами», «Создание отчетности для ювелирной мастерской», «Продажи ювелирной мастерской».

«Снабжение ювелирной мастерской товарами» содержит внешние данные: «Товары в ювелирной мастерской» (список товаров в наличии в ювелирной мастерской), «Товары поставщиков» (список товаров, которые приходят от определенного поставщика) и «Список поставщиков» (список поставщиков, которые поставляют товары в ювелирную мастерскую)

«Создание отчетности для ювелирной мастерской» содержит внешние данные «Вид диаграмм».

Блок «Снабжение ювелирной мастерской товарами» содержит хранилище данных:

• «Информация о товарах», где находится такая информация о товарах как, фото, артикул, категория, материал, цена, вес, наименование и количество.

• «Информация о поставщиках», где находится такая информация о поставщиках как, ФИО поставщика, наименование организации, товары от поставщика и контактная информация.

Блок «Создание отчетности для ювелирной мастерской» содержит хранилище данных: «Отчет», где можно создать отчет по продажам в ювелирной мастерской, там находится информация о дате и времени продаж за определенный период, сумме продаж и тп.

Блок «Продажи ювелирной мастерской» содержит хранилище данных:

• «Список продаж», где находится такая информация о продажах как, номер продажи, сумма продажи, дата и время, артикул и количество проданных изделий.

• «Список товаров в наличии», где находится информация о товарах в наличии в ювелирной мастерской.

1. Укрупненный сценарий использования

Пользователь с ролью Менеджер заходит в приложение, и система отображает список текущих товаров в магазине и варианты последующего использования этих товаров при продаже, поступлении от заказчиков, составлении отчета.

В верхней части экрана пользователь выбирает действие, которое хочет произвести в системе.

В правой части экрана пользователь выбирает действе «Добавить»/«Изменить» для этого выбирает соответствующую кнопку. Открывается форма, в которой менеджер может заполнить/изменить данные о товаре, такие как фото, артикул, наименование, категория, вес, цена, количество. После добавления/изменения новые данные отобразятся в списке товаров. В левой части пользователь выбирает нажимает кнопку «Категория» открывается окно со списком категорий товаров. Менеджер выбирает категорию и в списке товаров отображаются товары только выбранной категории. Пользователь снова выбирает кнопку «Категория» открывается этаже форма, пользователь выбирает кнопку «Сбросить фильтрацию» и список товаров отображается без фильтрации. Пользователь выбирает кнопку «Добавить»/«Изменить» и открывается окно, в котором пользователь может ввести/изменить данные о наименовании категории. Пользователь выбирает кнопку «Категория» выбирает в списке нужную категорию и нажимает кнопку «Изменить», в открывшемся окне выбирает кнопку «Удалить», после удаления система обновляет список категорий. Менеджер выбирает кнопку «Категория», в открывшемся окне в поле для ввода пишет нужную категорию, в общем списке выделяется искомая категория.

Пользователь выбирает действие открыть окно «Поставщики» для этого он нажимает на кнопку «Поставщики». В данном окне отображается полный список поставщиков товаров. В правой части пользователь выбирает кнопку «Добавить»/«Изменить» и открывается форма в которой менеджер может заполнить/изменить данные о поставщике, такие как фио, наименование организации, наименование товара, контактная информация. В окне с изменением данных о поставщике пользователь выбирает кнопку «Удалить поставщика», происходит удаление поставщика из общего списка. После добавления, изменения, удаления данных система обновляет список поставщиков для отображения актуальных данных.

Пользователь выбирает действие открыть окно «Продажи» для этого он нажимает на кнопку «Продажи». В данном окне отображается полный список продаж товаров в магазине. В правой части пользователь выбирает кнопку «Добавить»/«Изменить» и открывается форма в которой менеджер может заполнить/изменить данные о товаре, его артикул и количество. Такие данные как номер продажи, время продажи, общее количество проданных товаров и сумма продажи рассчитывается автоматически. В окне с изменением данных о продаже пользователь выбирает кнопку «Удалить продажу», происходит удаление продажи из общего списка. После добавления, изменения, удаления данных система обновляет список продаж для отображения актуальных данных.

Пользователь выбирает действие открыть окно «Отчеты» для этого он нажимает на кнопку «Отчеты». В данном окне присутствует кнопка «Сформировать». При нажатии на данную кнопку отрывается окно «Сформировать отчет», в котором пользователь выбирает данные, по которым хочет сформировать отчет, такие поля как период, категория, вид диаграммы, затем выбирает кнопку «Сформировать отчет» и в основном окне «Отчеты» формируется отчет по выбранным данным.

1. Диаграмма вариантов использования

В следующих пунктах рассмотрим основные функции информационной системы для ювелирного магазина.

4.1. Начать процесс продажи

Описание: пользователь имеет возможность создать новую продажу в окне «Продажи».

Предварительные условия: наличие товаров в системе.

Нормальное направление:

* пользователь переходит в окно «Продажи»;
* пользователь выбирает кнопку «Добавить»;
* открывается окно, где необходимо заполнить данные о продаже;
* поле заполнения данных, пользователь нажимает кнопку «Добавить продажу»;
* система автоматически заполняет поля номер продажи, дата продажи, сумма, общее количество товаров в данной продаже;
* система автоматически обновляет список продаж;
* система автоматически добавляет новую продажу в общий список продаж;

Выходные условия: в системе создан новая продажа, пользователь переходит в основной список продаж, где отображается новая продажа.

4.2. Редактирование продажи

Описание: пользователь имеет возможность вносить изменения в существующую продажу.

Предварительные условия: наличие товаров в системе.

Нормальное направление:

* пользователь выбирает нужную продажу из списка в окне «Продажи»;
* пользователь выбирает кнопку «Изменить»;
* открывается форма редактирования продажи, с данными, которые были внесены при создании продажи;
* пользователь вносит изменения в данных продажи;
* пользователь нажимает кнопку «Сохранить изменения»;
* система сохраняет изменения;
* система обновляет и отображает общий список продаж.

Выходные условия: продажа обновлена и сохранена в системе с учётом внесённых изменений.

4.3. Процесс формирования отчета

Описание: пользователь может сформировать отчет по продажам за определенный период по отпряденной категории, отобразить в виде выбранной диаграммы.

Предварительные условия: наличие продаж.

Нормальное направление:

* пользователь переходит в окно «Отчеты»;
* пользователь выбирает кнопку «Сформировать»;
* пользователь выбирает данные в полях «Период», «Категория», «Вид диаграммы»;
* пользователь нажимает кнопку «Сформировать отчет», после выбора необходимых данных;
* система автоматически формирует отчет по выбранным данным;
* система автоматически производит подсчет общей выручки по выбранной категории;
* система автоматически отображает данные в виде выбранной диаграммы.

Выходные условия: сформированный отчет по выбранным данным.

1. Уточнение структуры системы

При уточнении структуры системы была описана подробная информация об основных объектах показана в Таблице 4.

Таблица 4 – Информация об объектах

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Название сущности | Название атрибута | Тип |
| 1 | Список товаров | Фото | Фото |
| Артикул | Целое число |
| Наименование | Текст |
| Категория | Текст |
| Вес | Число с плавающей запятой |
| Цена | Целое число |
| Количество | Целое число |
| 2 | Список категорий | Наименование | Текст |
| 3 | Список поставщиков | ФИО | Текст |
| Наименование организации | Текст |
| Товары | Целое число |
| Контактная информация | Целое число |
| 4 | Список продаж | Номер продажи | Целое число |
| Сумма | Целое число |
| Дата | Текст |
| Артикул | Целое число |
| Количество проданных товаров | Целое число |

1. Пользовательские интерфейсы

Ниже представлены результаты пользовательского интерфейса приложения.

Главное окно приложения предоставлено на рисунке 4.

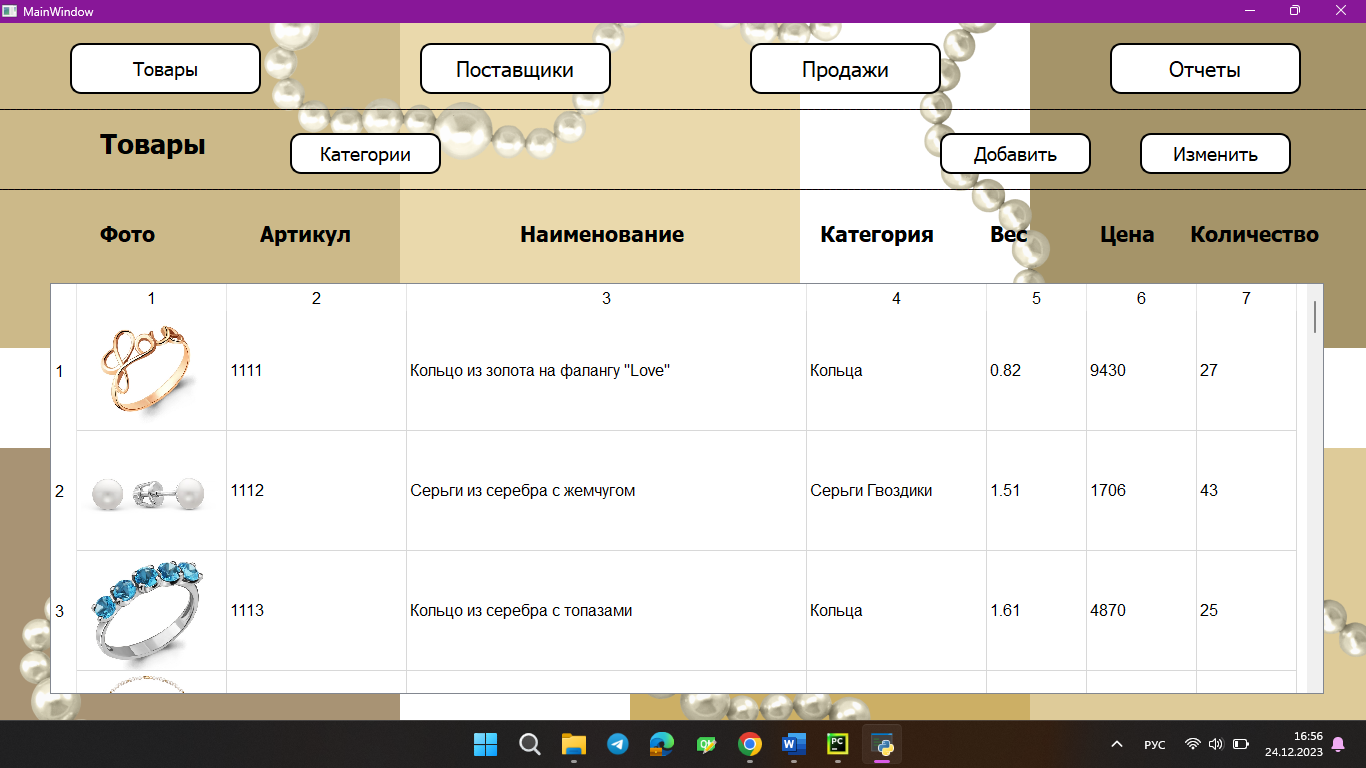


Рисунок 4 – Главное окно приложения

При запуске приложения открывается главное окно и происходит подключение к удаленной базе данных. Само приложение написано при помощи библиотеки PyQT5 с использованием QT Designer для создания окон приложения.

В приложении есть четыре главные кнопки «Товары», «Поставщики», «Продажи» и «Отчеты», при нажатии на которые открываются определенные окна.

В самом Главном окне находится информация о всех товарах ювелирной мастерской.

При нажатии нажатии на кнопку «Категории» в главном окне, открывается диалоговое окно со списком категорий. Там можно найти категорию, или сделать фильтрацию товаров по выбранной категории при нажатии на категорию ЛКМ.

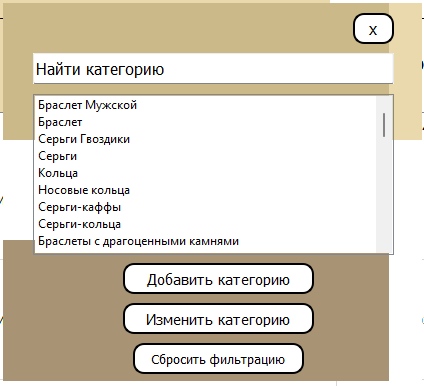


Рисунок 5 – диалоговое окно Категории

В окне есть три кнопки «Добавить категорию», «Изменить категорию» и «Сбросить фильтрацию».

При нажатии на кнопку «Сбросить фильтрацию» фильтрация по выбранной категории убирается в Главном окне.

При нажатии на кнопку «Добавить категорию», открывается диалоговое окно, где мы можем внести информацию о новой категории. Также в окне есть две кнопки «Сохранить изменения» и «Удалить категорию», при внесении данных и после нажатия на кнопку «Сохранить изменения» внесенные данные о новой категории сохраняются в БД и в окно Категории. При нажатии на кнопку «Удалить категорию» внесенные данные в поле Название стираются.

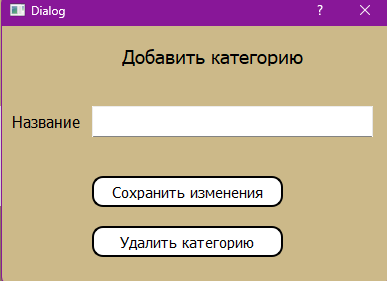


Рисунок 6 – диалоговое окно Добавить категорию

В окне Категории выборе категории ПКМ и нажатии на кнопку «Изменить категорию» открывается диалоговое окно, где можно изменить информацию о выбранной категории.

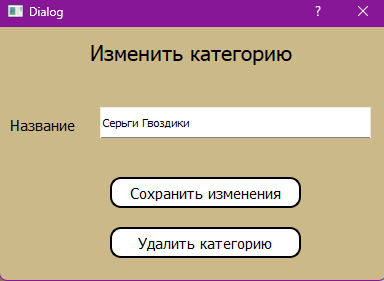


Рисунок 7 – диалоговое окно Изменить категорию

В окне «Изменить категорию» есть две кнопки «Сохранить изменения» и «Удалить категорию», при внесении новых данных и после нажатия на кнопку «Сохранить изменения» внесенные данные о категории сохраняются в БД и в окно Категории. При нажатии на кнопку «Удалить категорию» данные о категории удаляются из БД и приложения.

Еще в Главном окне присутствуют две кнопки «Добавить» и «Изменить», при нажатии на кнопку «Добавить» открывается диалоговое окно «Добавить товар».

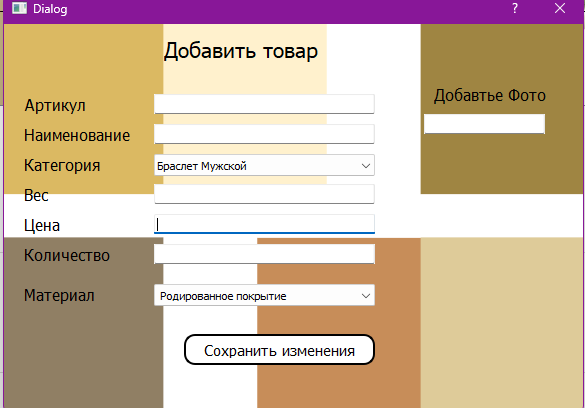


Рисунок 8 – диалоговое окно Добавить товар

В диалоговом окне «Добавить товар» мы можем внести данные о новом товаре: добавить Фото товара, Артикул, Наименование, выбрать Категорию, Вес, Цена, Количество и выбрать Материал товара. После внесения данных о новом товаре, при нажатии на кнопку «Сохранить изменения», данные о новом товаре сохраняются в БД и в приложение.

При выборе определенного артикула в главном окне и нажатии на кнопку «Изменить», открывается диалоговое окно «Изменить товар» с данными о выбранном товаре.

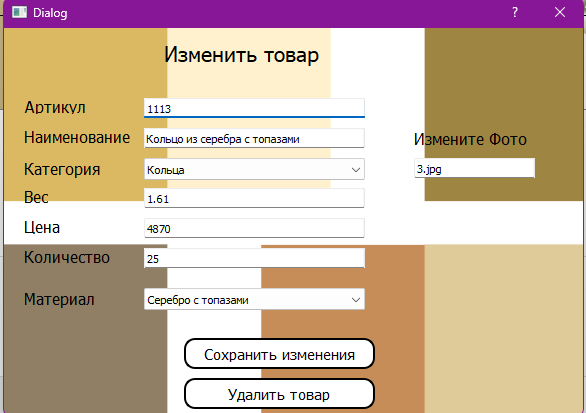


Рисунок 9 – диалоговое окно Изменить товар

В окне «Изменить товар», мы можем поменять нужную нам информацию для выбранного товара. После внесенных изменений при нажатии на кнопку «Сохранить изменения» внесенные данные о товаре сохраняются в БД и в Главном окне. При нажатии на кнопку «Удалить товар» данные о товаре удаляются из БД и из Главного окна.

В главном окне при нажатии на кнопку «Поставщики» открывается окно с информацией о поставщиках ювелирной мастерской.

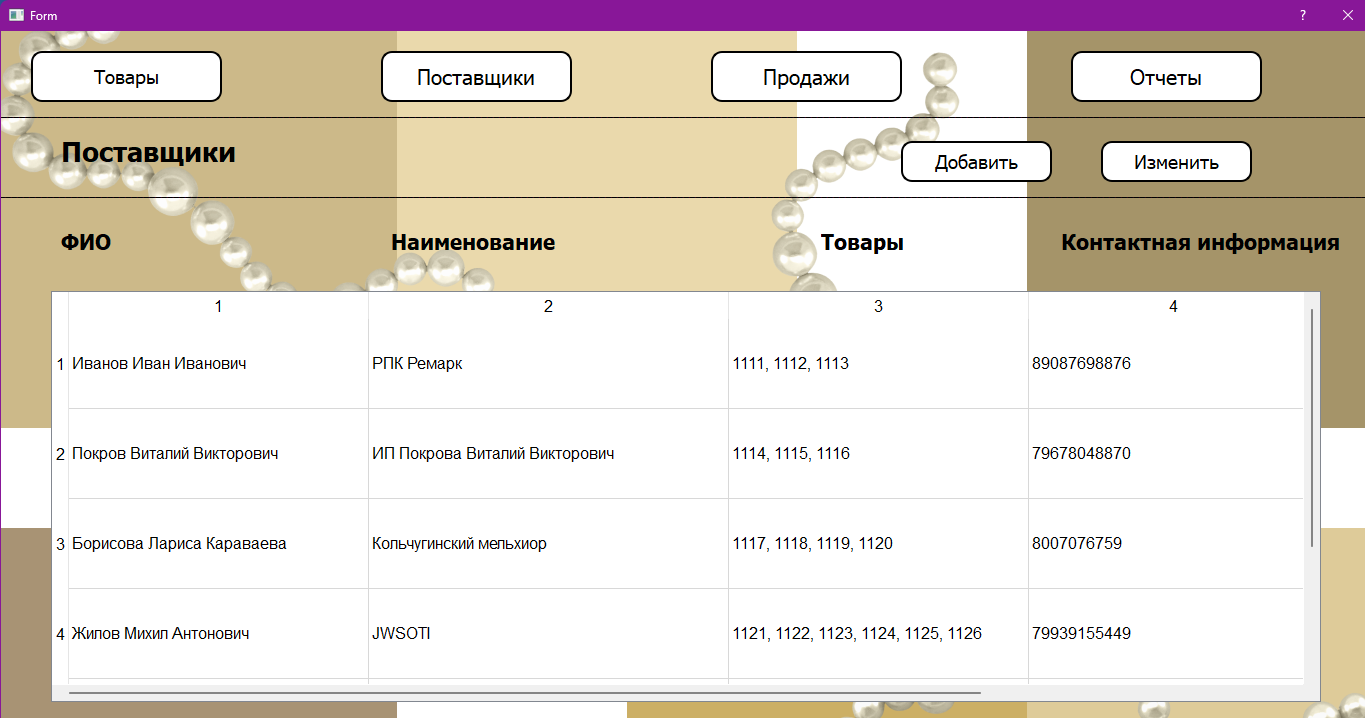


Рисунок 10 – окно Поставщики

В окне «Поставщики» также остаются четыре главные кнопки «Товары», «Поставщики», «Продажи» и «Отчеты», при нажатии на которые открываются определенные окна.

При нажатии на кнопку «Добавить» в окне «Поставщики», открывается диалоговое окно «Добавить поставщика». В данном окне мы можем внести данные о новом поставщике: ФИО, Наименование организации, Контактные данные и при нажатии на кнопку «Добавить товар», добавить товары поставщика. После внесения новых данных при нажатии на кнопку «Сохранить изменения» данные о новом поставщике сохраняются в БД и в окно «Поставщики».

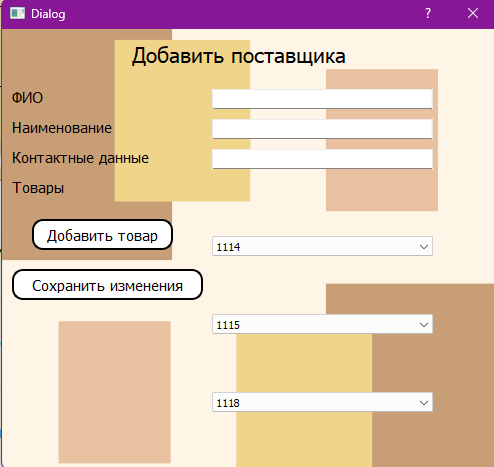


Рисунок 11 – диалоговое окно «Добавить поставщика»

При выборе определенного поставщика и нажатии на кнопку «Изменить», открывается диалоговое окно «Изменить поставщика». В данном окне прописывается информация о выбранном поставщике, которую можно изменить. После внесения изменений при нажатии на кнопку «Изменить поставщика» новые измененные данные сохраняются в БД и в окне «Поставщики». Если же мы ходит удалить выбранного поставщика мы нажимаем на кнопку «Удалить поставщика» и все данные поставщика удаляются из БД и приложения.

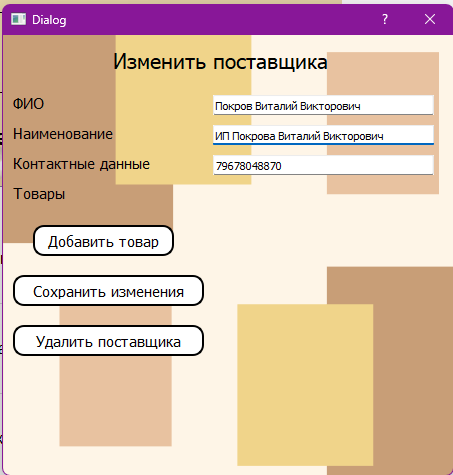


Рисунок 12 – диалоговое окно «Изменить поставщика»

При нажатии на кнопку «Продажи» открывается окно с информацией о продажах в ювелирной мастерской.

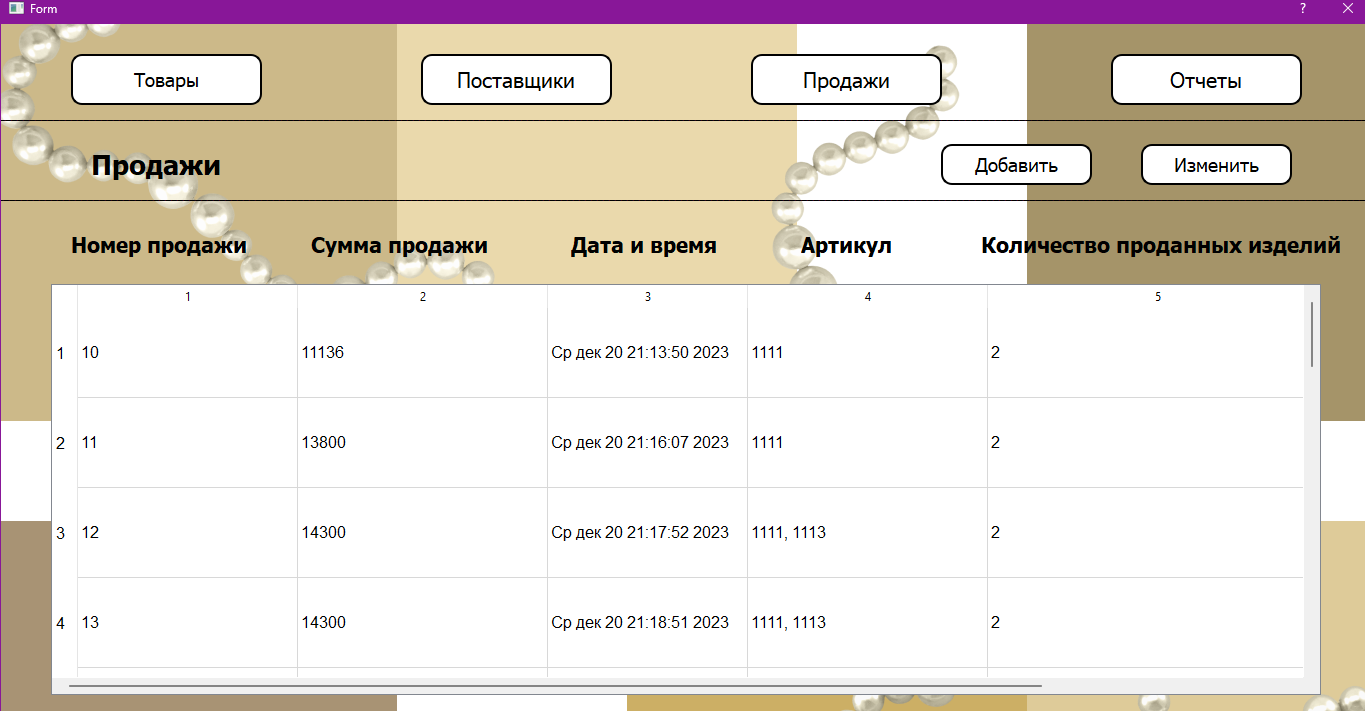


Рисунок 13 – окно «Продажи»

В данном окне также присутствую четыре главные кнопки перехода между окнами. При нажатии на кнопку «Добавить» в окне «Продажи», открывается диалоговое окно «Добавить продажу». В данном окне мы можем внести данные о новой продаже, а точнее о проданных товаров. Данные о Номере продажи, Дата и время продажи пишутся автоматически при открытии окна. После добавления проданных товаров при нажатии на кнопку «Добавить товар», данные о Сумме продажи и Количестве высчитываются автоматически. После внесения проданных товаров при нажатии на кнопку «Добавить продажу» данные о новой продажи сохраняются в БД и в окно «Продажи».

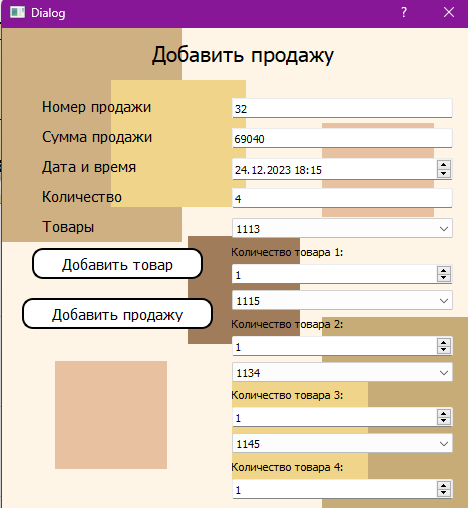


Рисунок 14 – диалоговое окно «Добавить продажу»

При выборе продажи и нажатии на кнопку «Изменить», открывается диалоговое окно «Изменить продажу», где модно поменять информацию о проданных товарах. После внесенных изменений мы можем сохранить изменения при нажатии на кнопку «Сохранить изменения», или удалить продажу при помощи кнопки «Удалить продажу».

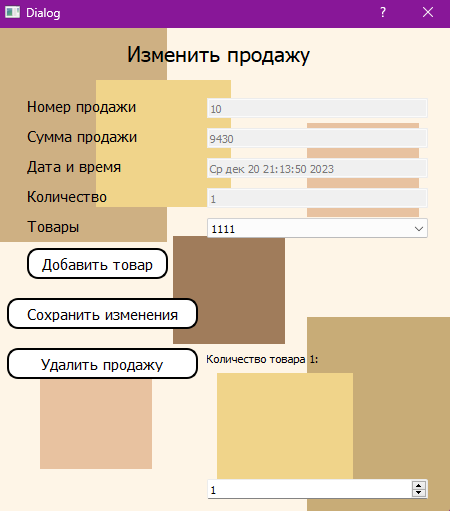


Рисунок 15 – диалоговое окно «Изменить продажу»

При нажатии на кнопку «Отчеты» открывается окно «Отчеты». В окне «Отчеты» также присутствуют четыре главные кнопки и кнопка «Сформировать отчет».

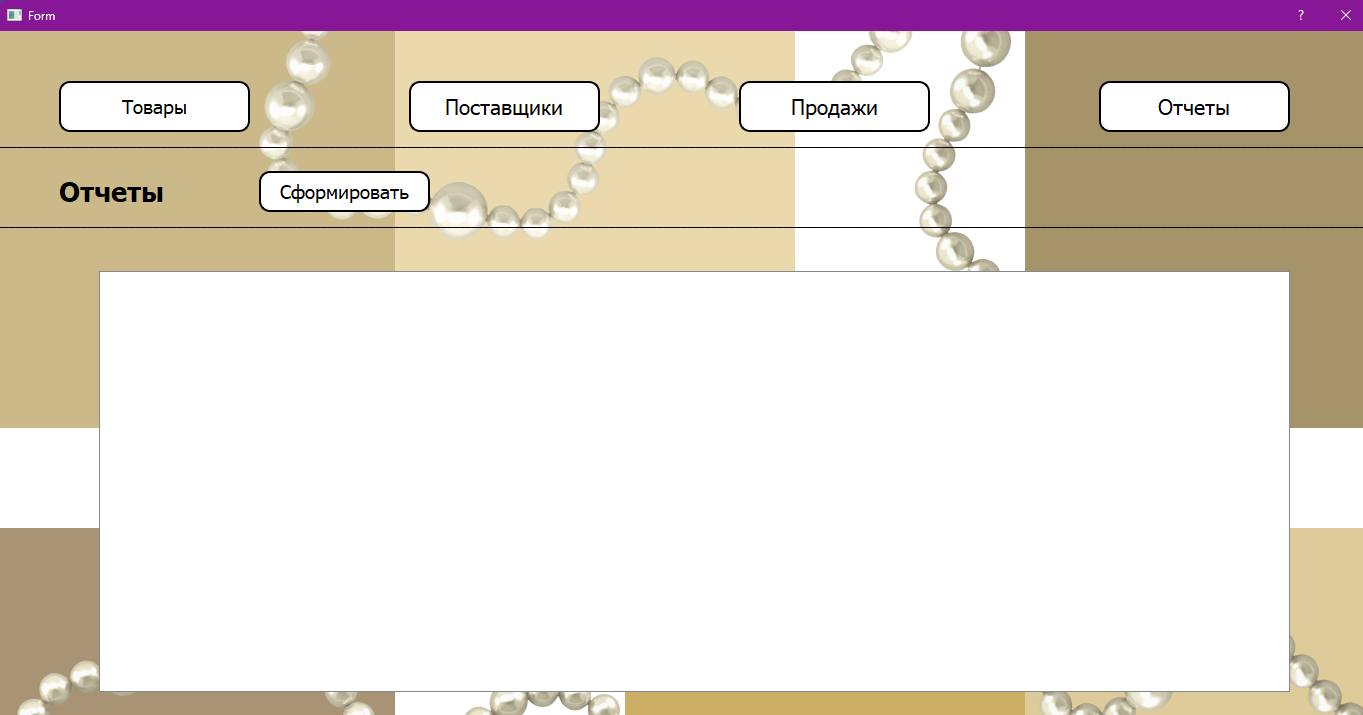


Рисунок 16 – окно «Отчеты»

При нажатии на кнопку «Сформировать отчет» открывается диалоговое окно, в котором мы можем выбрать: период времени продаж, категорию товаров и вид диаграммы.

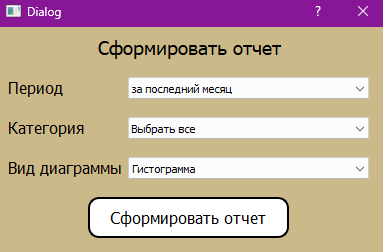


Рисунок 17 – диалоговое окно «Сформировать отчет»

После выбора критериев при нажатии на кнопку «Сформировать отчет», в окне «Отчеты» у нас формируется отчет по продажам по выбранным критериям.

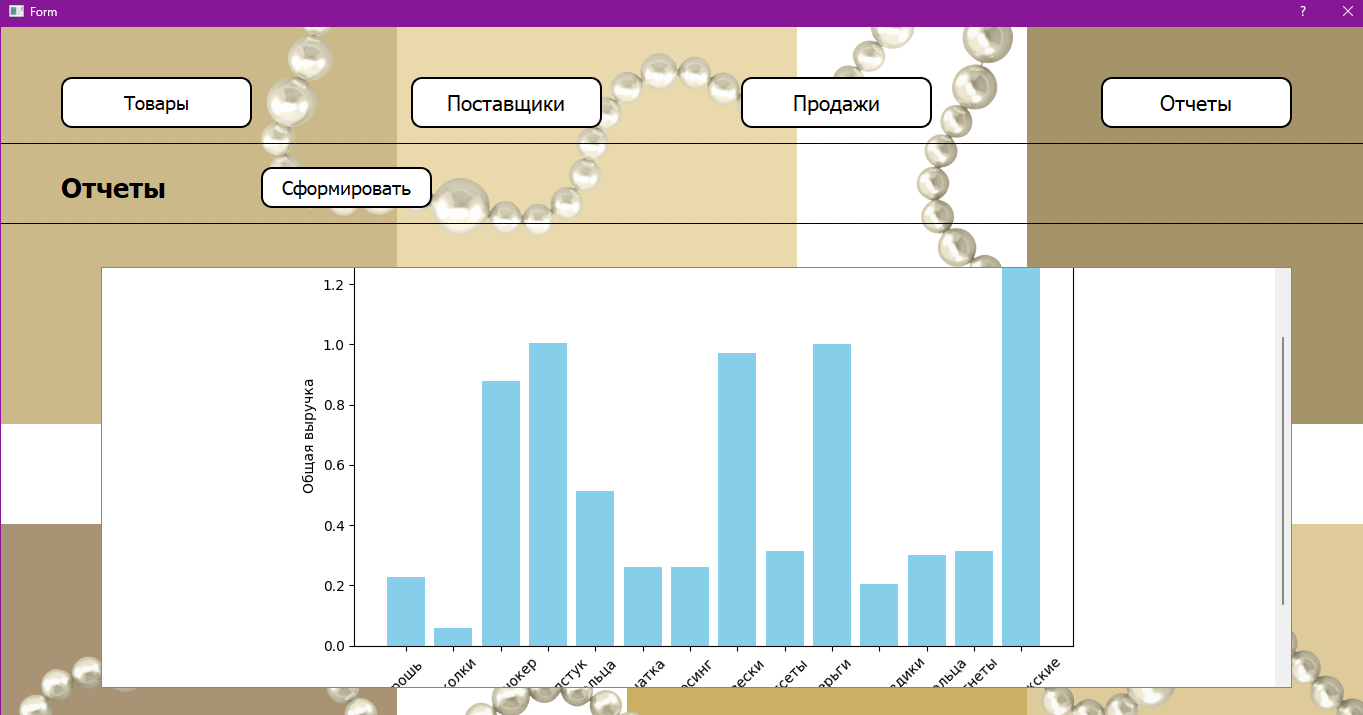


Рисунок 18– окно «Отчеты» со сформированным отчетам

1. Определение языка разработки

Для создания функционала информационной системы будет использован язык программирования Python. Этот язык — высокоуровневый, интерпретируемый, поддерживающий объектно-ориентированный подход и имеющий широкий набор библиотек, что делает его идеальным для разработки многофункциональных приложений.

Основные особенности Python:

* **универсальность и гибкость** – Python подходит для разработки различных приложений, от веб-сервисов до аналитических систем и автоматизации бизнес-процессов;
* **простота синтаксиса** – лаконичный синтаксис Python позволяет быстрее разрабатывать и поддерживать код, снижая вероятность ошибок и упрощая восприятие для новых разработчиков;
* **поддержка библиотек и фреймворков** – Python предоставляет большое количество библиотек для различных задач, включая Django и Flask для веб-приложений, а также инструменты для работы с данными, такие как pandas и NumPy;
* **встроенные возможности работы с данными** – Python содержит инструменты для обработки данных, анализа и визуализации, что упрощает ведение учета и отчетности в информационной системе.

Преимущества разработки на Python:

* **широкая поддержка сообщества** – наличие активного сообщества разработчиков и обширной документации облегчает процесс разработки и поддержки;
* **множество библиотек для анализа данных** – такие библиотеки, как pandas и NumPy, ускоряют процесс обработки данных, что полезно для ведения складского учета и аналитики;
* **гибкость интеграции** – Python легко интегрируется с другими языками и системами, что позволяет подключить систему к дополнительным модулям, если потребуется.

Использование Python обеспечит высокую скорость разработки и удобство поддержки, а также гибкость в расширении функционала информационной системы.

1. Определение системы управления базами данных

Для хранения информации о товарах, поставщиках и продажах будет использоваться система управления базами данных SQLite, управляемая через SQLiteStudio. SQLite — это легковесная встраиваемая СУБД, не требующая серверной инфраструктуры и подходящая для небольших и средних по объему данных приложений.

Основные особенности SQLiteStudio:

* **встраиваемая база данных** – SQLite не требует серверной настройки, а все данные хранятся в одном файле, что упрощает развертывание;
* **поддержка стандартов SQL** – система поддерживает стандартный SQL-синтаксис, что облегчает выполнение запросов и управление структурой базы;
* **удобный графический интерфейс** – SQLiteStudio предоставляет интерфейс для работы с базой, управления таблицами и создания запросов, что упрощает управление данными;
* **легковесность и быстродействие** – SQLite требует минимум ресурсов и обеспечивает высокую производительность при небольших объемах данных.

Преимущества использования SQLiteStudio:

* **простота установки и использования** – отсутствие серверной части делает SQLite удобной для быстрого разворачивания;
* **легкость резервного копирования и восстановления данных** – данные хранятся в одном файле, что упрощает создание резервных копий;
* **интеграция с языками разработки** – SQLite легко интегрируется с Python, что позволяет расширять функционал информационной системы без значительных затрат.

SQLiteStudio и SQLite обеспечат простоту управления данными и высокую производительность для системы учета ювелирного магазина.

1. Определение библиотеки для создания пользовательского интерфейса

Для разработки графического интерфейса информационной системы будет использоваться библиотека PyQt5. PyQt5 предоставляет набор инструментов для создания кросс-платформенных интерфейсов на базе фреймворка Qt, что позволяет создать современный и интуитивно понятный дизайн.

Основные особенности PyQt5:

* **обширный набор виджетов** – библиотека поддерживает множество виджетов, таких как кнопки, таблицы, поля ввода, что упрощает создание удобного интерфейса;
* **гибкость и настраиваемость** – PyQt5 позволяет детально настраивать элементы интерфейса, включая стилизацию с помощью CSS, что важно для создания привлекательного дизайна;
* **кросс-платформенность** – приложения на PyQt5 работают на Windows, macOS и Linux, что облегчает их использование на разных устройствах;
* **интеграция с Python** – PyQt5 легко взаимодействует с Python-кодом, что обеспечивает удобную связку между интерфейсом и бизнес-логикой приложения.

Преимущества использования PyQt5:

* **визуальное проектирование интерфейсов** – поддержка Qt Designer позволяет создавать интерфейс в визуальном редакторе, ускоряя процесс разработки;
* **широкая поддержка сообщества** – PyQt5 имеет большую базу ресурсов и документации, что делает его использование удобным и понятным;
* **поддержка сложных элементов интерфейса** – библиотека позволяет реализовать продвинутые интерфейсы с вкладками, таблицами и мультимедийными элементами.

Использование PyQt5 позволит создать привлекательный и функциональный интерфейс для системы ювелирного магазина, который будет удобен в использовании и поддержке.