

**КУРСОВОЙ ПРОЕКТ**

**по дисциплине: МДК 01.01 Разработка программных модулей**

**Тема: Разработка программного модуля**

**информационной системы «Обработка заказов»**

**Специальность 09.02.07 «Информационные системы и программирование»**

|  |  |  |
| --- | --- | --- |
| **Выполнил студент(ка) группы 311ИС-23** | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **А.М. Полежаев** |
| **Руководитель** | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Н.И. Кручинкина** |

**Москва 2024**



**УТВЕРЖДАЮ**

**Зам. директора КМПО**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_С.Ф. Гасанов**

**«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.**

**ЗАДАНИЕ НА КУРСОВОЙ ПРОЕКТ**

**по дисциплине: МДК 01.01 Разработка программных модулей**

**Специальность 09.02.07 «Информационные системы и программирование»**

**Студент группы 311ИС-23 Андрей Михайлович**

**ТЕМА: Разработка программного модуля информационной**

**системы «Обработка заказов»**

Дата выдачи задания «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 г.

Срок сдачи проекта «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2024 г.

**Москва 2024**

**Перечень вопросов, подлежащих разработке:**

ВВЕДЕНИЕ

1. Описание предметной области
   1. Введение в предметную область
   2. Анализ готовых решений
2. ПРОЕКТИРОВАНИЕ СИСТЕМЫ
   1. Общее назначение программы
   2. Требования к функциям, выполняемым системой
   3. План тестирования
3. РЕАЛИЗАЦИЯ ПРОЕКТНОЙ СИСТЕМЫ
   1. Описание среды разработки
      1. Выбор и описание программных инструментов
      2. Обоснование выбора инструментария по разработке
   2. Разработка программного модуля
      1. Реализация пользовательского интерфейса программы
      2. Описание кодом функциональных узлов модуля
      3. Результат работы и тестирования

ЗАКЛЮЧЕНИЕ

**Исходные данные:** 27.02.2006 №152-ФЗ, 27.02.2006 №149-ФЗ, 26.07.2017 №187-ФЗ, ГОСТ Р ИСО/МЭК 12207, ГОСТ 19.005–85, ГОСТ 19.105–78

**Перечень разрабатываемых материалов для визуализации:** ER-диаграмма базы данных, диаграмма прецедентов, архитектура программного модуля, экранные формы интерфейса, результаты тестирования.

Задание выдал:

Руководитель курсового проекта Н.И. Кручинкина

Задание принял к исполнению А.М. Полежаев

*Рассмотрено*

на заседании предметно-цикловой комиссии

информационных технологий и системного

администрирования

Протокол № \_\_\_от «\_\_» \_\_\_\_\_\_\_\_\_ 2024 г.

Председатель ПЦК \_\_\_\_\_\_\_\_\_\_\_\_\_

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 5](#_Toc185531590)

[**1.** **АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ** 7](#_Toc185531591)

[1.2. Обзор и анализ существующих программных решений 7](#_Toc185531592)

[1.3. Постановка задачи. Структура входной и выходной информации 8](#_Toc185531593)

[1.3.1. Постановка задачи 8](#_Toc185531594)

[1.3.2. Структура входной информации 8](#_Toc185531595)

[1.3.3. Структура выходной информации 9](#_Toc185531596)

[**2.** **ПРОЕКТИРОВАНИЯ И РАЗРАБОТКА ПРОГРАММЫ** 10](#_Toc185531597)

[2.1. Характеристики системы 10](#_Toc185531598)

[2.2. Функциональные требования 10](#_Toc185531599)

[2.3. Разработка сценария проекта 11](#_Toc185531600)

[2.4. Основные объекты и их описание 16](#_Toc185531601)

[3.3. План тестирования 24](#_Toc185531602)

[**ЗАКЛЮЧЕНИЕ** 26](#_Toc185531603)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 27](#_Toc185531604)

[**ПРИЛОЖЕНИЕ** 29](#_Toc185531605)

**ВВЕДЕНИЕ**

Курсовой проект посвящен созданию программного модуля для системы «Обработки заказов». В условиях современного рынка электронной коммерции и стремительного прогресса технологий автоматизации процессов обработки заказов приобретает критическое значение для успешной деятельности бизнеса. Это особенно актуально для организаций, которые обрабатывают значительные объемы заказов и требуют эффективных методов управления ими для обеспечения удовлетворенности клиентов.

Актуальность темы «Обработка заказов» представляет собой важный этап в цепочке поставок и взаимодействии с клиентами. Эффективная система обработки заказов, снижение количества ошибок и увеличение уровня удовлетворенности клиентов. В условиях высокой конкурентоспособности на рынке компании стремятся оптимизировать свои процессы для поддержания конкурентных приемушеств.

Цель проекта заключается в создании «Обработки заказов», который обеспечит эффективное управление заказами и упростит взаимодействие всех участников процесса.

Объектом исследования в курсовом проекте является деятельность компании, специализирующейся на обработке заказов. Предметом исследования является процесс учета и обработки заказов в организации.

1. Организация и контроль процессов обработки заказов.
2. Необходимость ведения учета данных о клиентах, заказах и товарах.
3. Проблемы взаимодействия между клиентами и администраторами системы.
4. Формирование отчетности по заказам компании.

Решение этих задач позволит улучшить процесс обработки заказов, что, в свою очередь, повысит общую эффективность работы компании.

Таким образом, разработка программного модуля для системы «Обработка заказов» имеет практическую значимость и будет способствовать улучшению организации работы с заказами в компаниях, работающих в сфере электронной коммерции.

1. **АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ**
   1. Описание проекта

Обработка заказов – это процесс, который включает в себя прием, управление, обработку и выполнение заказов от клиентов. Это ключевая функция в различных сферах бизнеса, таких как розничная торговля электронная коммерция, логистика, производство и другие области, где требуется взаимодействие с клиентами для предоставления товаров или услуг.

Обработка заказов является критическим важным процессом для любого бизнеса, так как она напрямую влияет на удовлетворенность клиент, лояльность и прибыльность компании.

Таким образов, обработка заказов – это комплексный процесс, который требует четкой организации, использования современных технологий и внимательного отношения к потребностям клиентов.

* 1. Обзор и анализ существующих программных решений

Обработка заказов является важным процессом для бизнеса, особенно в сферах электронной коммерции, розничной торговли, логистики и производства. Для автоматизации этого процесса разработано множество программных приложений, которые помогают упростить управление заказами, повысить эффективность работы и улучшить взаимодействие с клиентами. Проведенный анализ выявляет следующие ключевые платформы:

Shopify – это платформа для создания интернет-магазинов, которая включает в себя инструменты для приема и обработки заказов.

Orderry – это специализированное приложение для управления заказами, ориентированное на малый и средний бизнес.

RetailOps – это мощное решение для управление заказами, ориентированное на крупные интернет-магазины.

Анализ демонстрирует разнообразие решений для разных масштабов бизнеса, важность функциональности и интеграции, влияние стоимости на выбор решения и простоту использования.

* 1. Постановка задачи. Структура входной и выходной информации
     1. Постановка задачи

Обработка заказов представляет собой систему управления заказами, которая позволяет пользователям регистрироваться, входить в систему, просматривать товары, добавлять их в корзину, оформлять заказы, а также пополнять баланс. Администратор системы имеет дополнительные возможности, такие как управление товарами, просмотр списка пользователей, заказов, а также просмотр статистики продаж.

Основные задачи проекта:

* управления товарами (администратор): добавление новых товаров, удаление товаров, просмотр списка товаров;
* управление пользователями (администратор): просмотр списка пользователей, удаление пользователей;
* управление заказами (администратор): просмотр списка всех заказов и формирования их;
* корзина и оформления заказа (пользователь): добавление товаров в корзину, удаление товаров из корзины, оформление заказа с проверкой баланса;
* пополнение баланса (пользователь): пополнение баланса пользователя;
* статистика (администратор): просмотр топ-5 товаров по продажам.
  + 1. Структура входной информации
* пользователи: логин и пароль для регистрации и входа, баланс пользователя для совершения покупок;
* товары: название, цена, количество товара, ID товара для идентификации;
* заказы: ID пользователя, ID товара, количество товара, дата создания заказа;
* администратор: уникальный логин «admin» для доступа к администраторской панели.
  + 1. Структура выходной информации
* пользователь: список доступных товаров, корзина с выбранными товарами, история заказов, обновленный баланс после покупок или пополнения;
* администратор: список всех пользователей, список всех товаров, список всех заказов, статистика продаж.

1. **ПРОЕКТИРОВАНИЯ И РАЗРАБОТКА ПРОГРАММЫ**
   1. Характеристики системы

Для обеспечения эффективной работы программы, а также ее безопасности и производительности, необходимо учитывать несколько ключевых характеристик.

К ним относятся:

* + 1. Производительность: производительность программы определяет, насколько быстро и эффективно она выполняет свои функции. В данном случае, программа должна быстро обрабатывать запросы пользователей, обновлять базу данных и отображать информацию;
    2. Безопасность: безопасность программы включает защиту данных пользователей, предотвращение несанкционированного доступа и обеспечение целостности данных;
    3. Масштабируемость: программа должна быть способна работать с большим количеством пользователей и данных без значительного снижение производительности;
    4. Удобство использования: программа должна быть удобной для пользователей, чтобы они могли легко выполнять свои задачи;
    5. Надежность: программа должна быть стабильной и корректно обрабатывать ошибки, чтобы избежать сбоев;
    6. Совместимость: программа должна работать на различных платформах и с разными версиями библиотек;
    7. Документирование: программа должна быть хорошо документирована, чтобы другие разработчики могли легко разобраться в её коде.
  1. Функциональные требования

Модуль для «Обработки заказов» должен обеспечивать выставление товаров, покупку товаров, корзину.

Функциональные требования включает в себя следующее:

* систему аутентификации
* управление пользователями
* управление товарами (только для администратора)
* просмотр заказов (только для администратора)
* покупка товаров (для обычных пользователей)
* корзина пользователя
* пополнение баланса (для обычных пользователей)
* история заказов (для обычных пользователей)
* поиск товаров
* статистика (только для администратора)
* фильтрация товаров
* интерфейс пользователя
* обработка ошибок
* выход из системы
  1. Разработка сценария проекта

Проектирование сценариев проекта – это важные этап, который помогает точно определить, как будущая система будет работать с людьми и какие значения она будет решать. Эти сценарии позволяют представить себе, как система будет использоваться и служат основой для дальнейшей разработки.

Сценарий 1: Регистрация нового пользователя

Участники: Пользователь.

Цель: Зарегистрировать нового пользователя в системе.

Сценарий:

* пользователь нажимает кнопку "Регистрация" на экране входа;
* пользователь вводит логин и пароль в соответствующие поля;
* пользователь нажимает кнопку "Зарегистрироваться";
* система проверяет, существует ли уже пользователь с таким логином;
* если логин уникален, пользователь добавляется в базу данных, и система отображает сообщение: "Регистрация прошла успешно";
* пользователь возвращается на экран входа;
* если логин уже существует, система отображает сообщение: "Пользователь с таким логином уже существует".

Сценарий 2: Вход в систему

Участники: Пользователь.

Цель: Войти в систему под своей учетной записью.

Сценарий:

* пользователь вводит логин и пароль в соответствующие поля;
* пользователь нажимает кнопку "Войти";
* система проверяет, существует ли пользователь с таким логином и паролем;
* если данные корректны: если пользователь — администратор, открывается админ-панель, если пользователь — обычный пользователь, открывается интерфейс пользователя;
* если данные некорректны, система отображает сообщение: "Неверный логин или пароль".

Сценарий 3: Добавление товара (администратор)

Участники: Администратор.

Цель: Добавить новый товар в систему.

Сценарий:

* администратор нажимает кнопку "Добавить товар" в админ-панели.
* открывается новое окно с полями для ввода:
* название товара.
* цена товара.
* количество товара.
* администратор заполняет поля и нажимает кнопку "Добавить".
* система добавляет товар в базу данных и закрывает окно.

Сценарий 4: Просмотр списка товаров (пользователь)

Участники: Пользователь.

Цель: Просмотреть список доступных товаров.

Сценарий:

* пользователь открывает интерфейс пользователя;
* система загружает список товаров из базы данных и отображает их в Listbox;
* пользователь видит информацию о каждом товаре: ID товара, название, цена, количество на складе.

Сценарий 5: Добавление товара в корзину (пользователь)

Участники: Пользователь.

Цель: Добавить товар в корзину.

Сценарий:

* пользователь нажимает кнопку "Добавить в корзину".
* открывается новое окно с полями для ввода: ID товара, количество;
* пользователь вводит данные и нажимает кнопку "Добавить";
* система проверяет, есть ли товар на складе в указанном количестве;
* если товар доступен, он добавляется в корзину, и система отображает сообщение: "Товар добавлен в корзину";
* если товара недостаточно, система отображает сообщение: "Недостаточно товара на складе".

Сценарий 6: Просмотр корзины (пользователь)

Участники: Пользователь.

Цель: Просмотреть содержимое корзины.

Сценарий:

* пользователь нажимает кнопку "Корзина";
* открывается новое окно, в котором отображается список товаров в корзине;
* пользователь видит: ID товара, название, цена, количество;
* пользователь может нажать кнопку "Оплатить", чтобы оплатить товары, или "Удалить товар", чтобы удалить товар из корзины.

7. Оплата товаров (пользователь)

Участники: Пользователь.

Цель: Оплатить товары в корзине.

Сценарий:

* пользователь нажимает кнопку "Оплатить" в окне корзины;
* система проверяет, достаточно ли средств на балансе пользователя;
* если средств достаточно:

Система вычитает стоимость товаров из баланса пользователя.

Система уменьшает количество товаров на складе.

Система добавляет заказ в базу данных.

* система очищает корзину и отображает сообщение: "Оплата прошла успешно";
* если средств недостаточно, система отображает сообщение: "Недостаточно средств на балансе".

8. Пополнение баланса (пользователь)

Участники: Пользователь.

Цель: Пополнить баланс пользователя.

Сценарий:

* пользователь нажимает кнопку "Пополнить баланс";
* открывается новое окно с полем для ввода суммы пополнения;
* пользователь вводит сумму и нажимает кнопку "Пополнить";
* система проверяет, что сумма положительная;
* если сумма корректна, система обновляет баланс пользователя и отображает сообщение: "Баланс пополнен на [сумма]".

9. Просмотр истории заказов (пользователь)

Участники: Пользователь.

Цель: Просмотреть историю своих заказов.

Сценарий:

* пользователь нажимает кнопку "История заказов";
* открывается новое окно, в котором отображается список заказов пользователя;
* пользователь видит информацию о каждом заказе: ID заказа, название товара, количество, дата и время заказа.

10. Поиск товара (пользователь)

Участники: Пользователь.

Цель: Найти товар по названию.

Сценарий:

* пользователь нажимает кнопку "Поиск товара";
* открывается новое окно с полем для ввода названия товара;
* пользователь вводит название и нажимает кнопку "Найти";
* система ищет товары, содержащие введенное название, и отображает результаты.

11. Просмотр статистики (администратор)

Участники: Администратор.

Цель: Просмотреть статистику по продажам.

Сценарий:

* администратор нажимает кнопку "Статистика" в админ-панели;
* открывается новое окно, в котором отображается график топ-5 товаров по продажам;
* администратор видит: название товара, количество продаж.

12. Удаление товара (администратор)

Участники: Администратор.

Цель: Удалить товар из системы.

Сценарий:

* администратор нажимает кнопку "Удалить товар" в админ-панели;
* открывается новое окно с полем для ввода ID товара;
* администратор вводит ID товара и нажимает кнопку "Удалить";
* система удаляет товар из базы данных.
  1. Основные объекты и их описание

В базе данных «orders.db» присутствуют данные таблицы:

Таблица 1 – Объекты предметной области

|  |  |  |
| --- | --- | --- |
| № | Заказы | Краткое описание |
| 1 | Заказы | Основной объект, представляющий заказы |
| 2 | Товары | Объект, представляющий товары в заказах |
| 3 | Пользователи | Пользователи с доступом к программе |
| 4 | Статусы заказов | Статусы заказов |

Таблица 2 – Сущность «orders»

|  |  |  |
| --- | --- | --- |
| Атрибуты | Тип | Описание |
| «order\_id» | INTEGER | Идентификатор заказа |
| «user\_id» | INTEGER | Идентификатор пользователя |
| «product\_id» | INTEGER | Идентификатор товара |
| «quantity» | INTEGER | Количество товара в заказе |
| «status» | VARCHAR(20) | Статус заказа |
| «created\_at» | DATETIME | Дата и время создания заказа |

Таблица 3 – Сущность «users»

|  |  |  |
| --- | --- | --- |
| Атрибуты | Тип | Описание |
| «id» | INTEGER PRIMARY KEY | Идентификатор пользователя |
| «username» | TEXT UNIQUE NOT NULL | Логин пользователя |
| «password» | TEXT NOT NULL | Пароль пользователя |
| «balance» | REAL DEFAULT 0 | Баланс пользователя |

Таблицы 4 – Сущность «products»

|  |  |  |
| --- | --- | --- |
| Атрибуты | Тип | Описание |
| «id» | INTEGER PRIMARY KEY | Идентификатор продукта |
| «name» | TEXT NOT NULL | Название продукта |
| «price» | REAL NOT NULL | Цена продукта |
| «quantity» | INTEGER NOT NULL | Количество продукта |

Связи в программе:

Связь «users» – «orders», тип связи: 1 – М.

Связь «orders» - «products», тип связи: 1 – М.

ERD модели

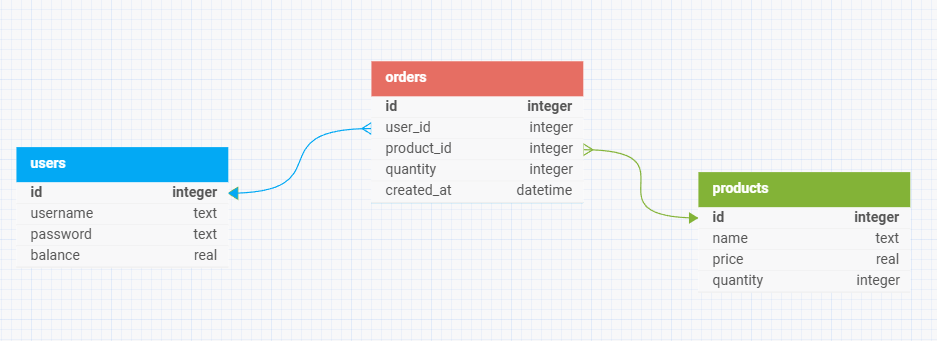


Рисунок 1 – IDEF1X-Диаграмма БД обработки заказов.

UseCase модель

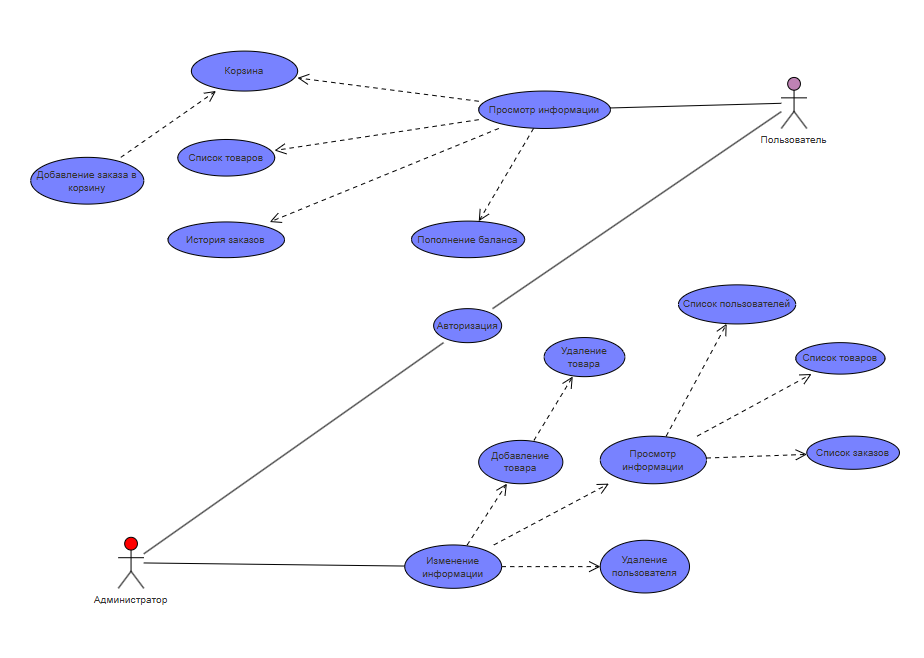


Рисунок 2 – UseCase базы данных Обработки заказов.

* 1. План тестирования
  + Регистрация и авторизация:

проверка регистрации, авторизации, ошибок ввода;

* + Функционал администратора:

добавление/удаление товаров и пользователей, просмотр заказов, статистики;

* + Функционал пользователя:

просмотр товаров, добавление в корзину, оплата, пополнение баланса, история заказов, поиск;

* + Фильтрация и сортировка товаров:

сортировка по цене и количеству;

* + Графический интерфейс:

проверка отображения элементов, стилей, графиков.

3. **ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ МОДУЛЯ**

3.1. Описание среды разработки

Для разработки программного модуля «Обработки заказов» были выбраны следующие инструменты:

Среда разработки:

PyCharm – мощная IDE для Python, предоставляющая интеллектуальную поддержку кода, встроенный отладчик и поддержку тестирования. Язык программирования: Python – универсальный язык с простым синтаксисом, кроссплатформенностью и обширной библиотекой.

Библиотеки:

* tkinter – для создания GUI.
* tkinter.ttk – для современных виджетов.
* Matplotlib – для создания графиков и диаграмм.
* atetime – для работы с датой и временем.

База данных:

SQLite3 – встраиваемая СУБД с поддержкой SQL, высокой производительностью и низким потреблением ресурсов.

Преимущества выбранного инструментария:

* PyCharm: обеспечивает интеллектуальную поддержку кода, автоматическое дополнение, встроенный отладчик для выявления ошибок и поддержку тестирования

(unittest, pytest, doctest).

* Python: простой и доступный язык с кроссплатформенностью, богатой встроенной библиотекой и огромным количеством внешних библиотек для различных задач.
* Tkinter: встроен в Python, прост в использовании, поддерживает кроссплатформенность и предоставляет гибкость для создания различных типов интерфейсов.
* SQLite3: встраиваемая СУБД без необходимости отдельного сервера,

поддерживает стандартные SQL-команды, обладает высокой производительностью и низким потреблением ресурсов.

3.2. Реализация пользовательского интерфейса приложения

Для удобности взаимодействия пользователя с программным модулем «Обработки заказа» был разработан пользовательский интерфейс. Реализация пользовательского интерфейса включает в себя размещение кнопок, отображение информации, диаграмм и обработку пользовательских действий.

При первичном запуске приложения пользователь попадает в окно авторизации, где он может зарегистрироваться если отсутствует его аккаунт. После ввода логина и пароля, пользователь попадает в главное окно с основной информацией и функционалом.

Интерфейс окна показан на рисунке 3.

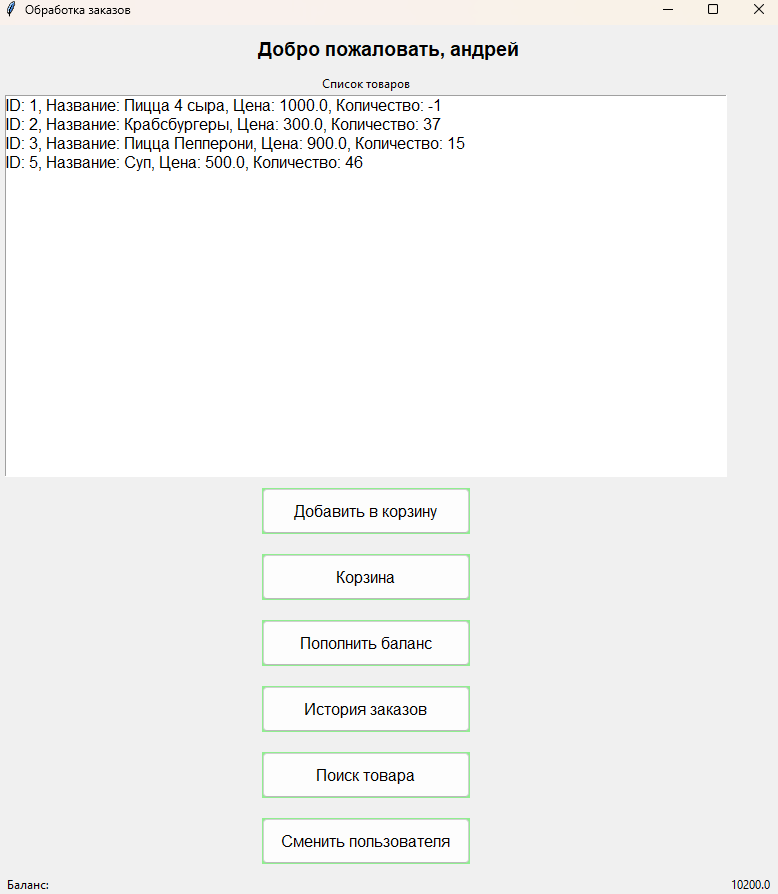


Рисунок 3 – Главное окно пользователя

У пользователя дан список товаров, которые он может приобрести, найти нужный товар он может через поиск товаров. Просмотреть, что он приобретал может через историю заказов, может пополнить баланс и оплатить товары в корзине.

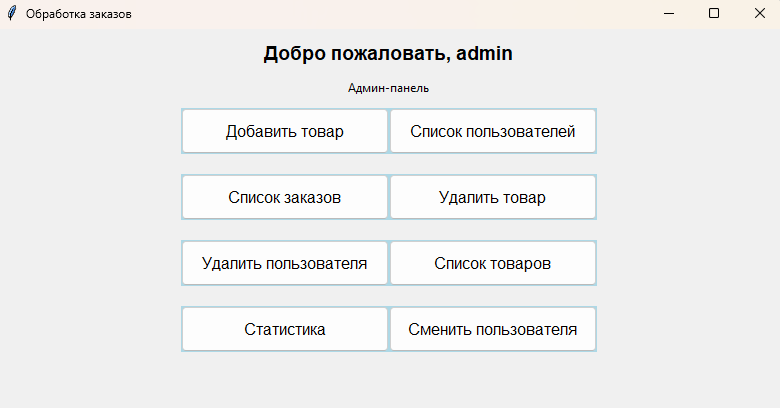


Рисунок 4 – Главное окно администратора

Администратору даны такие возможности, как: добавление заказа, удаление заказа, просмотр статистики за все проданные товары, просматривать список, в котором пользователи заказывали товары и возможностью удалить пользователя.

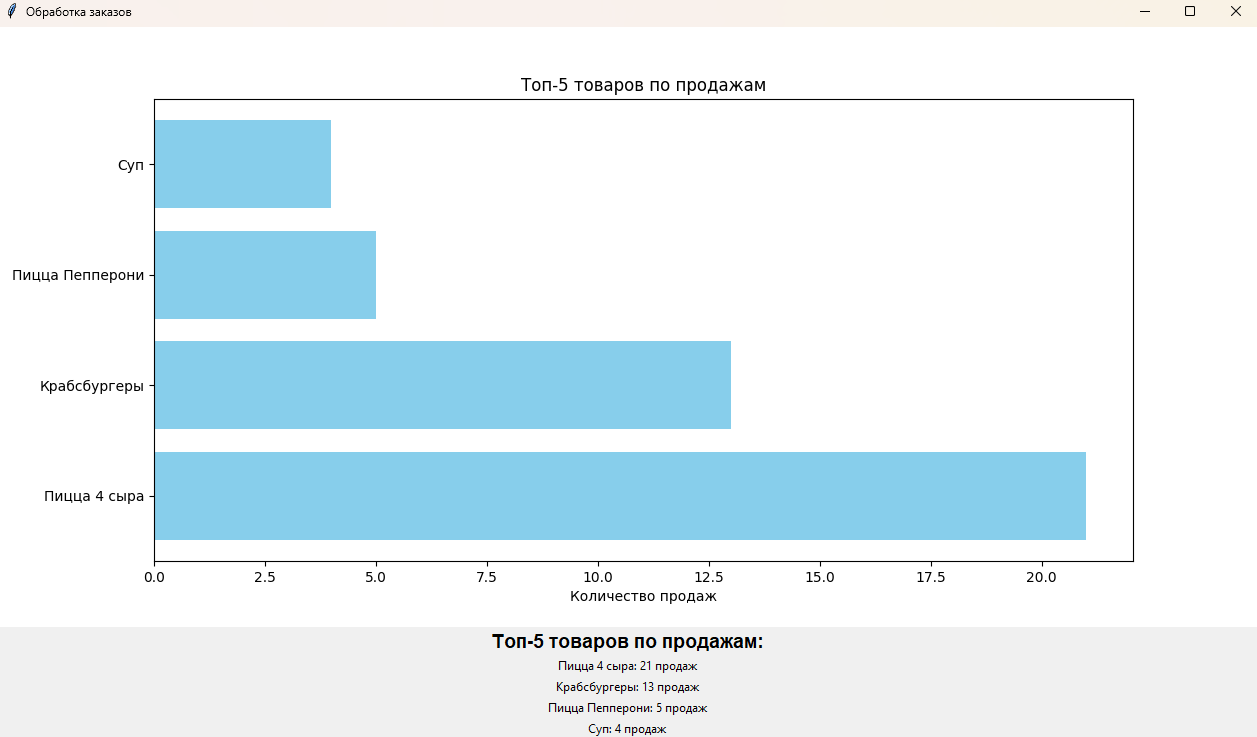


Рисунок 5 – Окно диаграммы.

Данное окно показывает топ-5 товаров, которые больше всего покупали.

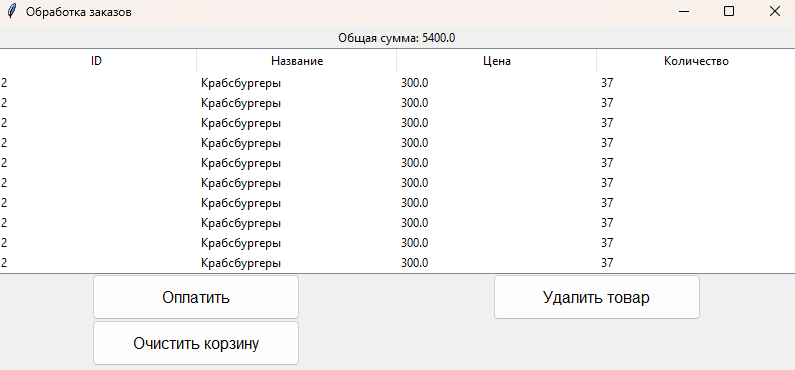


Рисунок 6 – Окно корзины.

Данное окно дает пользователю доступ к корзине, он может оплатить ее, удалить какой-то один товар и его количество и очистить корзину.

3.3. План тестирования

Тест план:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | **Описание теста (тип)** | **Предусловия** | **Шаги для воспроизведения** | **Ожидаемый результат** | **Фактический результат** |
| 1 | Создание нового заказа (позитивный) | Администратор авторизован | 1. Ввести данные заказа (товары, количество, цена).  2. Нажать кнопку "Создать заказ". | Заказ успешно создан, отображается в списке заказов | Заказ успешно создан, отображается в списке заказов. |
| 2 | Создание заказа с пустыми полями (негативный) | Пользователь авторизован | 1. Оставить все поля пустыми.  2. Нажать кнопку "Создать заказ". | Появляется сообщение об ошибке, заказ не создается. | Появляется сообщение об ошибке, заказ не создается. |
| 3 | Просмотр деталей заказа (позитивный) | Заказ создан | 1. Выбрать заказ из списка.  2. Просмотреть заказы | Отображаются все детали заказа (товары, количество, статус). | Отображаются все детали заказа (клиент, товары, количество, статус). |
| 4 | Изменение статуса заказа (позитивный) | Заказ создан | 1. Выбрать заказ из списка.  2. Изменить статус заказа.  3. Нажать кнопку "Сохранить". | Статус заказа успешно изменен. | Статус заказа успешно изменен. |
| 5 | Изменение статуса заказа на недопустимый (негативный) | Заказ создан | 1. Выбрать заказ из списка.  2. Ввести недопустимый статус.  3. Нажать кнопку "Сохранить". | Появляется сообщение об ошибке, статус не изменяется. | Появляется сообщение об ошибке, статус не изменяется. |
| 6 | Удаление заказа (позитивный) | Заказ создан | 1. Выбрать заказ из списка.  2. Нажать кнопку "Удалить заказ". | Заказ успешно удален из списка. | Заказ успешно удален из списка. |

**ЗАКЛЮЧЕНИЕ**

В ходе выполнения курсового проекта была разработана информационная система «Обработка заказов», которая предназначена для автоматизации процессов управления заказами в компаниях, работающих в сфере электронной коммерции. Основной целью проекта было создание программного модуля, который позволит эффективно управлять заказами, упростить взаимодействие между клиентами и администраторами системы, а также обеспечить формирование отчетности по заказам.

В первой главе был проведен анализ предметной области, где были рассмотрены ключевые аспекты обработки заказов, включая их важность для бизнеса, а также проведен обзор существующих программных решений.

Во второй главе были спроектированы и разработаны основные компоненты программы. Были определены характеристики системы, функциональные требования, разработаны сценарии использования, а также описаны основные объекты и их взаимосвязи.

В третьей главе была реализована практическая часть проекта. Была описана среда разработки, выбранные инструменты и библиотеки, такие как Python, Tkinter, SQLite3 и Matplotlib. Был разработан пользовательский интерфейс, который включает окна для регистрации, авторизации, управления товарами, заказами, корзиной и статистикой. Также были описаны основные методы и функции, реализованные в коде, которые обеспечивают работу системы.

В заключение можно отметить, что выполнение данного курсового проекта позволило не только закрепить теоретические знания, полученные в ходе изучения дисциплины «Разработка программных модулей», но и приобрести практические навыки разработки программного обеспечения. Результаты работы могут быть использованы в дальнейшей профессиональной деятельности, а также в рамках других проектов, связанных с разработкой информационных систем.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. GitHub ссылка на репозиторий - https://github.com/pimkaav/kursach
2. ГОСТ Р ИСО/МЭК 12207-2010 "Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств".
3. ГОСТ 19.105-78 "Единая система программной документации. Общие требования к программным документам".
4. ГОСТ 19.005-85 "Единая система программной документации. Требования к программным документам, выполненным печатным способом".
5. Python Documentation – Официальная документация Python. https://docs.python.org/3/.
6. Tkinter 8.5 reference: a GUI for Python – Официальная документация Tkinter. https://tkdocs.com/.
7. SQLite Documentation – Официальная документация SQLite. https://www.sqlite.org/docs.html.
8. Matplotlib Documentation – Официальная документация Matplotlib. https://matplotlib.org/stable/contents.html.
9. Flask Documentation – Официальная документация Flask (если используется в проекте). https://flask.palletsprojects.com/.
10. Django Documentation – Официальная документация Django (если используется в проекте). https://docs.djangoproject.com/.
11. Larman, C. "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development". – Prentice Hall, 2004.
12. Pressman, R.S. "Software Engineering: A Practitioner's Approach". – McGraw-Hill, 2014.
13. Sommerville, I. "Software Engineering". – Addison-Wesley, 2011.
14. Fowler, M. "UML Distilled: A Brief Guide to the Standard Object Modeling Language". – Addison-Wesley, 2003.
15. Севостьянов, В.В. "Разработка программного обеспечения: Учебное пособие". – М.: ИНФРА-М, 2012.
16. Бизли, Д. "Python. К вершинам мастерства". – М.: ДМК Пресс, 2016.
17. Лутц, М. "Изучаем Python". – СПб.: Символ-Плюс, 2011.
18. Васильев, А.Н. "Tkinter. Графический интерфейс пользователя в Python". – М.: ДМК Пресс, 2017.
19. Робинсон, С. "SQLite. Основы работы с СУБД". – М.: ДМК Пресс, 2015.
20. МакФарланд, Д. "Изучаем SQL". – СПб.: Символ-Плюс, 2011.
21. Корнеев, Г.Г. "Базы данных: Учебное пособие". – М.: КноРус, 2013.

**ПРИЛОЖЕНИЕ**

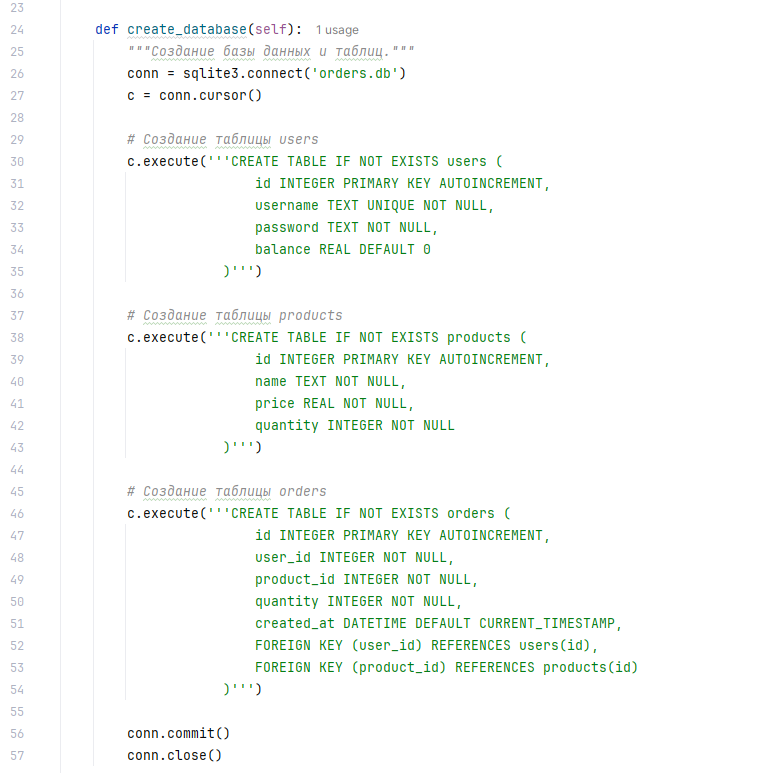


Рисунок 6 – Инициализация базы данных



Рисунок 7 – Код окна авторизации



Рисунок 8 – Код окна регистрации и логина



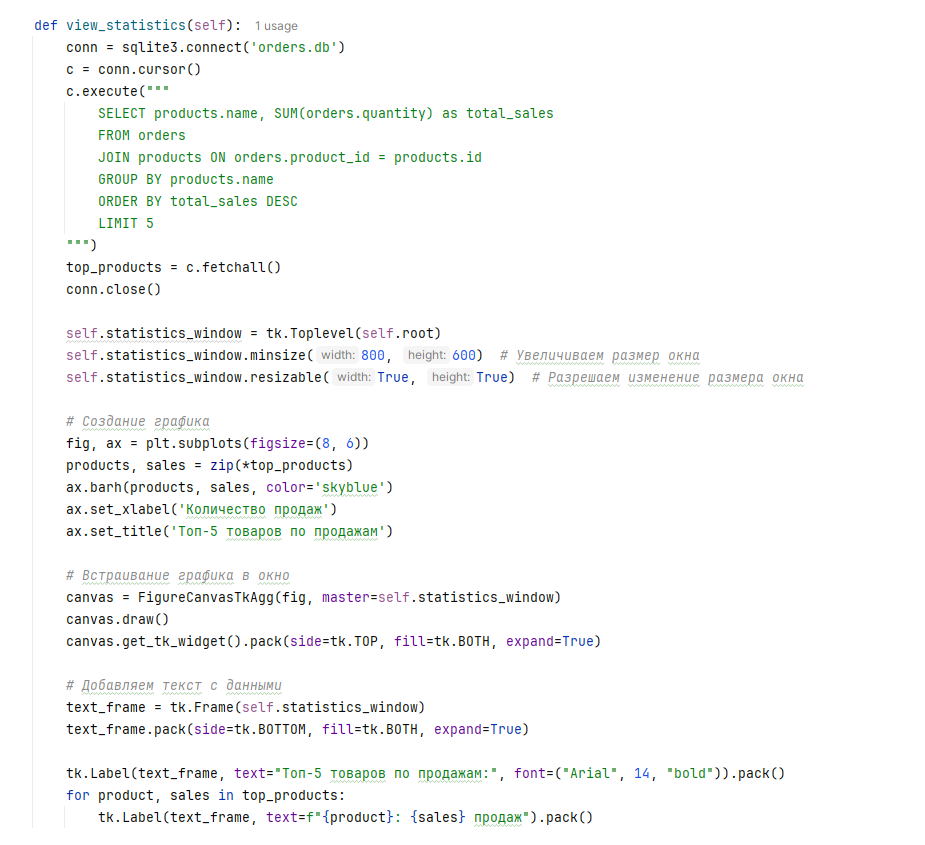


Рисунок 9 – Код главного окна администратора





Рисунок 10 – Код главного окна пользователя



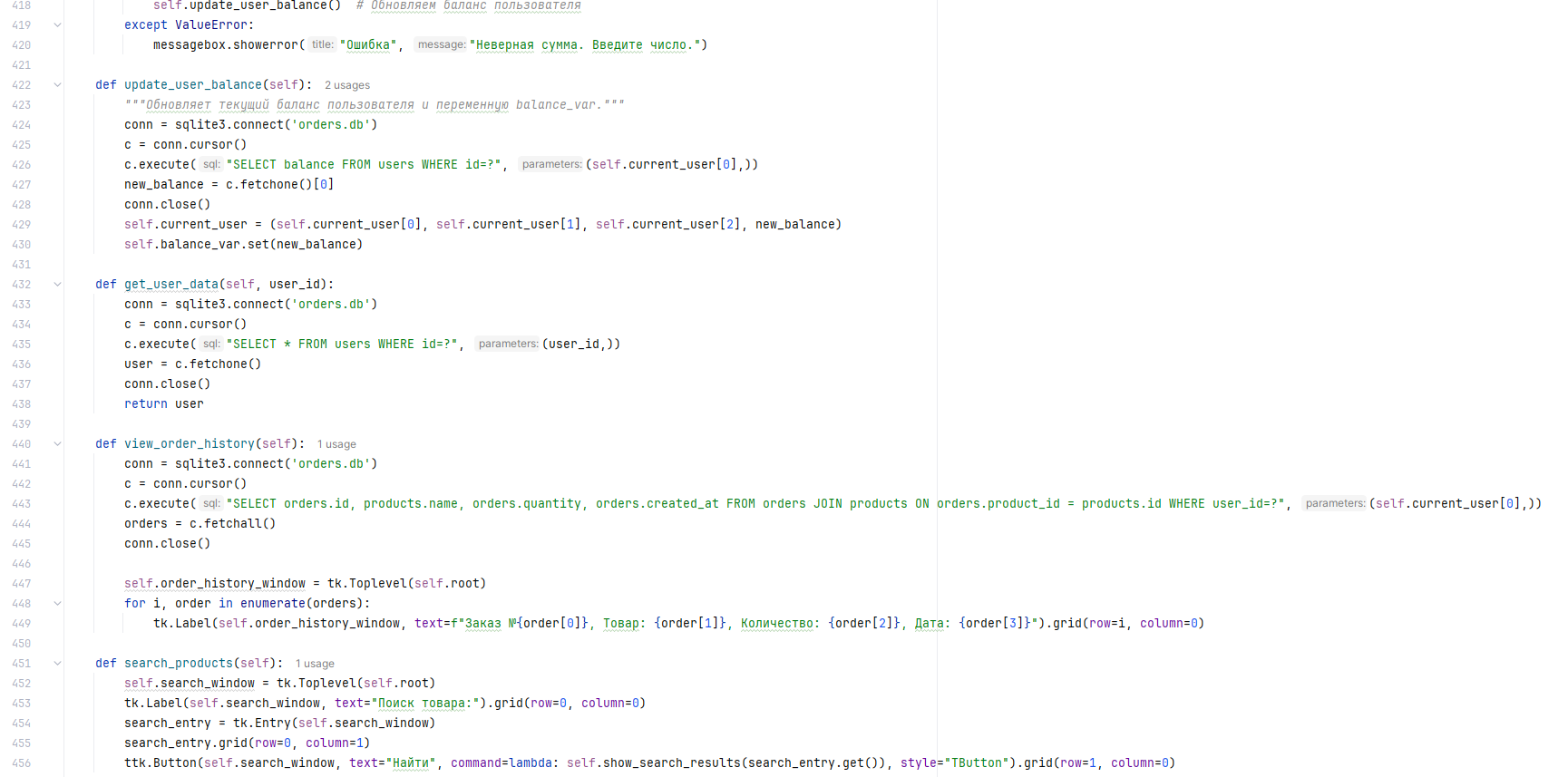


Рисунок 11 – Функции в коде.