**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ 3**](#_Toc184425559)

[**1. Описание предметной области 5**](#_Toc184425560)

[1.1. Введение в предметную область 5](#_Toc184425561)

[1.2. Анализ готовых решений 8](#_Toc184425562)

[**2. Проектирование системы 12**](#_Toc184425563)

[2.1. Общее назначение системы 12](#_Toc184425564)

[2.2. Требование к функциям выполняемых системой 12](#_Toc184425565)

[2.3. План тестирования 13](#_Toc184425566)

[**3. Реализация проекта системы 15**](#_Toc184425567)

[3.1. Описание среды разработки 15](#_Toc184425568)

[3.1.1. Выбор и описание программных инструментов 15](#_Toc184425569)

[3.1.2. Обоснование выбора инструментария по разработке 16](#_Toc184425570)

[3.2. Разработка программного модуля 17](#_Toc184425571)

[3.2.1. Реализация пользовательского интерфейса программы 17](#_Toc184425572)

[3.2.2. Описание кодом функциональных узлов модуля 21](#_Toc184425573)

[3.2.3. Результат работы и тестирования 22](#_Toc184425574)

[**ЗАКЛЮЧЕНИЕ 24**](#_Toc184425575)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 25**](#_Toc184425576)

[**ПРИЛОЖЕНИЕ 1 27**](#_Toc184425577)

[**ПРИЛОЖЕНИЕ 2 31**](#_Toc184425578)

[**ПРИЛОЖЕНИЕ 3 38**](#_Toc184425579)

# **ВВЕДЕНИЕ**

Складской учет — это хранение и обработка данных по товарам, сырью, финансовым ресурсам. Здесь не учитываются незавершённое производство, активы и основные средства. Он является ключевым элементом логистики, обеспечивающим контроль над запасами, движением товаров и минимизацией рисков. [1]

Работа склада — это масса рутинных операций: приём, отгрузка, перемещение, списание и множество других действий. Ускорить их выполнение можно с помощью автоматизации.

Целью данного курсового проекта является разработка автоматизированной системы учета товаров на складе, которая обеспечит упрощение процессов складского учета, позволит оперативно получать актуальные данные и тем самым повысит эффективность управления складом. В рамках разработки особое внимание уделяется пользовательскому интерфейсу и гибкости системы. Система ориентирована на удобство взаимодействия пользователей различных ролей, включая администраторов, сотрудников и клиентов.

В задачи курсового проекта входят:

1. Проанализировать предметную область складского учета, изучить процессы, которые происходят на складе, и потребности пользователей системы.
2. Проанализировать существующие программные решения для автоматизации складского учета с целью выявления их преимуществ и недостатков.
3. Подготовить техническое задание, в котором будут подробно описаны функциональные требования к системе.
4. Разработать план тестирования для проверки функциональности системы, в том числе тесты на корректность учета товаров, работу с базой данных и взаимодействие через пользовательский интерфейс.
5. Обоснование выбора инструментов и средств разработки.
6. Описать процесс реализации технического задания, включая создание пользовательского интерфейса, настройку базы данных и разработку механизмов взаимодействия между интерфейсом и базой данных.
7. Выполнить тестирование разработанной системы на основе подготовленного плана тестирования, выявить и исправить возможные ошибки.

Объектом исследования является процесс автоматизации учета товаров на складе. Учёт товаров на складе включает в себя управление запасами, регистрацию поступления и выбытия товаров, а также оптимизацию процессов контроля за движением товарных единиц.

Предметом исследования выступает процессы управление запасами, регистрацию поступления и выбытия товаров обеспечения для автоматизации учета на складе.

Актуальность разработки системы автоматизированного учета товаров на складе связана с необходимостью повышения эффективности управления складскими операциями. В современных условиях точное и своевременное управление запасами является критически важным. Система автоматизации также снижает затраты на обработку информации и обеспечивает надежное хранение данных, что минимизирует влияние человеческого фактора.

Таким образом, автоматизация учета товаров на складе способствует повышению эффективности работы склада, за счет уменьшения времени на обработку информации и исключения ошибок. Автоматизированная система учета делает складские процессы более прозрачными и контролируемыми, что особенно важно в современных условиях.

# [**Описание предметной**](#содержание) **области**

* 1. **Введение в предметную область**

Складской учет представляет собой важную составляющую процесса управления материальными запасами в любом предприятии. Он включает в себя контроль над приемкой, хранением, перемещением и отпуском товаров, а также поддержание информации о текущем состоянии запасов. Складской учет необходим для обеспечения точности данных о наличии продукции, её передвижении и остатках, что напрямую влияет на рентабельность и эффективность предприятия. [15]

Склад, как объект, является центром логистики, где происходит физическое хранение и перемещение товаров. Управление такими процессами вручную требует больших затрат ресурсов, что увеличивает вероятность возникновения ошибок. Для оптимизации этих процессов широко используются системы автоматизированного учета. Такие системы позволяют не только сократить влияние человеческого фактора, но и ускорить обработку данных, обеспечивая точность в реальном времени.

Системы автоматизации склада представляют собой программные решения, предназначенные для регистрации всех операций, связанных с товарами: поступление, выбытие, перемещение и инвентаризация. В условиях автоматизации бизнес-процессов системы учета товаров на складе являются незаменимыми инструментами для современного предприятия, поскольку они позволяют избежать ошибок, снизить издержки, связанные с избыточным или недостаточным запасом, и улучшить взаимодействие с другими подразделениями компании, такими как отдел закупок или продажи.

Современные подходы к автоматизации складского учета предполагают использование информационных систем, которые включают комплекс программных решений для регистрации, хранения и обработки данных о товарных запасах. Основные цели таких систем — повышение точности учета, снижение затрат и улучшение контроля над логистикой. Теоретическая база включает принципы системного подхода к управлению данными, где информационные потоки интегрируются в единую систему для оптимизации процессов.

Методы разработки ИС включают адаптацию системы под специфические нужды компании, управление складом в режиме реального времени, интеграцию с другими бизнес-процессами и использование реляционных баз данных. Эти подходы позволяют автоматизировать ключевые складские операции: приемку, хранение, инвентаризацию и отгрузку товаров.

Автоматизированная система учета, разрабатываемая в рамках данного курсового проекта, призвана обеспечить удобный интерфейс для работы с товарами на складе. Использование языка программирования Python в связке с библиотекой PyQt6 и базой данных на основе SQL позволит создать доступную, гибкую и надежную систему, способную решать задачи управления запасами на складе.

Современные системы складского учета должны соблюдать следующие требования:

* + - Модульность и масштабируемость
    - Интеграция с другими системами
    - Простота в использовании и интерфейс.
    - Безопасность данных
    - Поддержка работы в реальном времени

Для автоматизации учета товаров на складе необходимо учитывать следующие законодательные и нормативные акты, которые регулируют использование информационных систем и вопросы защиты данных в Российской Федерации:

* **«Конституция Российской Федерации» (принята всенародным голосованием 12.12.1993 с изменениями, одобренными в ходе общероссийского голосования (01.07.2020)**  
  Конституция РФ является основным законом, устанавливающим основы государственного устройства, права и свободы граждан. Для разработки системы складского учёта она важна как источник понимания прав граждан на защиту данных, свободное предпринимательство и регулирование отношений между организациями и государством. [1]
* ФЗ «О персональных данных» № 152-ФЗ от 27.07.2006 г.  
  Данный закон регулирует обработку, хранение и защиту персональных данных. В рамках автоматизации складского учета — это важно, если система хранит данные о сотрудниках, клиентах или контрагентах. Необходимо обеспечить меры по защите таких данных, предотвратив их утечку или несанкционированный доступ. [2]
* ФЗ «Об информации, информационных технологиях и о защите информации» № 149-ФЗ от 27.07.2006 г.  
  Этот закон устанавливает основные принципы использования информационных технологий, требования к защите информации и правила обеспечения её доступности, конфиденциальности и целостности. [3]
* ФЗ «О бухгалтерском учете» № 402-ФЗ от 06.12.2011 г.  
  Регламентирует требования к учету и хранению документов, связанных с товарно-материальными ценностями, и обязанность ведения учета в электронных системах, включая автоматизированные системы управления складом. [4]
* ГОСТ Р ИСО/МЭК 27001-2012 «Информационная технология. Методы и средства обеспечения безопасности. Системы управления информационной безопасностью. Требования»  
  Определяет требования к созданию и поддержанию системы управления информационной безопасностью, включая контроль за доступом к информации, управление рисками и обеспечение конфиденциальности. [5]
* ГОСТ Р 7.0.97-2016 «Системы менеджмента безопасности информации. Основные положения и словарь»  
  Содержит общие термины и определения, которые применяются в области управления информационной безопасностью, и описывает требования к организации процессов защиты информации. [6]
* ГОСТ Р 34.11-2018 «Информационные технологии. Защита информации. Криптографическая защита информации»  
  Регламентирует использование криптографических методов для обеспечения целостности и конфиденциальности данных в информационных системах. [7]

Agile — гибкая методология разработки программного обеспечения, основанная на итеративном подходе и быстрой адаптации к изменениям. Проект в Agile делится на небольшие циклы, называемые спринтами, каждый из которых длится от одной до четырех недель. В каждом спринте команда реализует часть функциональности, тестирует её и демонстрирует результаты. Такой подход позволяет регулярно получать обратную связь и вносить изменения в проект по мере необходимости, что делает его особенно эффективным для проектов с изменяющимися требованиями.

Основные принципы Agile включают:

* Итеративность разработки
* Гибкость и адаптивность
* Постоянная обратная связь
* Кросс-функциональные команды
  1. **Анализ готовых решений**

На современном рынке существует множество готовых решений для автоматизации складского учета, которые обладают различными функциональными возможностями, уровнем сложности и стоимости. Рассмотрим несколько популярных систем:

1. **1С: Управление торговлей**

**Одним из наиболее распространенных решений для автоматизации складского учета является программа «1С: Управление торговлей». Данное решение поддерживает учет запасов, обработку заказов, ведение остатков, а также формирование отчетности. Преимуществом данной системы является её интеграция с бухгалтерскими и финансовыми модулями, что делает её удобной для средних и крупных предприятий. Однако основной недостаток 1С заключается в её высокой стоимости и необходимости покупки лицензий на использование.**

* **Преимущества:**

**Широкая функциональность для учета товаров и управления заказами.**

**Интеграция с бухгалтерскими и финансовыми модулями, что обеспечивает комплексное управление бизнесом.**

**Соответствие нормативным требованиям и локальным стандартам учета.**

* **Недостатки:**

**Высокая стоимость лицензий и обновлений.**

**Сложность настройки и использования, требующая обучения персонала.**

**Ограниченная гибкость при адаптации под специфические процессы компании.**

1. **SAP ERP**

**SAP ERP — это многофункциональная система управления бизнес-процессами, которая включает модуль складского учета. Она позволяет отслеживать весь жизненный цикл товара на складе — от поступления до продажи или списания. SAP активно используется крупными предприятиями и транснациональными корпорациями благодаря своей гибкости и возможности интеграции с другими модулями системы. Однако SAP ERP является сложной и дорогостоящей в реализации, что делает её менее доступной для малого и среднего бизнеса.**

* **Преимущества:**

**Поддержка полного цикла управления складом и интеграция с другими бизнес-процессами.**

**Высокая степень масштабируемости, подходящая для крупных компаний.**

**Возможность настройки под сложные логистические и производственные процессы.**

* **Недостатки:**

**Высокая стоимость внедрения и обслуживания.**

**Сложность освоения и долгий процесс обучения персонала.**

**Трудоемкость настройки и адаптации под конкретные потребности.**

1. **Odoo**

**Odoo — это система с открытым исходным кодом, которая включает модуль управления складом. Она обладает широким функционалом: отслеживание движения товаров, управление запасами, формирование отчетов. Преимуществом Odoo является её модульная структура, которая позволяет настраивать систему под конкретные нужды бизнеса. Кроме того, за счет открытого исходного кода она более доступна в плане затрат, но может требовать дополнительных усилий на настройку и интеграцию.**

* **Преимущества:**

**Открытый исходный код, позволяющий настраивать систему под специфические потребности.**

**Модульная структура, которая обеспечивает гибкость и адаптивность системы.**

**Низкая стоимость внедрения по сравнению с коммерческими аналогами.**

* **Недостатки:**

**Требует времени и ресурсов на настройку и интеграцию.**

**Ограниченная поддержка по сравнению с крупными коммерческими системами.**

**Возможные сложности с совместимостью при обновлении отдельных модулей.**

1. **Zoho Inventory**

**Zoho Inventory — облачная система для управления запасами и складом, которая позволяет вести учет товаров, отслеживать заказы и контролировать уровень запасов. Она идеально подходит для малого бизнеса, так как предлагает широкий функционал по относительно невысокой цене. Основным преимуществом является её простота в использовании и возможность доступа к данным через интернет с любого устройства. Однако облачные решения не всегда могут обеспечить достаточную гибкость в настройке для крупных предприятий.**

* **Преимущества:**

**Доступность из любого места благодаря облачной архитектуре.**

**Интуитивно понятный интерфейс, подходящий для малого бизнеса.**

**Низкая стоимость внедрения и использования.**

* **Недостатки:**

**Ограниченная гибкость для настройки под специфические процессы крупных компаний.**

**Зависимость от стабильного интернет-соединения.**

**Ограниченный функционал для масштабируемости в крупных компаниях.**

Таким образом, современные информационные системы для складского учета играют ключевую роль в автоматизации процессов управления запасами, позволяя значительно повысить точность учета, снизить затраты и минимизировать человеческий фактор. Применение гибких методологий разработки, таких как Agile, обеспечивает адаптивность системы и позволяет оперативно вносить изменения в ходе проекта. Соблюдение законодательных и нормативных требований (ФЗ и ГОСТы) гарантирует безопасность данных и соответствие стандартам. Анализ готовых решений на рынке показывает, что выбор системы должен основываться на потребностях конкретного бизнеса, и разработка собственной системы может стать оптимальным вариантом для удовлетворения уникальных требований организации.

# **2** **Проектирование системы**

## **Общее назначение системы**

В рамках данного курсового проекта разрабатывается модуль для автоматизации учета товаров на складе, который будет интегрирован в информационную систему предприятия. Основная цель внедрения данной системы — повышение точности учета товарных запасов, сокращение ошибок при работе с данными и повышение прозрачности процесса управления складом. Ожидается, что система упростит учет, контроль и анализ запасов. Основной эффект от внедрения системы заключается в значительном сокращении времени на обработку и ввод данных, а также в уменьшении риска ошибок, возникающих при ручном учете.

## **2.2 Требование к функциям выполняемых системой**

Для решения задач учета и управления запасами складская информационная система должна обеспечивать следующие основные функции:

* Регистрация поступления и выбытия товаров: учет всех операций, связанных с движением товаров на складе.
* Автоматическое формирование отчетов: создание отчетов по остаткам и движениям товаров на складе, что позволяет анализировать текущее состояние запасов и исторические данные.
* Контроль уровня запасов: отслеживание текущего состояния запасов с автоматическим уведомлением о достижении минимального или максимального уровня.
* История операций: ведение полного журнала всех операций для анализа и последующей оптимизации складских процессов.
* Управление пользователями: обеспечение безопасного доступа и управление правами пользователей.

Основные объекты предметной области представлены в (ПРИЛОЖЕНИЕ 1, «Таблица 1» - «Таблица 4»). Схема связи объектов представлена в (ПРИЛОЖЕНИЕ 1, «Рисунок 1»).

ER-диаграмма объектов предметной области представлена в (ПРИЛОЖЕНИЕ 1, «Рисунок 2»). Диаграмма прецедентности представлена в (ПРИЛОЖЕНИЕ 1, «Рисунок 3»).

## **2.3 План тестирования**

Задачей тестирования является проверка правильности реализации функций и устойчивости системы. Необходимо удостовериться, что система корректно выполняет задачи по учету и хранению данных, а также обеспечивает безопасность и доступность информации.

Задачи тестирования включают:

1. Проверку корректности выполнения всех функций, предусмотренных системой.
2. Оценку производительности приложения под нагрузкой при увеличении объёма данных.
3. Тестирование безопасности системы, включая обработку и защиту пользовательских данных.
4. Убедиться в удобстве и понятности пользовательского интерфейса.
5. Проверку совместимости программы с различными версиями операционных систем.

Объекты тестирования:

1. **Авторизация и регистрация пользователей:**

* Проверка входа и регистрации пользователей с разными ролями
* Валидация данных и обработка ошибок при некорректном вводе.

1. **Управление заказами:**

* Проверка правильности аутентификации при введении верных и неверных данных.
* Создание заказа пользователем с проверкой доступности товара.
* Отмена заказа и корректное обновление данных о товаре на складе.
* Изменение статуса заказа сотрудником.

1. **Работа с пользовательскими данными:**

* Управление пользователями (добавление, изменение ролей, удаление) администратором.
* Проверка корректности изменений в базе данных.

1. **Управление товарами:**

* Добавление, редактирование и удаление товаров администратором.
* Проверка корректного обновления данных о товаре в системе.

1. **Генерация отчётов:**

* Создание отчётов о заказах, проверка расчёта сумм заказов и форматирования данных.
* Формирование отчёта о состоянии склада с расчётом общей стоимости товаров.

Таким образом, проведённое тестирование показало, что разработанная система учёта товаров на складе соответствует заявленным требованиям по функциональности, безопасности и удобству использования. Основные функции, такие как регистрация пользователей, создание и отмена заказов, управление товарами, генерация отчётов и выдача чеков, работают корректно и устойчиво. С результатами тестирования можно ознакомиться в (ПРИЛОЖЕНИЕ 2, таблица «План-тест»).

**3 Реализация проекта системы**

**3.1 Описание среды разработки**

Для реализации проекта системы был выбран набор инструментов, которые обеспечивают эффективную разработку, производительность и удобство использования. Описание и обоснование выбора инструментов приведены ниже.

### **3.1.1 Выбор и описание программных инструментов**

Для создания программного модуля системы учёта товаров на складе были проанализированы и выбраны наиболее подходящие программные инструменты, соответствующие требованиям проекта. При этом учитывались их функциональные возможности, простота интеграции, производительность и доступность.

Для написания кода рассматривались такие языки программирования, как Python, Java, C# и C++. В конечном итоге выбор пал на Python благодаря его простоте, читаемости кода и широким возможностям.

Для организации и хранения данных в проекте рассматривались системы управления базами данных, такие как MySQL, PostgreSQL и SQLite. SQLite была выбрана за её лёгкость и возможность работы без установки дополнительного серверного программного обеспечения. Эта встроенная база данных идеально подходит для приложений с локальным хранением данных, обеспечивая простоту настройки и управления.

Для разработки графического пользовательского интерфейса рассматривались библиотеки PyQt и Tkinter. Библиотека PyQt6 была выбрана как наиболее мощное и функциональное решение, предоставляющее доступ к богатым возможностям фреймворка Qt, включая кроссплатформенную поддержку и удобство работы с интерфейсами. PyQt6 позволяет реализовывать сложные интерфейсы с минимальными усилиями, сохраняя при этом гибкость и масштабируемость.

Для работы с данными и создания отчётов использовались инструменты анализа и экспорта данных. Рассматривались библиотеки Pandas, Openpyxl и xlrd. Для работы была выбрана библиотека Pandas благодаря её удобству работы с табличными данными, а также встроенной поддержке экспорта в форматы Excel. Это решение позволяет генерировать отчёты о состоянии склада и заказах, предоставляя пользователю наглядное представление о данных.

Интеграция всех выбранных инструментов происходила в среде разработки PyCharm. PyCharm обеспечивает удобное написание, отладку и тестирование кода, предоставляя инструменты для управления проектами, работы с зависимостями и обеспечения высокой производительности разработчиков.

Таким образом, выбранный стек инструментов — это Python, SQLite, PyQt6, Pandas и PyCharm. Они позволяют реализовать необходимую функциональность, создавая удобное, надёжное и производительное приложение.

### **3.1.2 Обоснование выбора инструментария по разработке**

Выбор инструментов для создания системы учёта товаров на складе был основан на требованиях к простоте внедрения, надёжности и удобстве работы. Подбор инструментов учитывался как их совместимость между собой, так и возможность их использования в локальной среде без необходимости сложных настроек.

Основным языком программирования был выбран **Python**. Этот язык предоставляет широкий функционал для решения задач различной сложности и имеет простой синтаксис, что значительно сокращает время разработки. Благодаря поддержке объектно-ориентированного и процедурного подходов, Python позволяет структурировать код, делая его лёгким для понимания и сопровождения. Ещё одной причиной выбора Python является наличие большого количества готовых библиотек, которые позволяют ускорить реализацию таких функций, как работа с базами данных, создание графических интерфейсов и генерация отчётов. [10], [14]

Для хранения данных была использована встроенная база данных **SQLite**. Это реляционная база данных, которая позволяет управлять данными с использованием стандартного SQL. SQLite не требует развёртывания серверной инфраструктуры, что делает её удобной для использования в локальных приложениях. Её простота в интеграции с Python и минимальные системные требования позволили быстро наладить хранение и обработку данных. [9], [12]

Графический интерфейс был реализован с помощью библиотеки **PyQt6**, которая обеспечивает возможности создания удобных и функциональных оконных приложений. PyQt6 предоставляет мощный инструментарий для построения интерфейсов с использованием технологий Qt, что позволило внедрить современные элементы управления, обеспечивая высокое качество взаимодействия с пользователем. [11]

Для создания отчётов и экспорта данных в удобные для анализа форматы использовалась библиотека **Pandas**. Её возможности включают работу с таблицами данных, автоматизацию расчётов и подготовку сводных данных. Интеграция Pandas с библиотекой **Openpyxl** позволила экспортировать данные в формате Excel, что дало пользователям возможность анализировать и использовать данные вне приложения. [13]

В качестве среды разработки была выбрана **PyCharm**. Эта среда предложила удобный инструментарий для написания, тестирования и отладки кода. Функции авто дополнения, подсветки синтаксиса, а также встроенные инструменты работы с виртуальными окружениями и библиотеками сделали процесс разработки более структурированным и удобным.

В совокупности выбранные инструменты обеспечили успешное выполнение задач проекта, что позволило создать надёжное и интуитивно понятное приложение, соответствующее современным требованиям.

## **3.2 Разработка программного модуля**

### **3.2.1 Реализация пользовательского интерфейса программы**

Для обеспечения удобства и эффективности взаимодействия с программным модулем «Система учёта товара на складе» был спроектирован и реализован простой, интуитивно понятный пользовательский интерфейс. Основное внимание уделено структурированному размещению элементов управления, отображению необходимых данных и обеспечению чёткой обработки действий пользователя. Все элементы интерфейса интегрированы в общую функциональную среду, что создаёт комфортные условия для работы с системой.

При запуске программы нас встречает окно «Вход в систему» (ПРИЛОЖЕНИЕ 3, «Рисунок 4»).

В верхней части окна находится название окна «Вход в систему» и 3 кнопки управления: свернуть, развернуть, закрыть.

Интерфейс окна содержит два текстовых поля для ввода логина и пароля. Поле для ввода пароля защищено, что обеспечивает конфиденциальность введённых данных.

Под текстовыми полями расположены две кнопки: «Войти» и «Регистрация».

Кнопка «Войти» позволяет проверить введённые данные. При успешной аутентификации пользователя, в зависимости от его роли, открывается соответствующее основное окно интерфейса.

Кнопка «Регистрация» открывает окно регистрации нового пользователя, позволяя создать учётную запись для дальнейшей работы с программой. (ПРИЛОЖЕНИЕ 3, «Рисунок 5»).

При вводе некорректного логина или пароля появляется сообщение об ошибке, указывающее на необходимость проверить введённые данные.

В верхней части окна находится название окна «Регистрация» и кнопка управления закрыть.

Окно содержит два подписанных текстовых поля для ввода логина и пароля, а также кнопку «Зарегистрироваться». Поле для ввода пароля защищено от визуального отображения ввода, что обеспечивает конфиденциальность данных пользователя.

При вводе некорректных данных, например, если пароль не соответствует минимальным требованиям по длине или отсутствует специальный символ, появится соответствующее сообщение об ошибке. Если логин уже зарегистрирован, пользователю также будет предоставлено предупреждение.

После успешного заполнения всех полей появится уведомление о завершении регистрации, и окно автоматически закроется. Пользователь будет перенаправлен в окно входа в систему.

Окно пользователя:

В верхней части окна находится название «Система учёта товаров на складе», а также стандартные кнопки управления окном: свернуть, развернуть и закрыть.

Основное пространство интерфейса занимает таблица, в которой отображается информация о доступных товарах. В таблице представлены такие параметры, как наименование товара, его вес в граммах, стоимость за единицу в рублях и количество товара на складе.

Под таблицей расположены кнопки, позволяющие пользователю выполнять основные действия.

* Кнопка «Сделать заказ» открывает окно для оформления нового заказа, где пользователь может выбрать товар, указать количество и завершить оформление.
* Кнопка «Мои заказы» предоставляет доступ к списку уже оформленных заказов с подробной информацией о каждом заказе, включая id заказа, наименование товара, количество, цену за единицу и общую стоимость.
* Кнопка «Отменить заказ» даёт возможность отменить ранее оформленный заказ, при этом выбранное количество товара возвращается на склад.
* Кнопка «Выйти из аккаунта» позволяет завершить работу и вернуться на экран входа в систему. (ПРИЛОЖЕНИЕ 3, «Рисунок 6» – «Рисунок 9»).

Окно сотрудника:

В верхней части окна находится заголовок «Система учёта товаров на складе» и стандартные кнопки управления окном: свернуть, развернуть и закрыть.

Центральную часть интерфейса занимает таблица, в которой отображается информация о заказах. В таблице представлены такие параметры, как ID заказа, имя пользователя, наименование товара, количество единиц товара в заказе и текущий статус заказа.

Под таблицей расположены кнопки для выполнения основных действий сотрудника.

* Кнопка «Обновить статус заказа» открывает окно, позволяющее выбрать заказ и изменить его статус на один из доступных.
* Кнопка «Сформировать отчёт» предоставляет возможность создать и сохранить отчёт о заказах в формате Excel, который может использоваться для анализа и хранения данных.
* Кнопка «Выдать чек», которая открывает меню выбора пользователя и позволяет сформировать Excel-документ с деталями заказов выбранного клиента, включая общую стоимость.
* Кнопка «Выйти из аккаунта», которая позволяет завершить сеанс работы сотрудника и вернуться на экран входа в систему. (ПРИЛОЖЕНИЕ 3, «Рисунок 10» – «Рисунок 14»).

Окно администратора:

В верхней части окна отображается заголовок «Система учёта товаров на складе» и стандартные элементы управления окном: свернуть, развернуть, закрыть.

Центральную часть интерфейса занимает таблица с данными о товарах на складе. Таблица отображает следующие столбцы: ID товара, наименование, вес, стоимость и количество единиц товара на складе. Эта информация позволяет администратору быстро получить сводку о текущем состоянии склада.

Под таблицей располагаются кнопки для выполнения действий администратора.

* Кнопка «Добавить товар» открывает окно, в котором можно внести данные о новом товаре, такие как его название, вес, стоимость и количество.
* Кнопка «Редактировать товар» позволяет изменить данные существующего товара.
* Кнопка «Удалить товар» предназначена для удаления товаров из базы данных. При её нажатии открывается окно, где администратор может выбрать товар для удаления. После подтверждения выбранный товар удаляется из системы.
* Кнопка «Управление пользователями» позволяет администратору управлять учетными записями пользователей. Нажатие на кнопку открывает окно, в котором можно изменять роли пользователей или удалить учетную записи из системы.
* Кнопка «Просмотр изменений» предоставляет доступ к журналу изменений. После её нажатия открывается окно с таблицей, где фиксируются все действия, произведённые в системе, включая добавление, изменение и удаление товаров, а также изменения ролей пользователей.
* Кнопка «Отчёт о состоянии склада» позволяет сформировать отчёт в формате Excel с данными о товарах, включая их стоимость и общую сумму на складе.
* Кнопка «Выйти из аккаунта», обеспечивающая завершение работы администратора и возвращение к окну входа в систему. (ПРИЛОЖЕНИЕ 3, «Рисунок 15» – «Рисунок 21»).

**3.2.2 Описание кодом функциональных узлов модуля**

Программный модуль системы учёта товаров на складе включает в себя несколько взаимосвязанных функциональных частей, которые обеспечивают работу с базой данных, графическим интерфейсом и бизнес-логикой приложения.

Окна входа и регистрации:

Начальным этапом взаимодействия пользователя с системой является окно входа. В нём расположены поля для ввода логина и пароля, а также кнопки «Войти» и «Регистрация». При нажатии на кнопку «Войти» происходит проверка данных в базе, и, если авторизация успешна, открывается основное окно программы, соответствующее роли пользователя. Кнопка «Регистрация» позволяет перейти в окно регистрации, где можно создать новый аккаунт, введя логин и пароль. Эти функции представлены в (ПРИЛОЖЕНИЕ 4, «Рисунок 22» и «Рисунок 23»).

Основное окно пользователя:

У пользователя есть функции: оформить заказ, просмотреть свои заказы и отменить заказ. Эти функции представлены в (ПРИЛОЖЕНИЕ 4, «Рисунок 24» - «Рисунок 26»).

Основное окно сотрудника:

У Сотрудника есть функции: обновление статуса заказа, формирование отчёта о заказах и формирование чека. Эти функции представлены в (ПРИЛОЖЕНИЕ 4, «Рисунок 27» - «Рисунок 29»).

Основное окно администратора:

У администратора есть функции: добавить товар, редактировать товар, удалить товар, управлять пользователями, просматривать изменения таблицы товаров и создания отчёта о состояние склада. Эти функции представлены в (ПРИЛОЖЕНИЕ 4, «Рисунок 30» - «Рисунок 35»).

Так же у всех пользователь присутствует кнопка выйти из аккаунта, функция представлена в (ПРИЛОЖЕНИЕ 4, «Рисунок 36»).

* + 1. **Результат работы и тестирования**
  1. Кейс «Регистрация пользователя»

При вводе данных для регистрации система проверяет логин и пароль. Пароль должен соответствовать требованиям по длине и символам. При успешной регистрации появляется сообщение, иначе — ошибка.

* 1. Кейс «Вход в систему»

Пользователь вводит логин и пароль. При корректных данных открывается интерфейс в зависимости от роли. При ошибке входа система уведомляет пользователя.

* 1. Кейс «Создание заказа»

Пользователь выбирает товар из списка, указывает количество, и система проверяет доступность на складе. При успешном оформлении выводится сообщение о сумме заказа.

* 1. Кейс «Отмена заказа»

Пользователь выбирает заказ для отмены. После подтверждения система обновляет количество товара на складе.

* 1. Кейс «Обновление статуса заказа»

Сотрудник изменяет статус заказа через интерфейс. Изменения сразу отображаются в таблице.

* 1. Кейс «Форматирование отчёта о заказах»

Создаётся Excel-файл с данными о заказах, включая стоимость и общую сумму.

* 1. Кейс «Создание чека»

Сотрудник генерирует чек для выбранного пользователя в виде Excel-файла, содержащего заказы и их общую стоимость.

* 1. Кейс «Управление товарами»

Администратор добавляет, редактирует и удаляет товары через соответствующие кнопки. Таблица обновляется автоматически.

* 1. Кейс «Управление пользователями»

Администратор изменяет роли пользователей и удаляет их из системы. Изменения отображаются в реальном времени.

* 1. Кейс «Создание отчёта о состояние склада»

Администратор генерирует отчёт, включающий данные о товарах и их общей стоимости.

Подробнее с тестами можно ознакомиться в (ПРИЛОЖЕНИЕ 2, таблица «Тест-кейс»).

Система обеспечивает удобство использования, надежное хранение данных, быстрое выполнение операций. Подробнее ознакомиться с проектом можно на GitHub. [8]

**ЗАКЛЮЧЕНИЕ**

В заключение следует отметить значимость разработанной системы учёта товаров на складе для автоматизации складских операций и упрощения управления процессами, связанными с учётом товаров, заказов и взаимодействием между пользователями.

В рамках проекта был создан программный модуль, предоставляющий возможности по управлению товарами на складе, оформлению и обработке заказов, а также учёту изменений, происходящих в системе.

Для реализации проекта были определены основные требования, проведён анализ доступных технологий, выбраны оптимальные инструменты разработки, такие как Python, PyQt6 и SQLite3, а также выполнена программная реализация интерфейса и функциональных модулей. Разработка системы включала проектирование интерфейса, программирование логики взаимодействия, интеграцию с базой данных и проведение тестирования.

Результаты тестирования подтвердили работоспособность системы, её соответствие предъявленным требованиям и корректность выполнения всех заявленных функций. Программа предоставляет удобный и интуитивно понятный интерфейс для каждой роли и обеспечивает функциональность, необходимую для решения задач, связанных с управлением складскими процессами.

Несмотря на успешное выполнение поставленных задач, необходимо отметить, что система может быть дополнительно усовершенствована. Например, возможно расширение функционала за счёт интеграции аналитических инструментов для прогнозирования спроса на товары и добавления системы уведомлений о низких остатках на складе.

Результаты работы продемонстрировали, что разработанная система учёта товаров на складе повышает эффективность управления складскими процессами, снижает временные затраты и минимизирует вероятность ошибок. Данный программный модуль может быть полезным инструментом для предприятий, стремящихся к автоматизации и оптимизации своих внутренних процессов.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

Стандарты и законодательные материалы

* 1. «Конституция Российской Федерации» (принята всенародным голосованием 12.12.1993 с изменениями, одобренными в ходе общероссийского голосования 01.07.2020)
  2. Федеральный закон «О персональных данных» от 27.07.2006 № 152-ФЗ. – М.: Государственная Дума Российской Федерации, 8 июля 2006 года.
  3. Федеральный закон «Об информации, информационных технологиях и о защите информации» от 27.07.2006 № 149-ФЗ. – М.: Государственная Дума Российской Федерации, 8 июля 2006 года.
  4. Федеральный закон «О бухгалтерском учете» от 06.12.2011 № 402-ФЗ. – М.: Государственная Дума Российской Федерации, 6 декабря 2011 года.
  5. ГОСТ Р ИСО/МЭК 27001-2012. Национальный стандарт Российской Федерации. Информационная технология. Методы и средства обеспечения безопасности. Системы менеджмента информационной безопасности. Требования. – М.: Стандартинформ, 2012.
  6. ГОСТ Р 7.0.97-2016. Национальный стандарт Российской Федерации. СИБИД. Библиографическая запись. Библиографическое описание. Общие требования и правила составления. – М.: Стандартинформ, 2016.
  7. ГОСТ Р 34.11-2018. Национальный стандарт Российской Федерации. Информационная технология. Криптографическая защита информации. Хэш-функция. – М.: Стандартинформ, 2018.

Монография

* 1. Бричковский А. В. «Система учёта товаров на складе «Инвентарист»» https://github.com/TAToshka159/Inventarist.git (05.12.2024)

Учебники и учебные пособия

* 1. «MySQL по максимуму» 3-е изд. / Б. Шварц, П. Зайцев, В. Ткаченко. - Санкт-Петербург: Питер, 2018
  2. «Простой Python. Современный стиль программирования» 2-е изд. - Санкт-Петербург: Питер, 2021.
  3. «Python 3 и PyQt 6. Разработка приложений.» / Н.А. Прохоренок, В.А. Дронов. - Санкт-Петербург: БХВ-Петербург, 2023.

Интернет источники

* 1. «Руководство по работе с базами данных в python» https://metanit.com/python/database/ (30.11.2024)
  2. Антон Яценко «Работаем с Pandas: основные понятия и реальные данные» https://skillbox.ru/media/code/rabotaem-s-pandas-osnovnye-ponyatiya-i-realnye-dannye/ (30.11.2024)
  3. «Руководство по языку программирования Python» https://metanit.com/python/tutorial/ (30.11.2024)
  4. Ксения Килина «Что такое складской учёт и как его вести» https://www.moysklad.ru/poleznoe/shkola-torgovli/organizaciya-skladskogo-ucheta/?utm\_source=google.com&utm\_medium=organic&utm\_campaign=google.com&utm\_referrer=google.com (30.11.2024)

# **ПРИЛОЖЕНИЕ 1**

Диаграммы, схемы и таблицы

Таблица 1 - Таблица «Пользователи»

|  |  |  |  |
| --- | --- | --- | --- |
| Объект | Атрибут | Тип данных | Описание |
| Пользователь | ID\_пользователя | int (primary key) | Идентификатор  пользователя |
|  | Логин | varchar(50) | Имя пользователя для входа |
|  | Пароль | varchar(50) | Пароль пользователя |
|  | Роль | varchar(20) | Уровень прав доступа |

Таблица 2 - Таблица «Товары»

|  |  |  |  |
| --- | --- | --- | --- |
| Объект | Атрибут | Тип данных | Описание |
| Товары | ID\_товара | int (primary key) | Идентификатор товара |
|  | Наименование | varchar(100) | Название товара |
|  | Вес | float | Вес товара |
|  | Стоимость | decimal(10,2) | Стоимость единицы товара |
|  | Количество | int | Количество товара на складе |

Таблица 3 - Таблица «Заказы»

|  |  |  |  |
| --- | --- | --- | --- |
| Объект | Атрибут | Тип данных | Описание |
| Заказы | ID\_заказа | int (primary key) | Идентификатор товара |
|  | ID\_пользователя | int foreign key | Идентификатор пользователя |
|  | ID\_товара | int foreign key | Идентификатор товара |
|  | Количество | int | Количество товара на складе |
|  | Статус | varchar(20) | Статус заказа |
|  | Сумма | decimal(10,2) | Сумма заказа |

Таблица 4 - Таблица «История Изменений»

|  |  |  |  |
| --- | --- | --- | --- |
| Объект | Атрибут | Тип данных | Описание |
| Изменения | ID\_изменений | int (primary key) | Идентификатор изменений |
|  | ID\_пользователя | int | Идентификатор пользователя |
|  | Дата\_Время | datetime | Точная дата и время |
|  | Описание | text | Описание изменений |

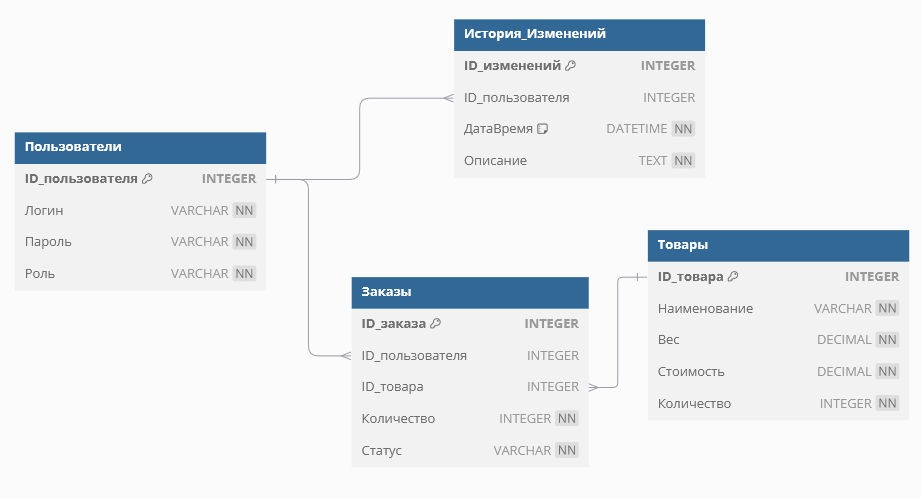


Рисунок 1 - Визуальная схема связи между объектами в базе данных

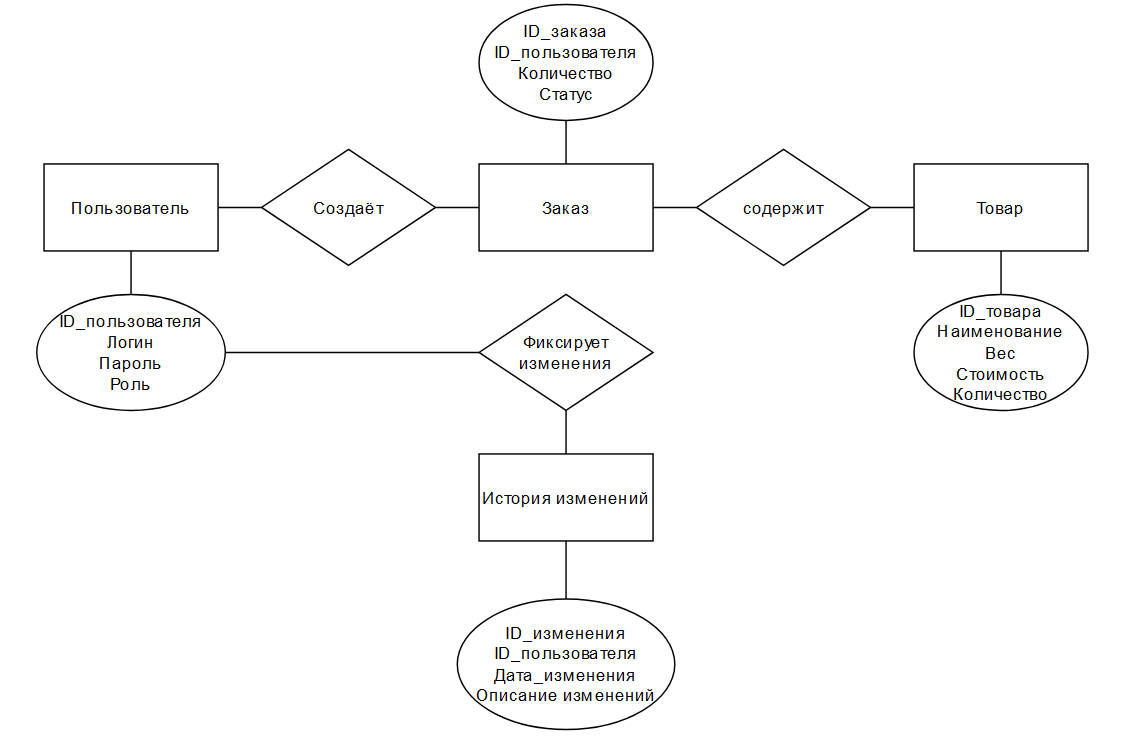


Рисунок 2 - ER-диаграмма

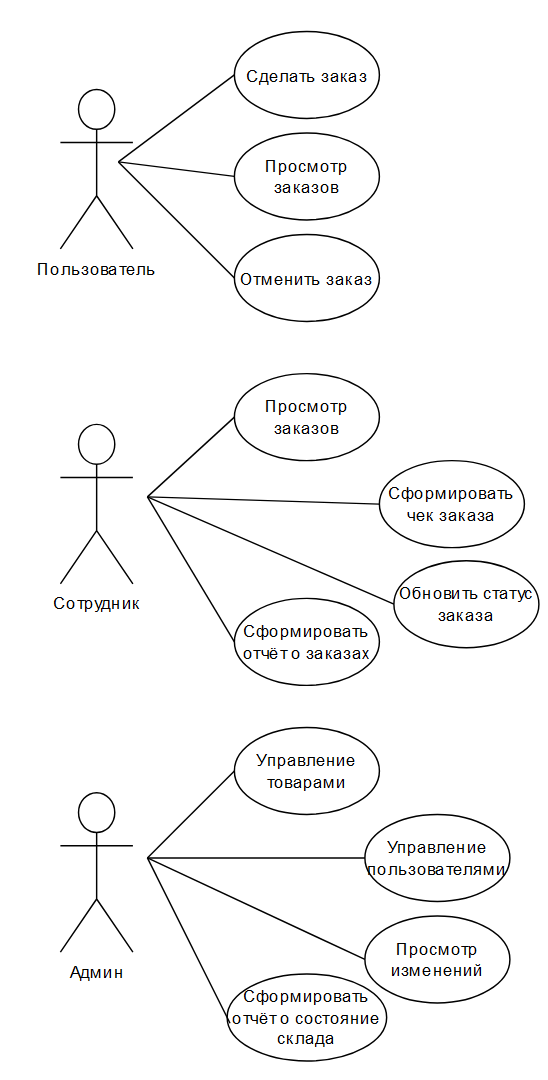


Рисунок 3 - Диаграмма прецедентности

# **ПРИЛОЖЕНИЕ 2**

План тест

Таблица 5 – План тест

| Порядковый номер | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат |
| --- | --- | --- | --- | --- |
| 1 | Регистрация | Логин | Ввести логин «test» | Успех «Регистрация прошла успешно» |
|  |  | Пароль | Ввести пароль «test1!» |
|  |  | Кнопка  «Зарегистрироваться» | Нажать на кнопку |
| Фактический результат | | | | |
| 2 | Вход | Логин | Ввести логин «test» | Успех «Вход выполнен. Роль:  Пользователь» |
|  |  | Пароль | Ввести пароль «test1!» |
|  |  | Кнопка «Войти» | Нажать на кнопку |
| Фактический результат | | | | |
| 3 | Создание заказа | Выпадающий список | Выбрать из списка «мышку» | Успех «Заказ оформлен» |
|  |  | Количество | Ввести 1 |
|  |  | Кнопка «Оформить заказ» | Нажать на кнопку |

Продолжение таблицы 5

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Порядковый номер | | Наименование функциональности | | Наименование поля | Тестовый набор | | Ожидаемый результат | |
| Фактический результат | | | | | | | | |
| 4 | Отмена заказа | | Выпадающий список «Заказ» | | | Выбрать заказ ID 10 | | Успех «Заказ отменён» |
|  |  | | Отменить заказ | | | Нажать на кнопку | |
|  |  | | Кнопка «Yes» | | | Нажать на кнопку | |
| Фактический результат | | | | | | | | |
| 6 | Обновление статуса заказа | | Выпадающий список «Заказ» | | | Выбрать заказ ID 10 | | Успех «Статус заказа обновлён», таблица обновилась |
|  |  | | Выпадающий список «Статус заказа» | | | Выбрать статус Выдан | |
|  |  | | Кнопка «Обновить статус» | | | Нажать на кнопку | |

Продолжение таблицы 5

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Порядковый номер | | Наименование функциональности | | Наименование поля | Тестовый набор | | Ожидаемый результат | |
| Фактический результат | | | | | | | | |
| 7 | Формирование отчёта | | Кнопка «Сформировать отчёт» | | | Нажать на кнопку | | «Отчёт успешно сохранён в: «путь сохранения»» |
|  |  | | Кнопка «Сохранить» | | | Нажать на кнопку | |
| Фактический результат | | | | | | | | |
| 8 | Формирование чека | | Кнопка «Выдать чек» | | | Нажать на кнопку | | «Чек успешно сохранён в: «путь сохранения»» |
|  |  | | Выпадающий список | | | Выбираем test | |
|  |  | | Кнопка «Создать чек» | | | Нажать на кнопку | |

Продолжение таблицы 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Порядковый номер | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат |
|  |  | Кнопка «Сохранить» | Нажать на кнопку |  |
| Фактический результат | | | | |
| 9 | Добавление товара | Кнопка «Добавить товар» | Нажать на кнопку | Успех «Товар добавлен», таблица обновилась |
|  |  | Наименование | Ввести Батарейка |
|  |  | Вес | Ввести 50 |
|  |  | Стоимость | Ввести 100 |
|  |  | Количество | Ввести 2 |
|  |  | Кнопка «Добавить» | Нажать на кнопку |

Продолжение таблицы 5

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Порядковый номер | | Наименование функциональности | | Наименование поля | Тестовый набор | | Ожидаемый результат | |
| Фактический результат | | | | | | | | |
| 10 | Редактировать товар | | Кнопка «Редактировать товар» | | | Нажать на кнопку | | Успех «Изменения сохранены», таблица обновилась |
|  |  | | ID товара | | | Ввести 5 | |
|  |  | | Новое наименование | | | Ввести Болт | |
|  |  | | Новый вес | | | Ввести 5 | |
|  |  | | Новая стоимость | | | Ввести 10 | |
|  |  | | Новое количество | | | Ввести 100 | |
|  |  | | Кнопка «Сохранить изменения» | | | Нажать на кнопку | |
| Фактический результат | | | | | | | | |
| 11 | Удаление товара | | Кнопка «Удалить товар» | | | Нажать на кнопку | | Успех «Товар успешно удалён», таблица обновилась |
|  |  | | Выпадающий список | | | Выбрать Болт (ID: 5) | |
|  |  | | Кнопка «Удалить» | | | Нажать на кнопку | |
|  |  | | Кнопка «Yes» | | | Нажать на кнопку | |

Продолжение таблицы 5

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Порядковый номер | | Наименование функциональности | Наименование поля | | Тестовый набор | | Ожидаемый результат | |
| Фактический результат | | | | | | | | |
| 12 | Управление пользователями | | | Кнопка «Управление пользователями» | | Нажать на кнопку | | Успех «Роли обновлены» |
|  |  | | | Выпадающий список «Роли» | | Выбрать роль «Сотрудник» у пользователя «test» | |
|  |  | | | Кнопка «Обновить роли пользователей» | | Нажать на кнопку | |
| Фактический результат | | | | | | | | |

Продолжение таблицы 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Порядковый номер | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат |
|  |  | Чекбокс | Поставить галочку около пользователя «test» | Успех «Пользователи удалены» |
|  |  | Кнопка «Удалить выбранных пользователей» | Нажать кнопку |
| Фактический результат | | | | |
| 13 | Отчёт о состояние склада | Кнопка «Отчёт о состояние» | Нажать на кнопку | «Отчёт успешно сохранён в: «путь сохранения»» |
|  |  | Кнопка «Сохранить» | Нажать на кнопку |
| Фактический результат | | | | |

# **ПРИЛОЖЕНИЕ 3**

Скриншоты программы и функций кода

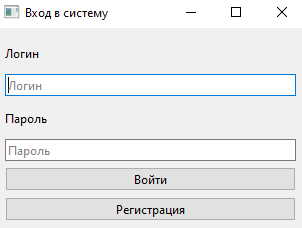
****

Рисунок 4 – Окно авторизации

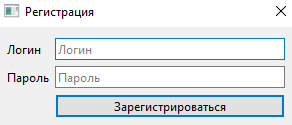


Рисунок 5 – Окно регистрации

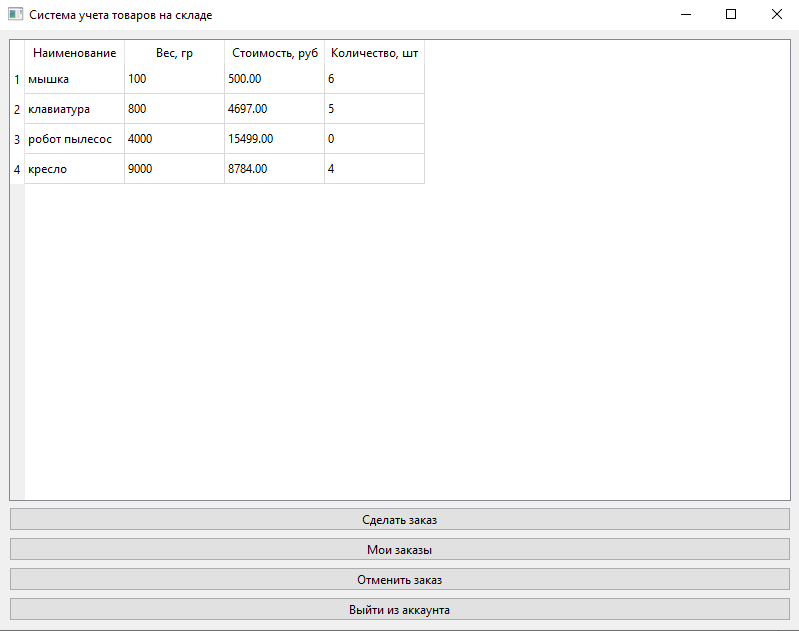


Рисунок 6 – Окно пользователя

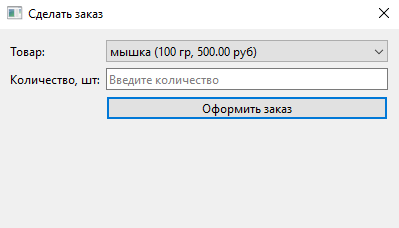


Рисунок 7 – Окно кнопки «Сделать заказ»

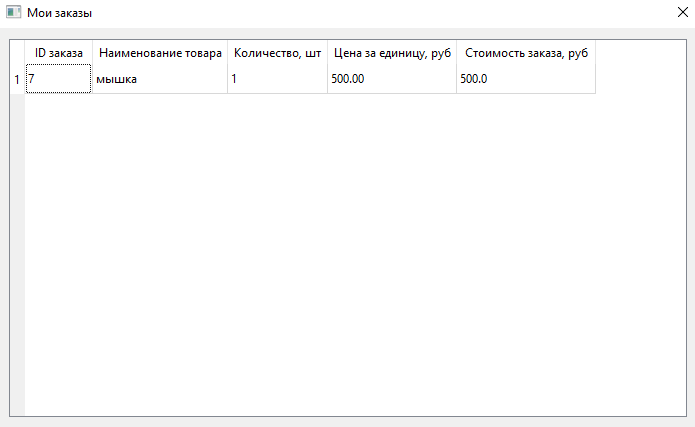


Рисунок 8 – Окно кнопки «Мои заказы»

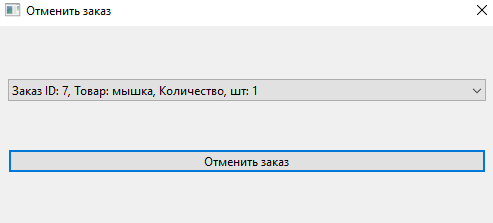


Рисунок 9 – Окно кнопки «Отменить заказ»

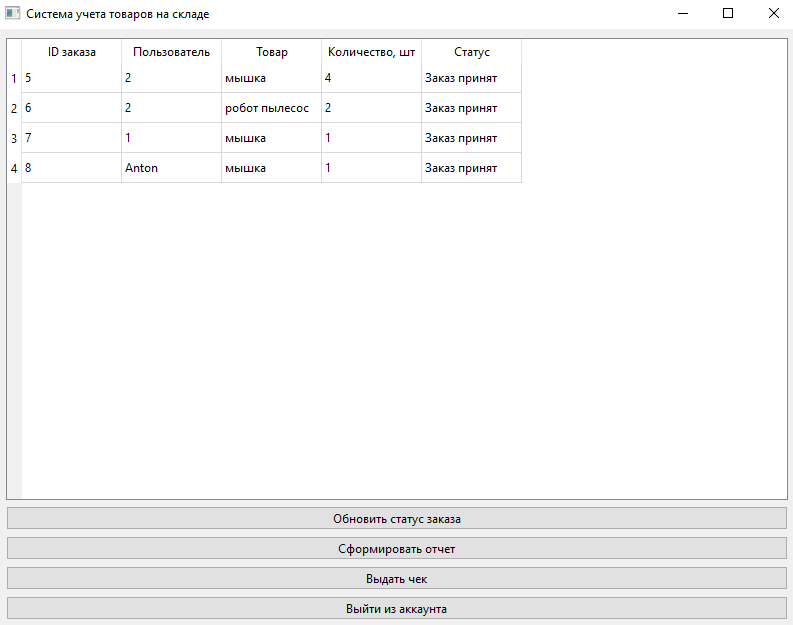


Рисунок 10 – Окно сотрудника

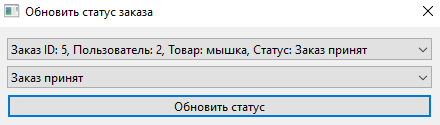


Рисунок 11 – Окно кнопки «Обновить статус заказа»

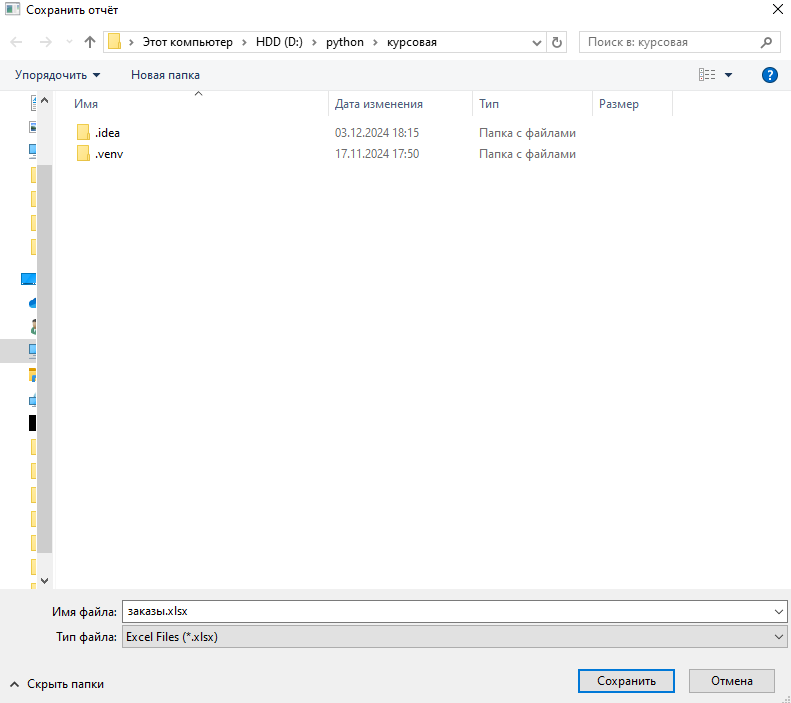


Рисунок 12 – Функция кнопки «Сформировать отчёт»

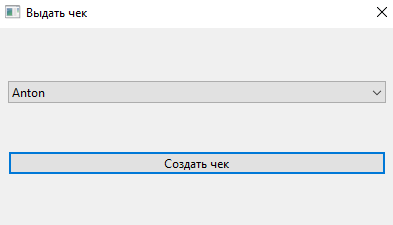


Рисунок 13 – Окно кнопки «Выдать чек»

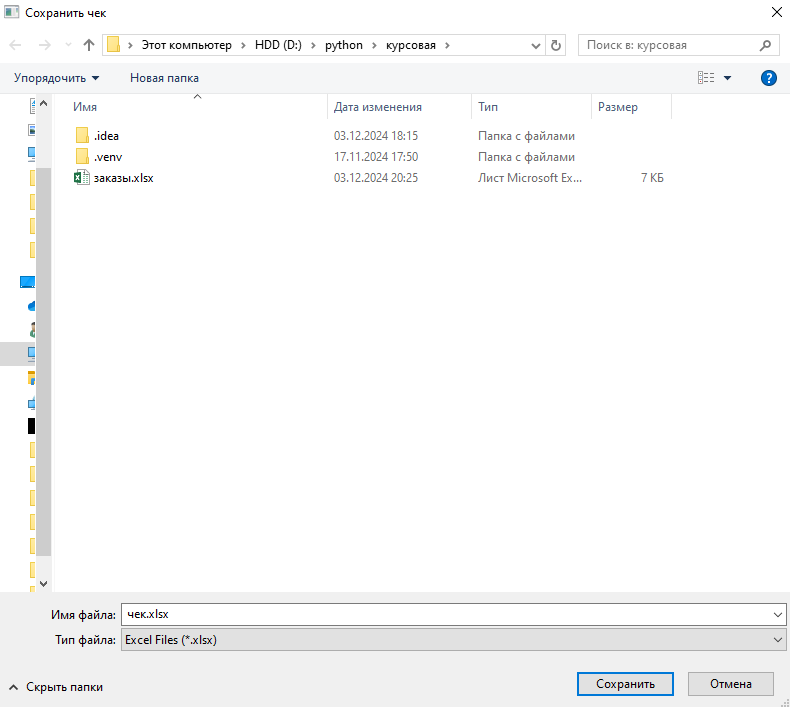


Рисунок 14 – Функционал кнопки «Создать чек»

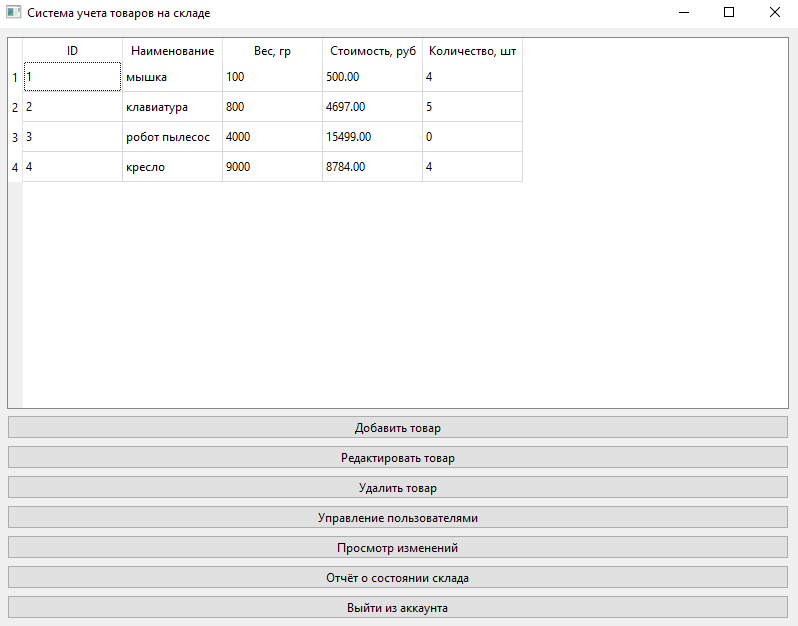


Рисунок 15 – Окно администратора

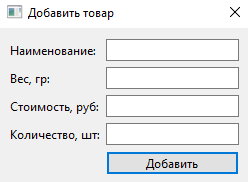


Рисунок 16 – Окно кнопки «Добавить товар»

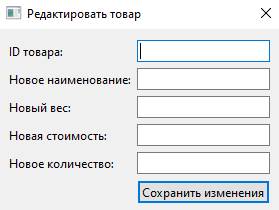


Рисунок 17 – Окно кнопки «Редактировать товар»

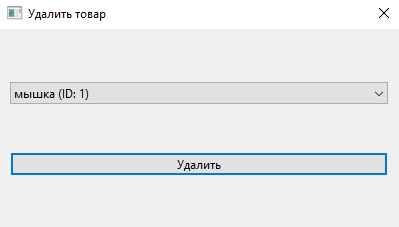


Рисунок 18 – Окно кнопки «Удалить товар»

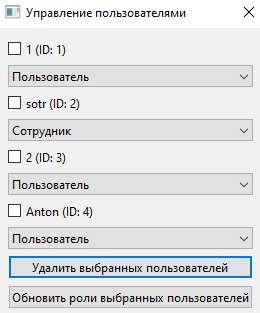


Рисунок 19 – Окно кнопки «Управление пользователями»

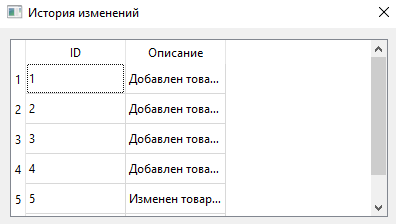


Рисунок 20 – Окно кнопки «Просмотр изменений»

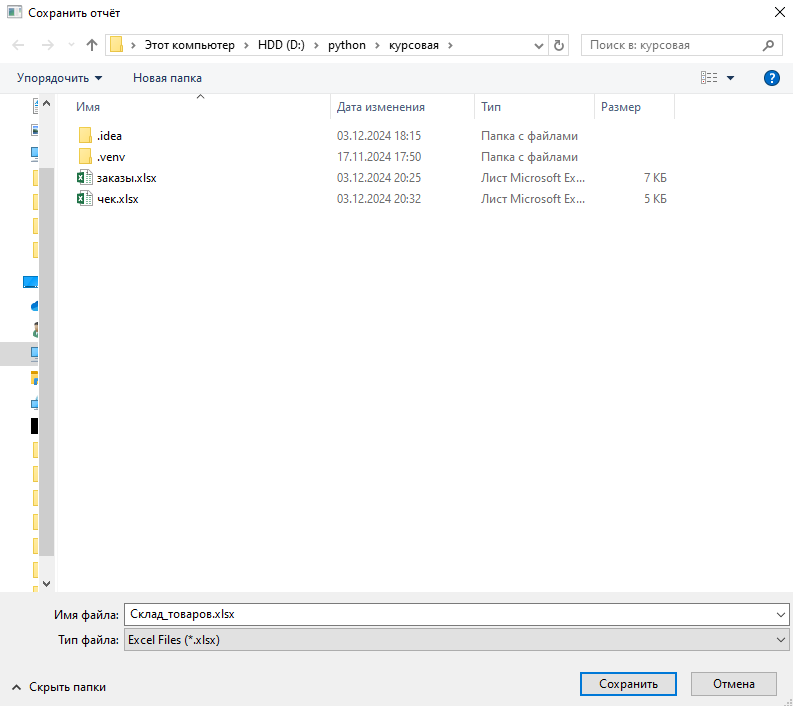


Рисунок 21 – Функционал кнопки «Отчёт о состояние склада»

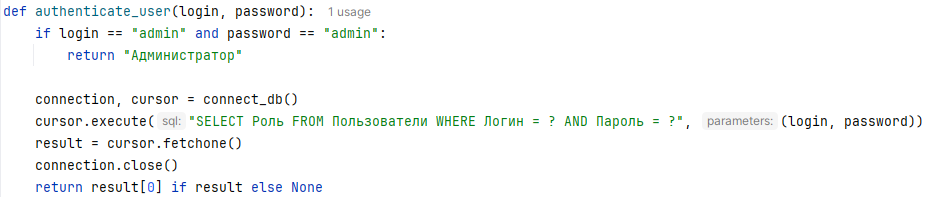


Рисунок 22 – Функция входа в аккаунт



Рисунок 23 – Функция регистрации

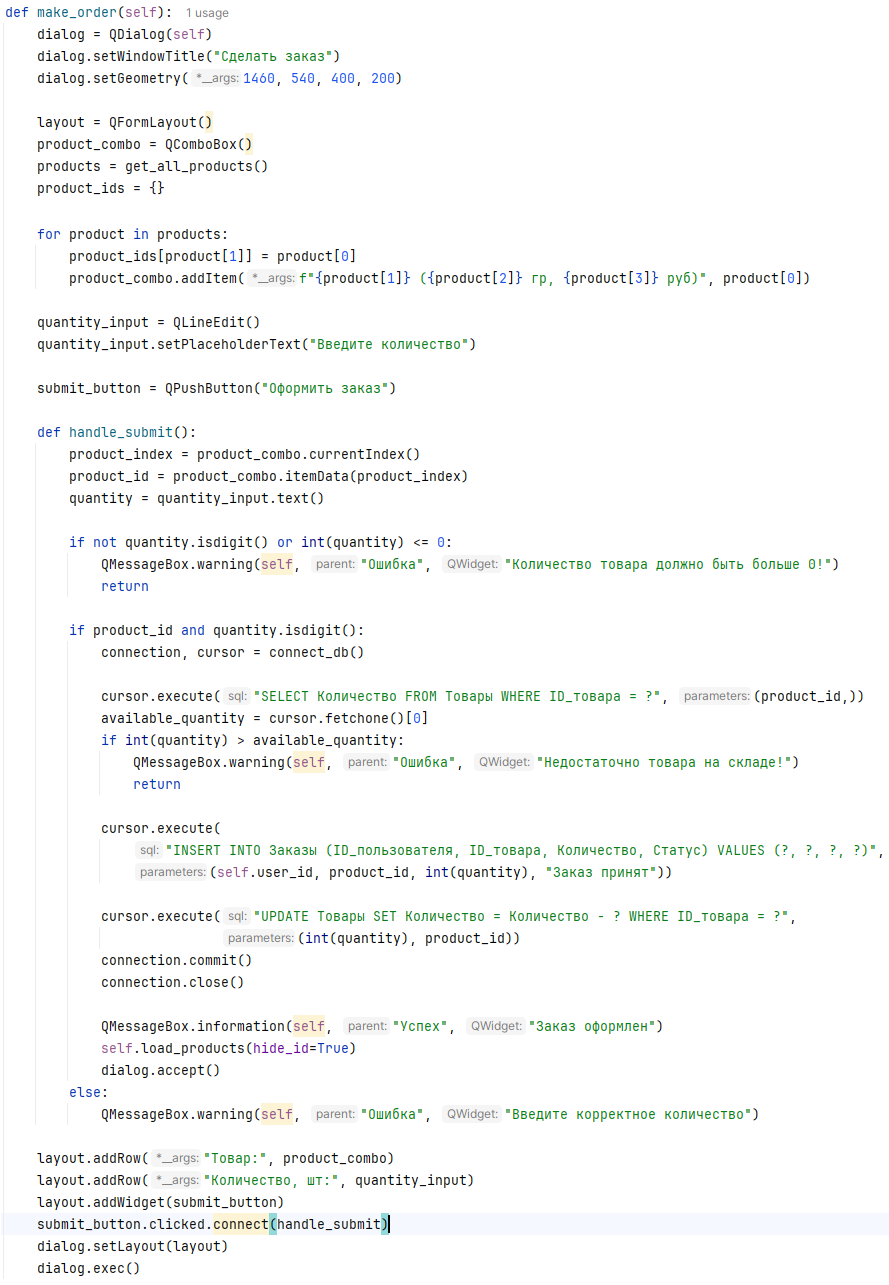


Рисунок 24 – Функция оформления заказа



Рисунок 25 - Функция просмотра своих заказов



Рисунок 26 - Функция отмены заказа



Рисунок 27 – Функция обновления статуса заказа



Рисунок 28 – Функция создания отчёта о заказах

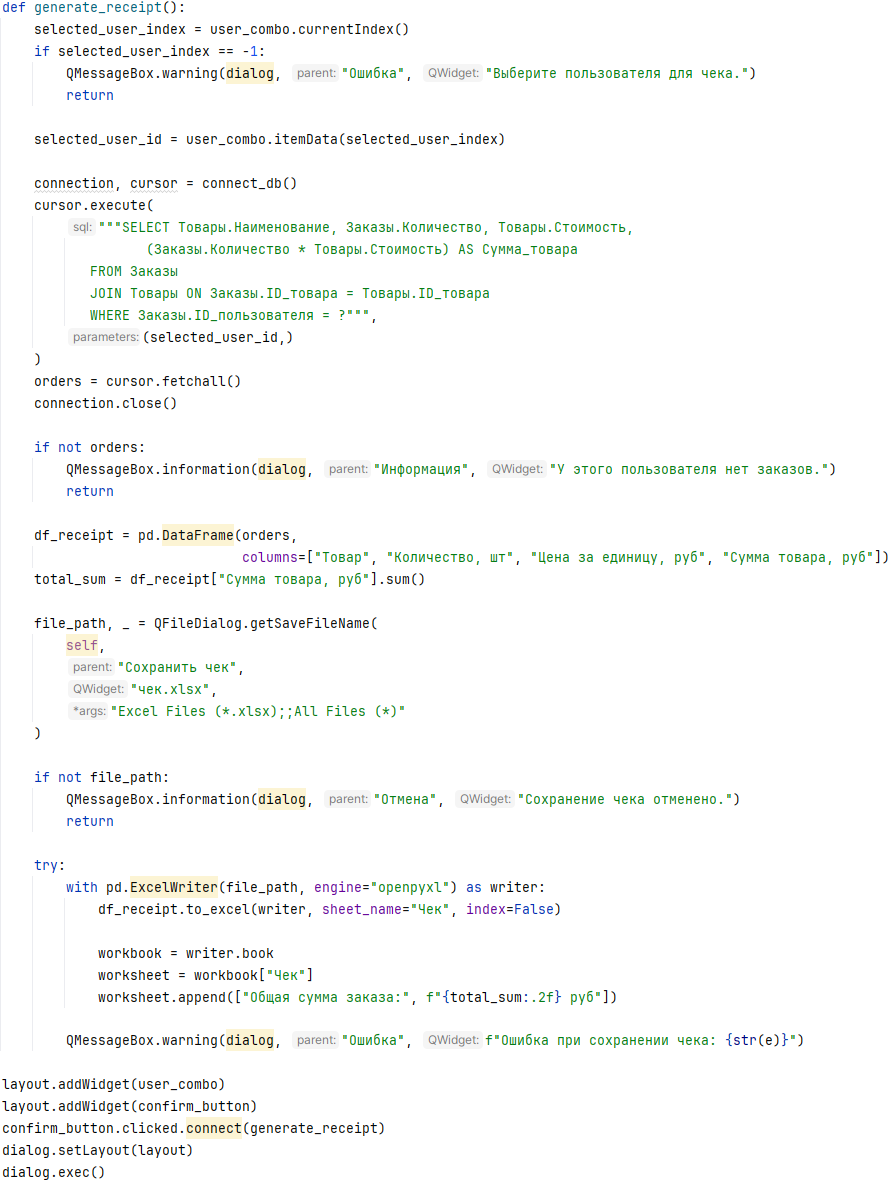


Рисунок 29 – Функция формирования чека

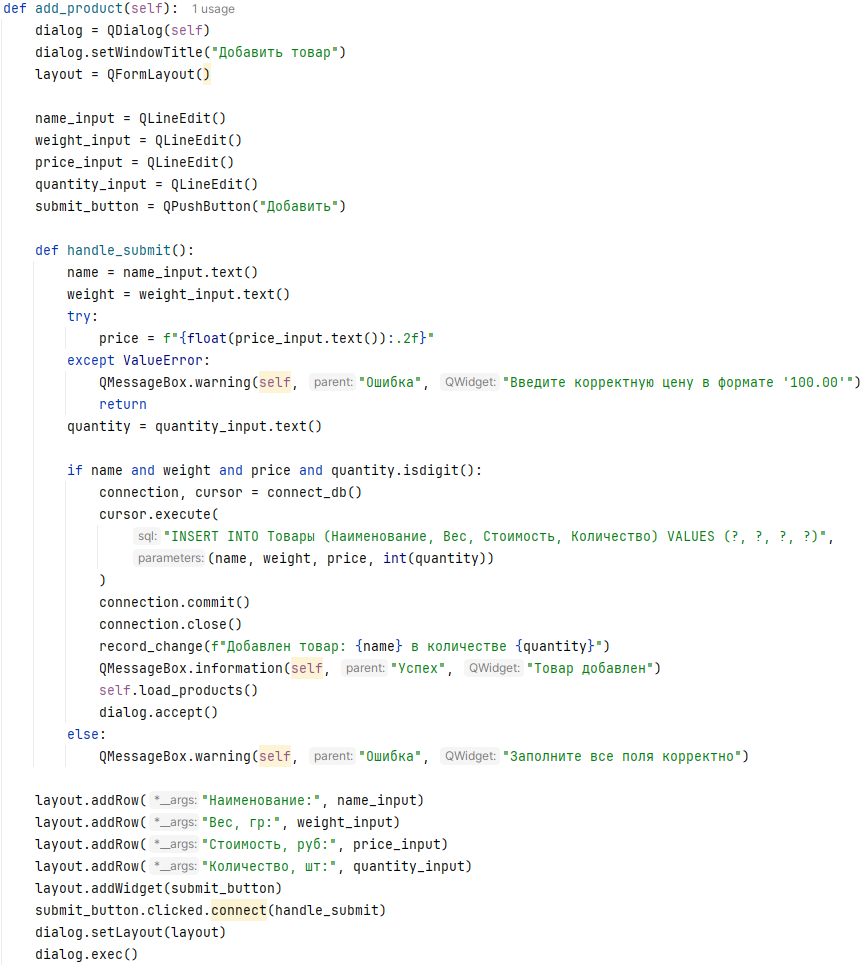


Рисунок 30 – Функция добавления товара



Рисунок 31 – Функция редактирования товара

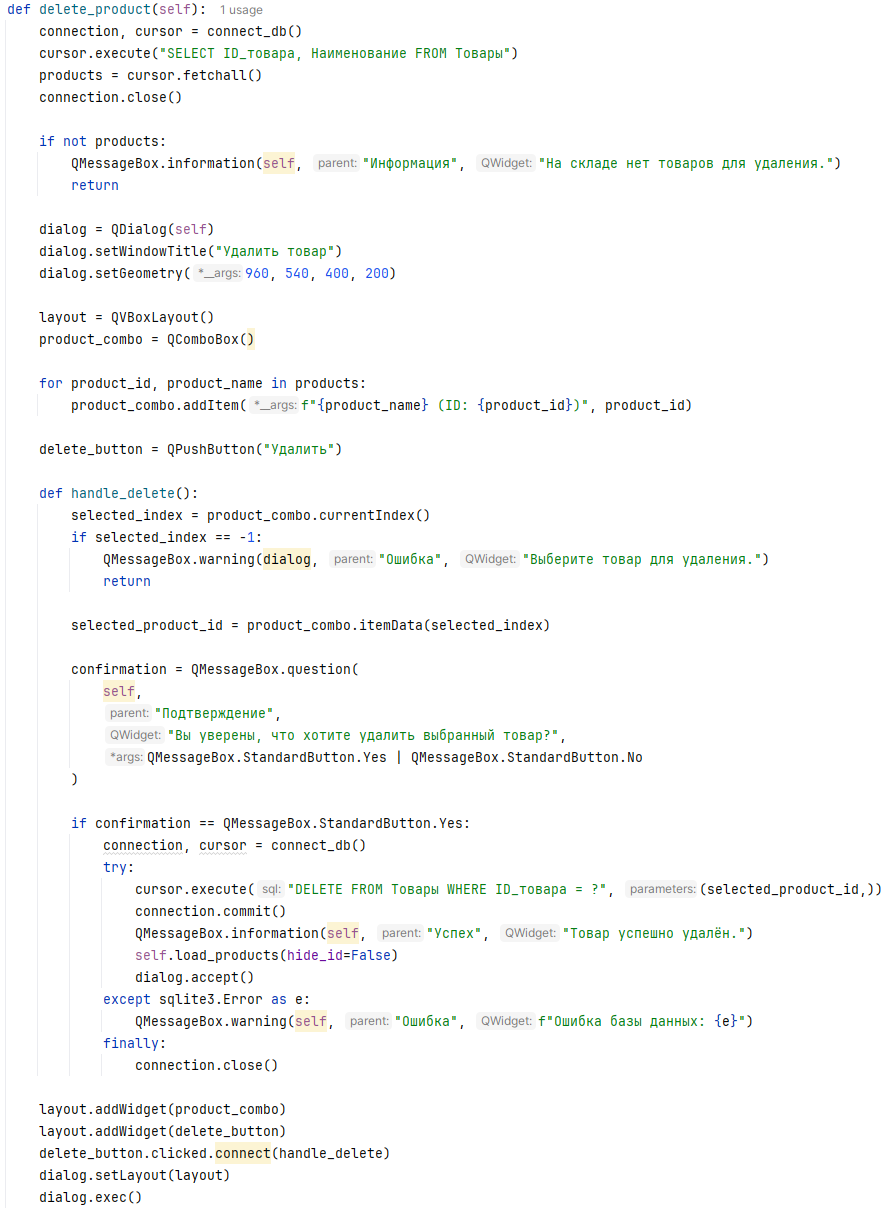


Рисунок 32 – Функция удаления товара



Рисунок 33 – Функция управления пользователями

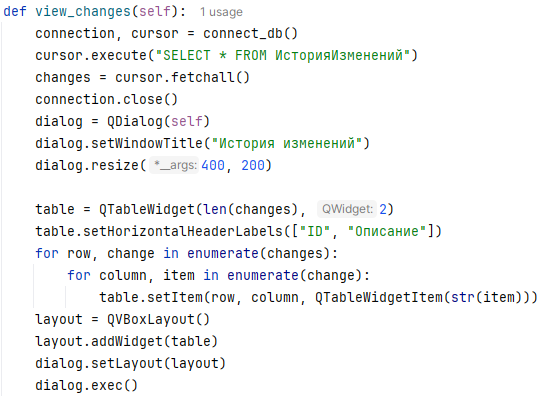


Рисунок 34 – Функция просмотра изменений



Рисунок 35 – Функция создания отчёта о состояние склада

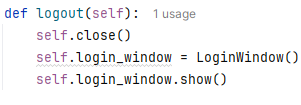


Рисунок 36 – Функция выхода из аккаунта