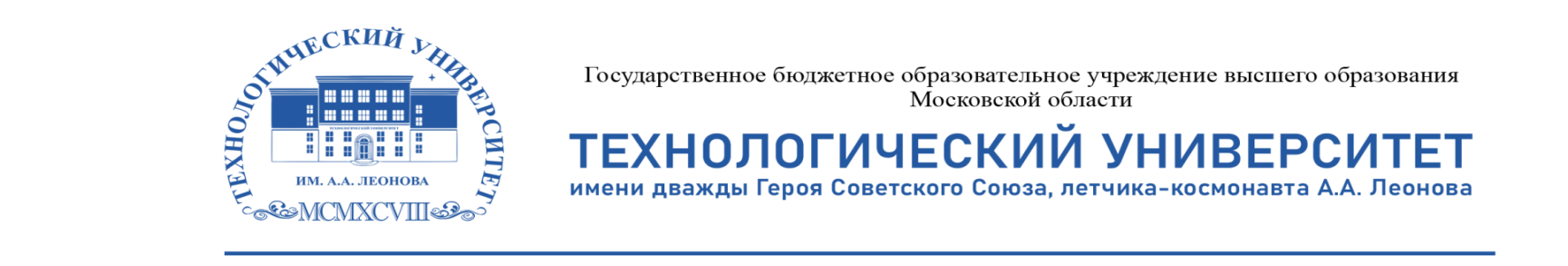
****

Колледж космического машиностроения и технологии

**Курсовой проект**

**Разработка системы классов для приложения «СКЛАД»**

Пояснительная записка

КП.09.02.03.22.016ПЗ

Обучающийся группы П2-19 Подъяблонский В. А.

Руководитель курсового проекта Гусятинер Л. Б.

Результат защиты\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Королев, 2022 г.

**ОГЛАВЛЕНИЕ**

[**1. Введение 4**](#_Toc107772846)

[**2. Теоретическая часть 5**](#_Toc107772847)

[**2.1. Описание предметной области 5**](#_Toc107772848)

[**2.2. Описание существующих разработок 7**](#_Toc107772849)

[**3. Теоретическая часть 15**](#_Toc107772850)

[**3.1. Построение диаграммы прецедентов 15**](#_Toc107772851)

[**3.2. Выбор инструментов 16**](#_Toc107772852)

[**3.2.1. Язык программирования 16**](#_Toc107772853)

[**3.2.2. Среда разработки 17**](#_Toc107772854)

[**3.2.3. База данных 17**](#_Toc107772855)

[**3.2.4. Графический интерфейс 17**](#_Toc107772856)

[**3.3. Проектирование сценария 18**](#_Toc107772857)

[**3.4. Построение диаграммы классов 19**](#_Toc107772858)

[**3.5. Описание главного модуля 22**](#_Toc107772859)

[**3.6. Описание спецификаций к модулям 23**](#_Toc107772860)

[**3.7. Описание модулей 24**](#_Toc107772861)

[**3.8. Описание текстовых наборов модулей 25**](#_Toc107772862)

[**3.9. Описание применения средств откладки 26**](#_Toc107772863)

[**4. Эксплуатационная часть 27**](#_Toc107772864)

[**4.1. Руководство оператора 27**](#_Toc107772865)

[**4.1.1. Назначение программы 27**](#_Toc107772866)

[**4.1.2. Условия запуска программы 27**](#_Toc107772867)

[**4.1.3. Выполнение программы 27**](#_Toc107772868)

[**4.1.4. Сообщение оператору 28**](#_Toc107772869)

[**4.2. To-Do лист 29**](#_Toc107772870)

[**5. Заключение 30**](#_Toc107772871)

[**6. Список использованной литературы и интернет-ресурсов 31**](#_Toc107772872)

[**7. Приложения 32**](#_Toc107772873)

[**7.1. Приложение 1 32**](#_Toc107772874)

[**7.2. Приложение 2 32**](#_Toc107772875)

[**7.3. Приложение 3 32**](#_Toc107772876)

[**7.4. Приложение 4 33**](#_Toc107772877)

[**7.5. Приложение 5 33**](#_Toc107772878)

[**7.6. Приложение 6 33**](#_Toc107772879)

[**7.7. Приложение 7 34**](#_Toc107772880)

[**7.8. Приложение 8 34**](#_Toc107772881)

[**7.9. Приложение 9 35**](#_Toc107772882)

[**7.10. Приложение 10 35**](#_Toc107772883)

# **Введение**

Данный курсовой проект заключается в разработке системы классов для приложения «Склад». Разработка приложения будет происходить на языке программирования Python и фреймворка PyQt.

В первой части будут рассмотрены предметная область и существующие продукты по данной теме.

Во второй части будут описаны инструменты разработки программы и модули программы.

В третьей части описаны взаимодействие пользователя с программой и сообщения оператору.

В заключительной части будут приведены общие выводы по проекту.

# **Теоретическая часть**

## **Описание предметной области**

**Настольное приложение** – это программа, которая работает на компьютере пользователя. Она функционирует на стороне клиента и запускается в виде обыкновенного исполняемого файла.

Настольное приложение реализует следующие возможности:

1. Работа приложения как с подключением, так и без подключения к интернету;
2. Приложение запускается с загрузкой данных из памяти устройства, на котором установлено, не требует каждый раз загрузки параметров из сети;
3. Максимальная эффективность работы за счет возможности многопроцессорных систем;

**Склад** – территория, помещение, предназначенное для хранения материальных ценностей и оказания складских услуг. Склады используются производителями, импортёрами, экспортерами, оптовыми торговцами, транспортными предприятиями, таможней и т. д.

В логистике склад выполняет функцию аккумулирования резервов материальных ресурсов, необходимых для демпфирования колебаний объёмов поставок и спроса, а также синхронизации скоростей потоков товаров в системах продвижения от изготовителей к потребителям или потоков материалов в технологических производственных системах.

**Основные разновидности складов**

По назначению различают следующие виды складов:

* Производственные – склады сырья, комплектующих, материалов, заводские и цеховые склады готовой продукции;
* Таможенные – склады временного хранения товаров в ожидании таможенной очистки;
* Коммерческие – склады, обслуживающие любых владельцев товаров;
* Розничные – склады торговых предприятий;
* Розничные – склады торговых предприятий;
* Военные – склады с разнообразным содержимым, необходимым для снабжения вооруженных сил.

**Основные принципы организации складских услуг**

В целом комплекс складских услуг представляет собой следующую последовательность:

* Разгрузка и погрузка транспорта (выполнение погрузочно-разгрузочных работ);
* Приемка товара (приемка поступивших грузов по количеству и по качеству. Приемка товара – начальная операция, связанная с движением товара на складе и возникновением материальной ответственности);
* Размещение на хранение (укладка товаров в стеллажи, штабели);
* Отбор товаров из мест хранения, подготовка к отпуску: упаковка, окантовка, маркировка и так далее);
* Внутри складское перемещение грузов.

## **Описание существующих разработок**

1. Система «LiteBox» [1] – облачный сервис, предназначен для ведения торгового и финансового учета представителями малого бизнеса. Функционал программы представлен шестью направлениями:

* Управление торговлей (рисунок 1);
* Складской учет (рисунок 2);
* Аналитические отчеты;
* Управление закупками;
* Документы;
* Маркетинговые инструменты.

Фактически программа учета «LiteBox» обладает типичными для своего класса функционалом: она позволяет вести торговлю и анализировать продажи.

Программная база находится в облаке, что позволяет работать с ней в любом месте, где есть интернет.

К плюсам системы «LiteBox» можно отнести:

1. Мультиплатформенность: программой можно пользоваться на компьютере, планшете, ноутбуке;
2. Доступные для малого бизнеса тарифный планы, в том числе довольно функциональный бесплатный тариф;
3. Наличие полнофункциональной демоверсии, позволяющей 14 дней пользоваться всеми возможностями программы;
4. Наличие телефона круглосуточной технической поддержки;
5. Функциональный складской учет, включающий, в том числе резервирование товаров и перемещение ТМЦ между магазинами и складами;
6. Поддержка формирования первичной учетной документации (форма ТОРГ-12, ТОРГ-16 и другие);
7. Облачные данные резервируются в современных DATA-центрах с сертификатом безопасности Tier3.

Минусы системы» «LiteBox»:

1. Отсутствие поддержки продаж через интернет-магазины;
2. Запутанный интерфейс, требующий обучения и длительного привыкания;
3. Сложность первоначальной самостоятельной настройки программы;
4. Поддержка индивидуальных дисконтных карт не подкреплена наличием клиентской базы. Карточки не привязываются к ФИО или номеру телефона;
5. Отсутствует возможность доработки программ под конкретного клиента;
6. Невозможность добавления произвольных характеристик в товарной карточке;
7. Отсутствие поддержки ОС семейства Mac.

Изображение выглядит как текст

Автоматически созданное описание

Рис.1. Управление торговлей

Изображение выглядит как текст

Автоматически созданное описание

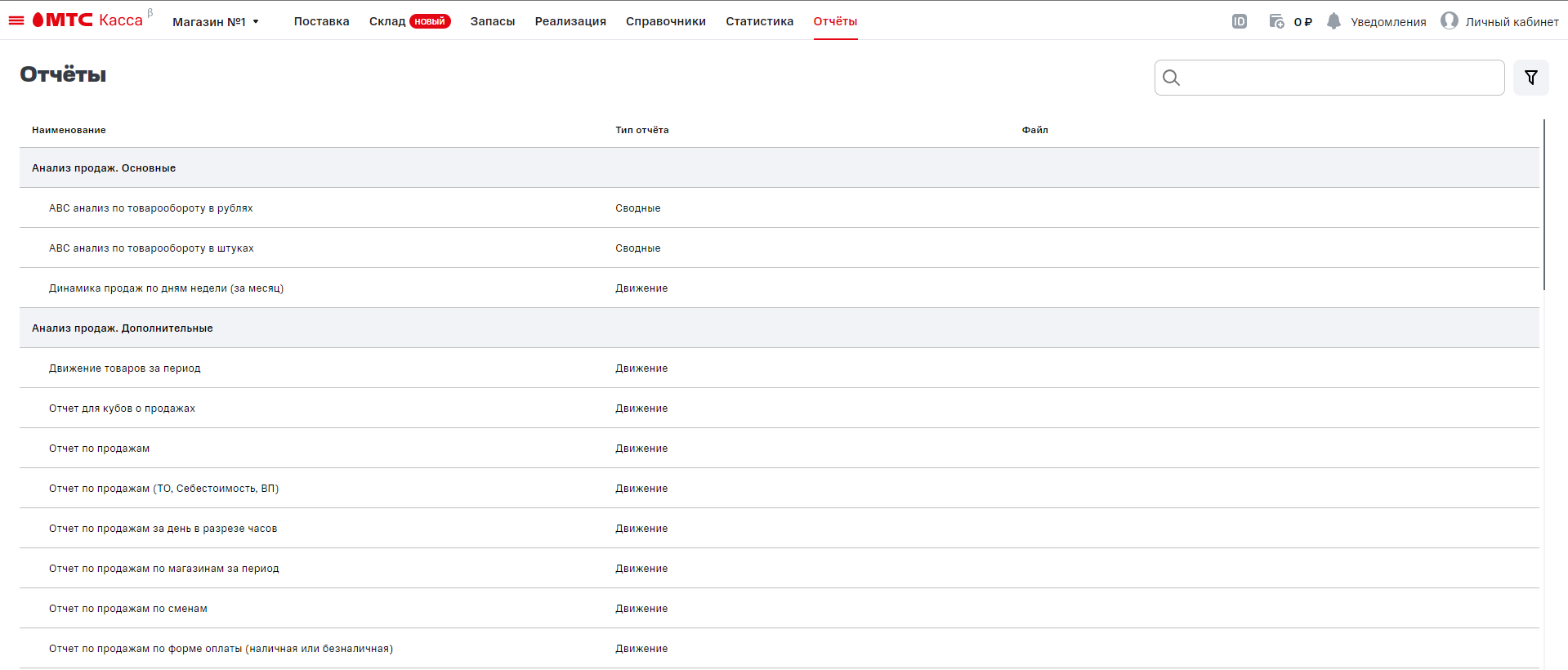
Рис.2. Складской учет 

Рис.3. Аналитический отчеты

1. Система «1С: Торговля и склад» [2] – предназначена для учета любых видом торговых операций. Благодаря гибкости и настраиваемой, система способна выполнять все функции учета – от ведения справочников и ввода первичных документов до получения различных ведомостей и аналитических отчетов.

В системе «1С: Торговля и склад» вы можете:

* Задать для каждого товара необходимое количество цен разного типа, хранить цены поставщиков, автоматически контролировать и оперативно изменять уровень цен;
* Работать со взаимосвязанными документами;
* Выполнять автоматический расчет цен списания товаров;
* Быстро вносить изменения с помощью групповых обработок справочников и документов;
* Ввести учет товаров в различных единицах измерения (рисунок 5);
* Получать самую разнообразную отчетную и аналитическую информацию о движении товаров и денег (рисунок 4);
* Автоматически формировать бухгалтерские проводки для «1С: Бухгалтерии».

Плюсы программы «1C: Торговля и склад»:

1. Наличие функционала для полноценного бухгалтерского, налогового и складского учета;
2. Интеграция с любым кассовым и торговым оборудованием;
3. Автоматическое формирование всех унифицированных торговых документов;
4. Консолидированный учет в нескольких торговых точках;
5. Высокая стабильность работы;
6. Возможность подстройки меню и функционала под конкретного клиента.

Минусы программы «1С: Торговля и склад»:

1. Высокая стоимость;
2. Сложность обучения новых сотрудников;
3. Длительный период внедрения и настройки;
4. Необходимость постоянного обновления ПО;
5. Отсутствие круглосуточной поддержки из-за её предоставления сторонними организациями;
6. Отсутствие CMR–системы;

Изображение выглядит как стол

Автоматически созданное описание

Рис.4. Получать самую разнообразную отчетную и аналитическую информацию о движении товаров и денег.

Изображение выглядит как стол

Автоматически созданное описание

Рис.5. Ввести учет товаров в различных единицах измерения.

1. Система «МойСклад» [3] - удобная система управления складом. Складской учет, интеграция оборудованием, печать документов, адресное хранение, операции сборки и разборки.

Возможности системы «МойСклад»:

* Складской учет – приемка, отгрузки, инвентаризации. Работа с размерами, цветами, упаковками, серийными номерами, весовым товаром. Печать документов и аналитика (рисунок 6);
* Обработка заказов – получение уведомлений о новых заказах, назначение менеджеров, статусы заказов, создание отгрузки. Легко сверяйте комплектации при сборе заказов на складе;
* Розничные продажи – для кассира или курьера, розницы, пункта выдачи, разносной торговли – бесплатная кассовая программа под Windows, Linux, Android, IOS;
* Сборка и производство – оформление операции по сборке и разборке, контролирование движения комплектующих, считайте себестоимость и планируйте потребности в материалах;
* Работа с клиентами – соберите в карточке все о клиенте: статистику, документы, задачи, встречи. Звоните и делайте рассылки по SMS и email. Сегментируйте базу, стройте воронки продаж;
* Управление финансами – контролируйте взаиморасчеты с клиентами. Получайте выписки из любого банка, отправляйте и печатайте платежные поручения (рисунок 7).

Плюсы программы «МойСклад»:

1. Широкий функционал, подходящий для розничной, оптовой торговли, сфера общепита и небольших производств;
2. Поддержка дисконтных карт, создание клиентской базы, формирование воронки продаж;
3. Стабильная работа;
4. Дружелюбный интерфейс и легкость освоение программы новыми сотрудниками;
5. Мультиплатформенность: программой можно пользоваться на OC Windows, macOS, Android, Linux, IOS;
6. Открытый API.

Минусы программы «МойСклад»:

1. Стоимость ежемесячной абонентской платы выше среднерыночной;
2. Отсутствие шаблонов для продажи;

Изображение выглядит как текст

Автоматически созданное описание

Рис.6. Создание отгрузки

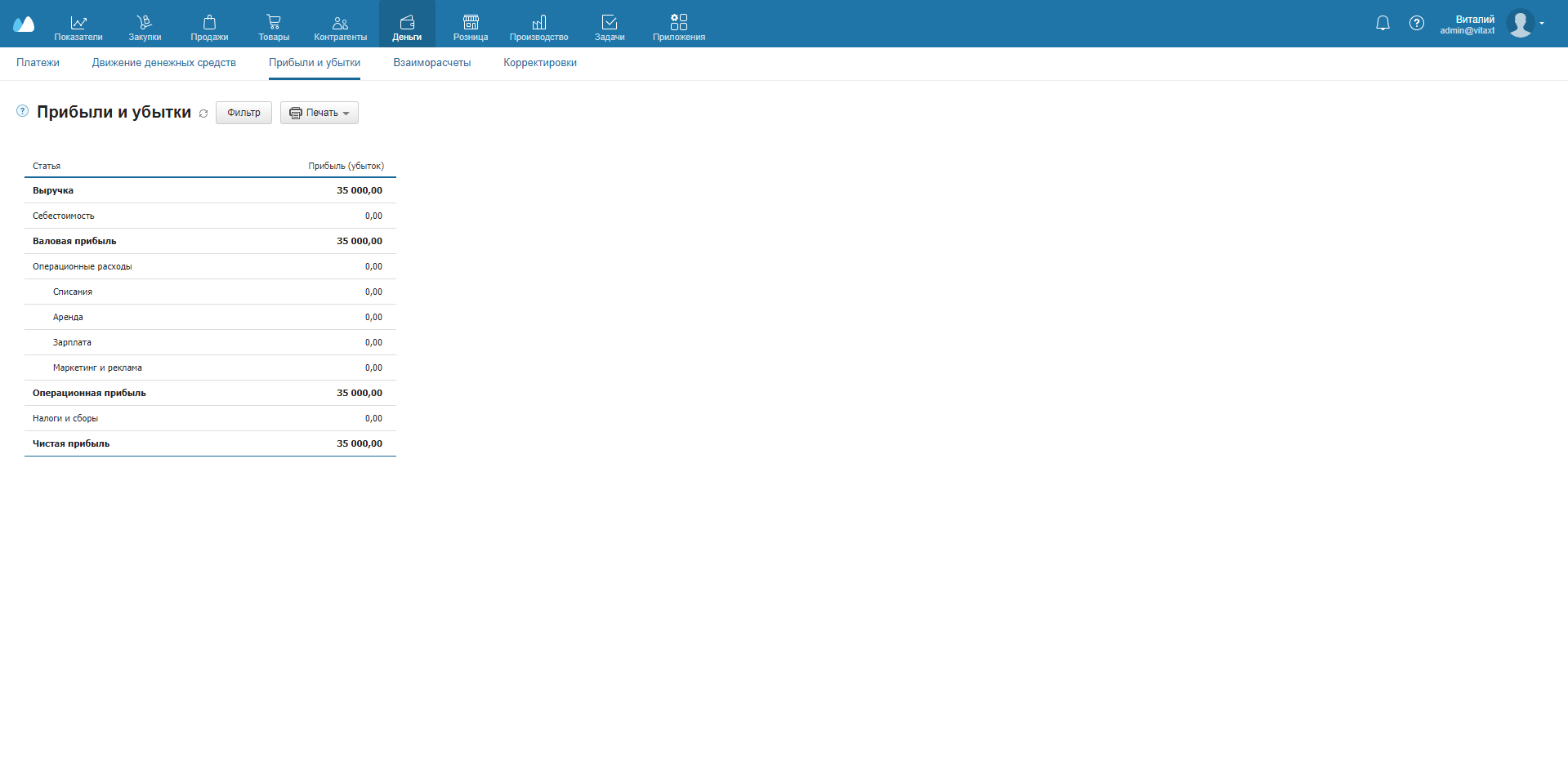


Рис.7. Управление финансами

# **Теоретическая часть**

## **Построение диаграммы прецедентов**

Данный раздел содержит диаграмму прецедентов, на которой отображается визуализация различных типов ролей в системе и то, как эти роли взаимодействуют с системой.

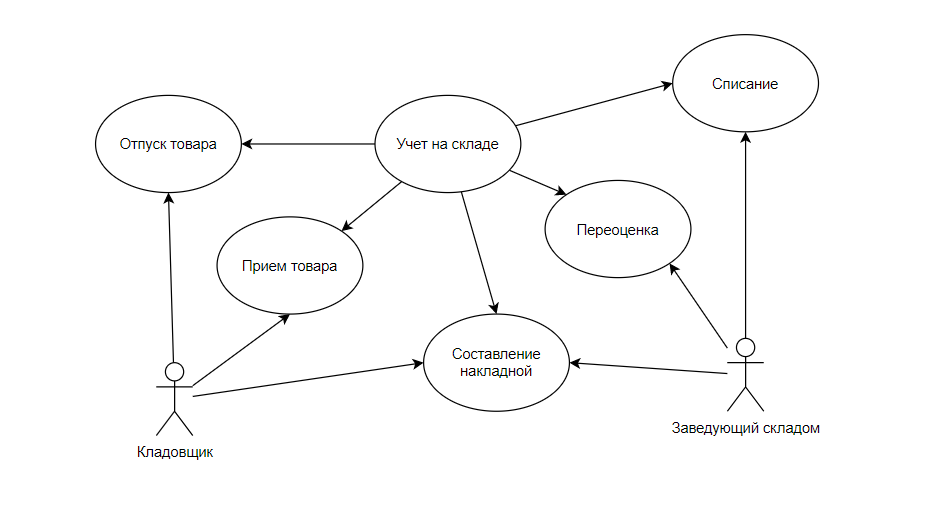


Рис. 8. Диаграмма прецедентов.

## **Выбор инструментов**

### **Язык программирования**

Для написания проекта использовался Python [4] – высокоуровневый язык программирования, к преимуществам которого относят высокую производительность программных решений и структурированный, хорошо читаемый код. Синтаксис максимально облегчен, ядро имеет удобную структуру, а широкий перечень встроенных библиотек позволяет применять внушительный набор полезных функций и возможностей.

Python поддерживает несколько моделей программирования: структурное, объекта-ориентированное, функциональное. В языке присутствует динамическая типизация, автоматическое управление памятью, механизм обработки исключений, поддержка многопоточных вычислений и удобные высокоуровневые структуры данных.

По данным сайта HIGHLOAD [8] за 2022 года, Python является вторым популярности языком программирования, оставив позади себя Java и семейство языков С (рисунок 9).

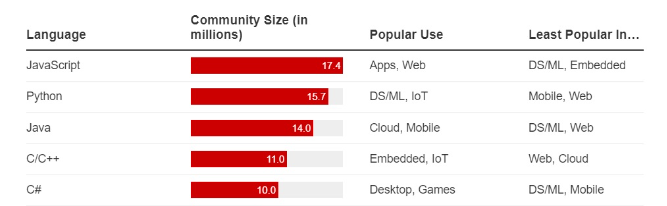


Рис. 9. Таблица «Популярные языки программирования»

### **Среда разработки**

PyCharm [5] – интегрированная кроссплатформенная среда разработки для языка программирования Python предоставляет средства для анализа кода, графический отладчик, инструмент для запуска юнит-тестов и поддерживает веб-разработку на Django. Разработка IDE ведется компанией JetBrains.

PyCharm предоставляется в двух вариантах: платный (PyCharm Professional Edition) и бесплатный (PyCharm Community Edition).

### **База данных**

Уже довольно много лет большинство приложений и сайтов хранят информацию или данные. Для этого они используют базу данных – это упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе. Обычно они управляются системой управления базами данных (СУБД).

В качестве базы данных в проекте была выбрана SQLite [6] – это быстрая и легкая встраиваемая одно файловая СУБД, написанная на языке С, которая не имеет сервера и позволяет хранить всю базу локально на одном компьютере. Для работы SQLite не нужны сторонние библиотеки или службы.

### **Графический интерфейс**

В качестве реализации графического интерфейса в проекте использовался PyQt [7] – это библиотека Python для создания приложения с графическим интерфейсом с помощью инструментария Qt. Библиотека предоставляет из себя набор модулей, содержащих большое количество классов и функций. Разработка ведётся компанией Riverbank Computing.

## **Проектирование сценария**

Данный раздел содержит сценарий проекта – определение сочетаний условий, которые приводят к определенным результатам (рисунок 10).

При запуске приложения открывается главное окно, в котором предоставляется возможность выбора таблицы, добавить элемент, изменить элемент, удалить элемент. вывести в Excel или JSON.

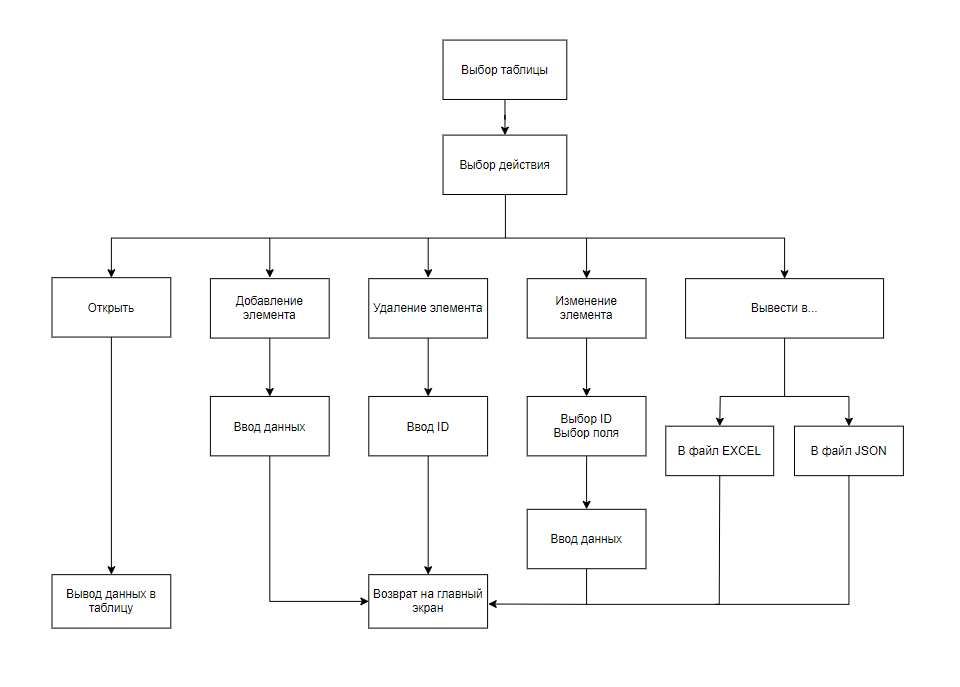


Рис. 10. Сценарий проекта.

## **Построение диаграммы классов**

Данный раздел содержит диаграммы классов, их виды, а также связь между ними.

Подробное описание разделов предоставлены в пунктах: п.3.6 «Описание спецификаций к модулям» и п.3.7 «Описание модулей».

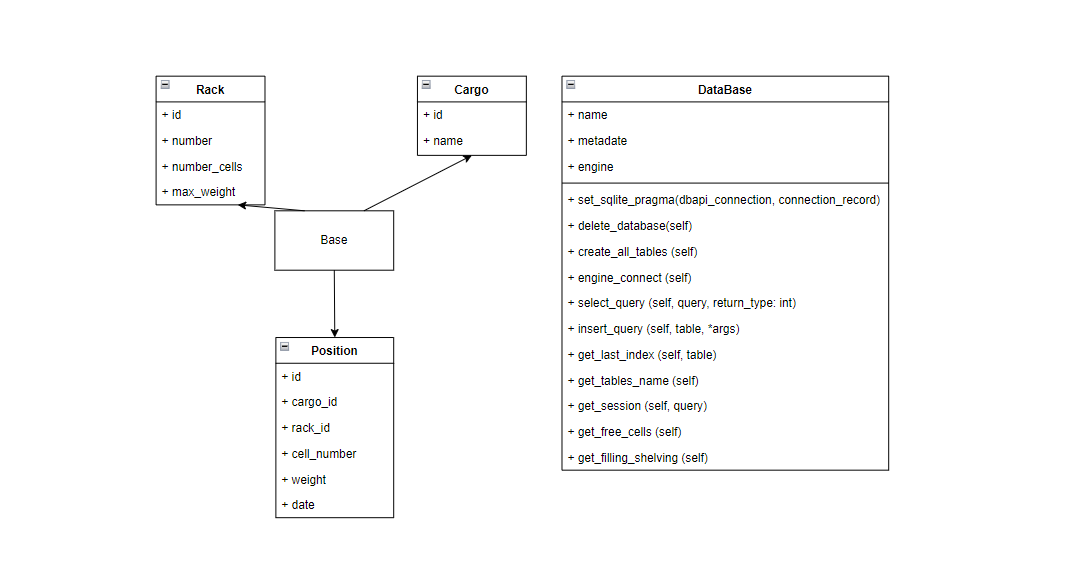


Рис. 11. Диаграмма классов раздел «База данных».

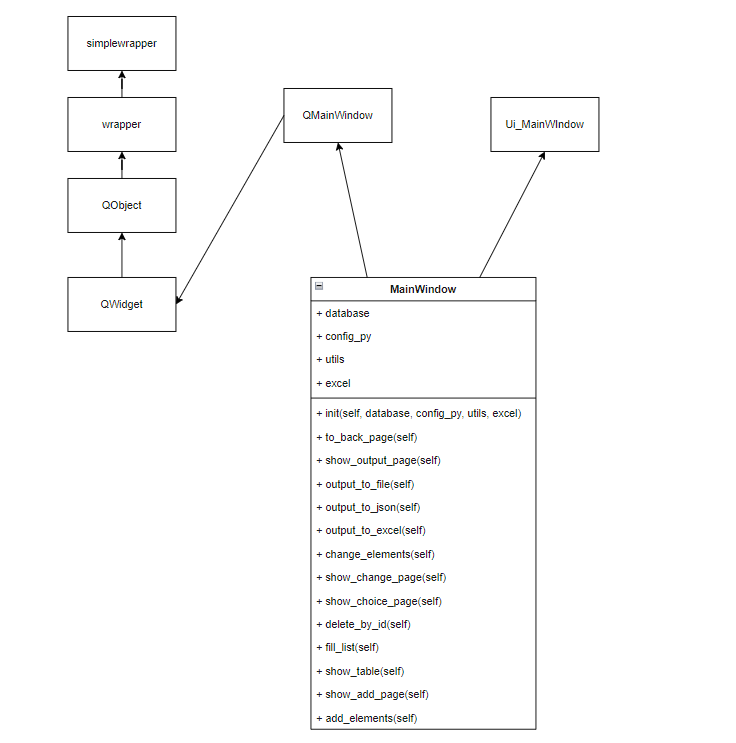


Рис. 12. Диаграмма классов раздел «Графический интерфейс».

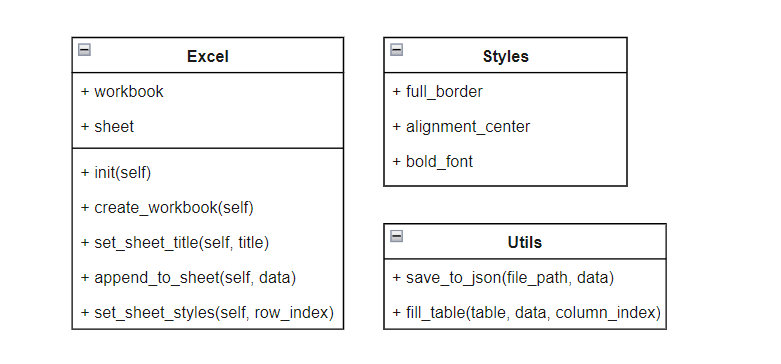


Рис. 13. Диаграмма классов раздел «Утилиты».

Изображение выглядит как текст

Автоматически созданное описание

Рис. 14. Диаграмма классов раздел «Конфигурационные файлы».

## **Описание главного модуля**

Главный модель (листинг 1) состоит из функции main. Данная функция включает в себя объекты необходимые для главного окна и запускает графический интерфейс.

Листинг 1. Главный модуль

import sys  
  
from PyQt5 import QtWidgets  
  
from config.config import Config  
from database.database import DataBase  
from gui.connection.main\_connection import MainWindow  
from utils.excel import Excel  
from utils.utils import Utils  
  
  
def main():  
 config = Config()  
 utils = Utils()  
 excel = Excel()  
  
 database = DataBase("composition")  
 database.create\_all\_tables()  
  
 app = QtWidgets.QApplication(sys.argv)  
 window = MainWindow(database, config, utils, excel)  
 window.show()  
 sys.exit(app.exec\_())  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

## **Описание спецификаций к модулям**

В проекте разработано множество модулей программы, оформленных в виде отдельных файлов.

Реализованные в проекте модули можно разделить на типы:

* База данных (рисунок 11):
* Описание моделей данных (models);
* Связь с базой данных (database);
* Графический интерфейс (рисунок 12):
* Работа главного окна (main\_connection);
* Вспомогательные (рисунок 13):
* Работа с выводом в Excel (excel);
* Работа со стилем в Excel (styles\_excel);
* Работа с сохранением в JSON (utils).
* Константные файлы (рисунок 14):
* Хранение списков и словарей (config);

Взаимодействие между модулями можно увидеть по диаграмме классов п.3.4 «Построение диаграммы классов».

## **Описание модулей**

* Тип «База данных» (рисунок 11) полностью отвечает за создание и получение данных из базы, который включает в себя два модуля:
* Модуль описание моделей базы «models» содержит описание всех моделей таблиц. Под описанием понимается: наименование таблиц, полей и настройки полей;
* Модуль связь с базой данных «database» содержит реализацию работы с базой данных: подключение к базе, создание таблиц, запросы.
* Тип «Графический интерфейс» (рисунок 12) полностью отвечает за визуализацию приложения и включает в себя один модуль:
* Модуль работы главного окна «main\_connection». Он инициализирует окна и страницы приложения. Реализует логику взаимодействия пользователя с интерфейсом.
* Тип «Вспомогательные» (рисунок 13) полностью отвечает за общие вспомогательные методы:
* Модуль «excel» отвечает за работу с одноименной программой;
* Модуль «styles\_excel» отвечает за стили для формирования текста в «MS Excel»;
* Модуль «utils» отвечает за методы, в равной степени используемые во всей программе в целом: вывод в файл c расширением json и т. д.
* Тип «Константные файлы» (рисунок 14) включает в себя один модуль:
* Модуль «config» хранит списки и словари, необходимые в программе.

## **Описание текстовых наборов модулей**

В проекте присутствует обработка ошибок, чтобы сообщить пользователю о проблеме и огородить его от сбоев программы и случайному завершению работы.

* Проверка на открытый файл (листинг 2).

В случае перезаписывания открытого файла excel, отобразится уведомление о соответствующей ошибке в отдельном окне (приложение 10).

Листинг 2. Проверка на открытый файл.

try:  
 self.excel.workbook.save(filename=file\_path)  
 self.to\_back\_page()  
except PermissionError:  
 QMessageBox.warning(self, "ОШИБКА", "Закройте выбранный файл")

## **Описание применения средств откладки**

При написании проекта использовалось огромное количество раз встроенное средство откладки в IDE «PyCharm». В результате чего было исправлено большое количество ошибок различных видов.

При запуске программы без использования средства откладки в нештатных ситуациях не позволяет определить и диагностировать проблему: текст ошибки, стек вызовов с именами функций и номерами строк. Данную проблему решал запуск программы в режиме отладки, где повторное действие приводило к той же ошибке, но среда разработки при завершении программы показывала саму ошибку и ее суть.

# **Эксплуатационная часть**

## **Руководство оператора**

### **Назначение программы**

Разработка настольного приложения для сокращения времени просмотра базы данных содержащие сведение о грузах и размещении на стеллажах.

Дополнительные возможности программы:

* Добавление новых элементов;
* Изменение элементов;
* Удаление элементов;
* Вывод в Excel или JSON:
* Процент наполнения зала по сеансам;
* Самый популярный жанр.

### **Условия запуска программы**

Запуск программы с помощью файла с расширением .exe:

1. Стабильно работающее устройство на OC Windows 10.

Запуск программы с помощью исходного кода:

1. Стабильно работающее устройство на OC Windows 10;
2. Python версии 3.10 и выше;
3. Установленные библиотеки, указанные в файле «requirements.txt»

### **Выполнение программы**

При запуске программы появляется окно «Склад» (приложение 1), с помощью которого можно: просмотреть любую таблицу в базе данных (приложение 2), добавить элемент в таблицу, изменить элемент в таблице, удалить элемент в таблице и вывести в Excel или JSON файл.

Страница «Добавить элемент» (приложение 3, 4, 5) зависит от открытой таблицы, позволяет добавить новый элемент в базу данных.

Страница «Изменить элемент 1» (приложение 6) зависит от открытой таблицы, позволяет выбрать ID и поле, которые мы хотим изменить.

Страница «Изменить элемент 2» (приложение 7) показывает изменяемый ID и изменяемый поле, позволяет ввести новое значение.

Страница «Удалить элемент» (приложение 8) зависит от открытой таблицы, отображает ID, которые можно удалить.

Страница «Вывести в...» (приложение 9), предоставляется выбор вывода: процент наполнения зала по сеансам, самый популярный жанр и таблицу, в файл Excel или JSON.

### **Сообщение оператору**

В случае вывода в открытый файл Excel появится соответствующее сообщение об ошибке (приложение 10).

## **To-Do лист**

Данный проект находится на стадии разработки. Планируется исправление множества ошибок и добавления нового функционала:

1. Просмотр сразу нескольких таблиц;
2. Исправление ошибок с масштабированием;
3. Исправления мелких ошибок;
4. Добавление проверок при «Добавление элемента»;
5. Пересмотр дизайна основного окна.

# **Заключение**

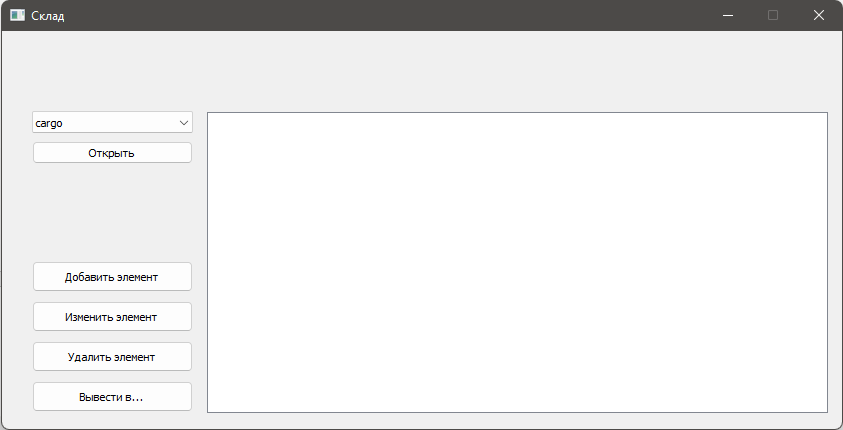
Данный проект предоставляет минимальный, но основной функционал для управления системы складского учета. В случае конечной реализации всех задумок, может быть внедрен в работу предприятий связанный с управлением склада, так как станет доступным аналогом существующих разработок.

# **Список использованной литературы и интернет-ресурсов**

1. Система «LiteBox»: <https://litebox.ru/>
2. Система «1С: Торговля и склад»: <https://1c.ru/rus/products/1c/predpr/torg77.htm>
3. Система «МойСклад»: <https://www.moysklad.ru/>
4. Язык программирования «Python»: <https://www.python.org/>
5. Среда разработки IDE «PyCharm»: <https://www.jetbrains.com/pycharm/>
6. База данных «SQLite»: <https://www.sqlite.org/index.html>
7. Графическая библиотека PyQt: <https://www.riverbankcomputing.com/software/pyqt/>
8. HIGHLOAD: [https://highload.today/](https://highload.today/top-15-yazykov-programmirovaniya-i-tehnologij-s-samym-bolshim-soobshhestvom-razrabotchikov-dannye-za-2022-god/)
9. Github: <https://github.com/BiterVi>

# **Приложения**

## Приложение 1

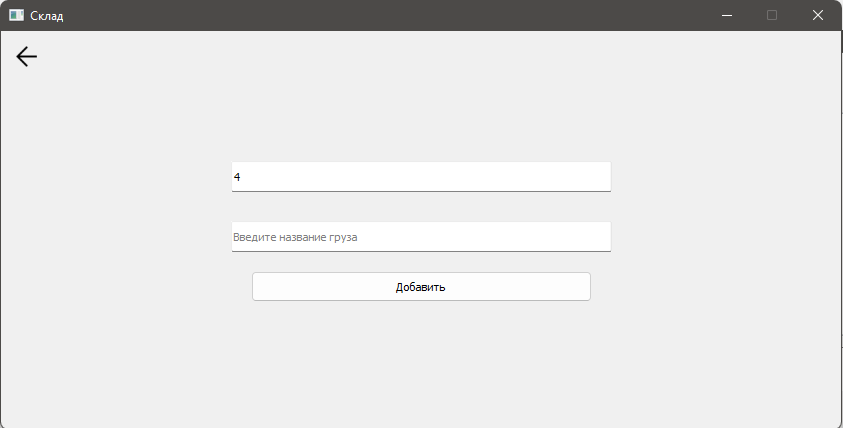


## Приложение 2

Изображение выглядит как текст

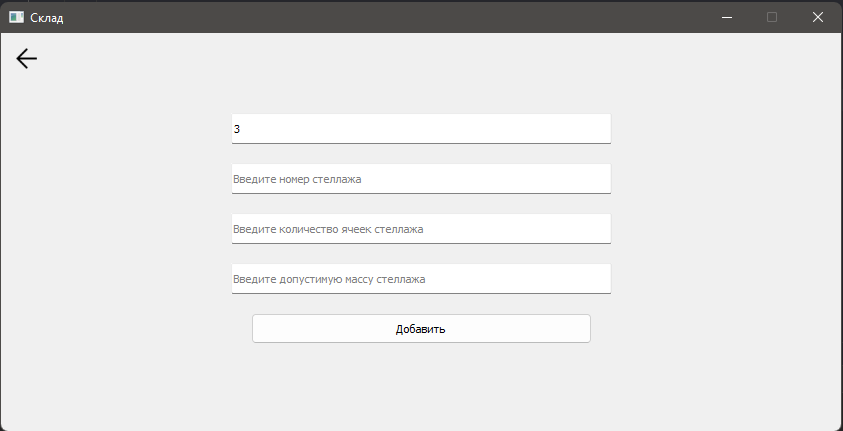
Автоматически созданное описание

## Приложение 3

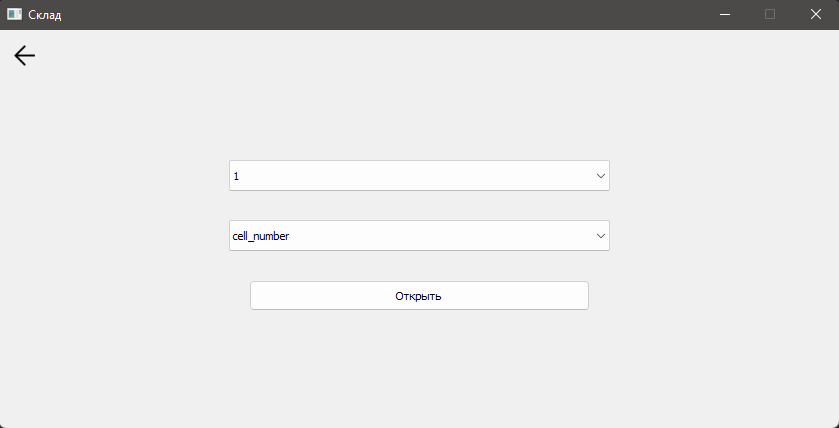


## Приложение 4

## Приложение 5



## Приложение 6

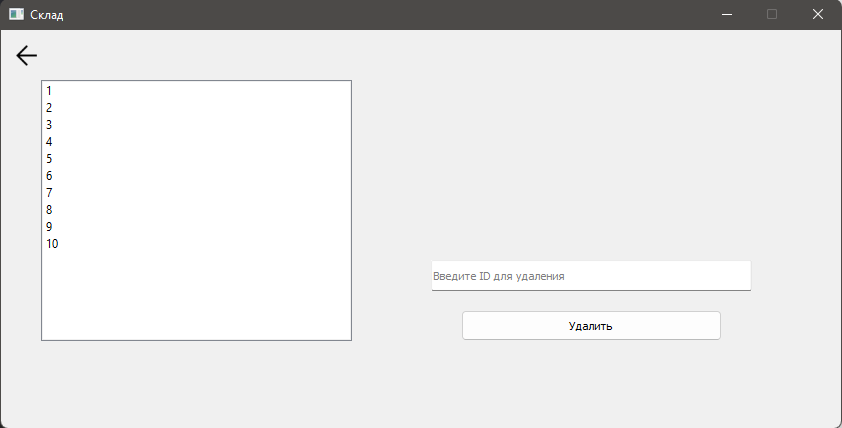


## Приложение 7

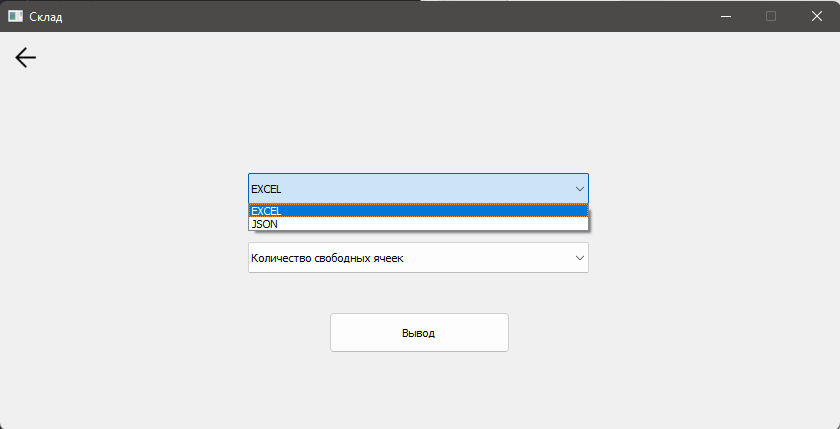
Изображение выглядит как текст

Автоматически созданное описание

## Приложение 8



## Приложение 9



## Приложение 10

Изображение выглядит как текст

Автоматически созданное описание