МИНИСТЕРСТВО ОБРАЗОВАНИЯ КИРОВСКОЙ ОБЛАСТИ

Кировское областное государственное профессиональное образовательное бюджетное учреждение

"Слободской колледж педагогики и социальных отношений"

**КУРСОВОЙ ПРОЕКТ**

по ПМ 01 «Разработка программных модулей» на тему:

**Разработка программного модуля для автоматизации учёта книг в библиотеке**

Выполнил: Збруев Антон Владимирович

Специальность 09.02.07

Информационные системы и программирование

Группа 21П-1

Форма обучения: очная

Руководитель: Махнев Александр Анатольевич

Дата защиты курсовой работы:

Председатель ПЦК:

Оценка за защиту курсовой работы:

Слободской

2024

**ОГЛАВЛЕНИЕ**

[ВВЕДЕНИЕ 3](#_Toc181210248)

[АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 6](#_Toc181210249)

[РАЗРАБОТКА ТЕХНИЧЕСКОГО ЗАДАНИЯ 13](#_Toc181210250)

[ОПИСАНИЕ АЛГОРИТМОВ И ФУНКЦИОНИРОВАНИЯ ПРОГРАММЫ 17](#_Toc181210251)

[ТЕСТИРОВАНИЕ ПРОГРАММНОГО МОДУЛЯ 23](#_Toc181210252)

[РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 26](#_Toc181210253)

[ЗАКЛЮЧЕНИЕ. 30](#_Toc181210254)

[СПИСОК ЛИТЕРАТУРЫ 31](#_Toc181210255)

[ПРИЛОЖЕНИЕ 33](#_Toc181210256)

ВВЕДЕНИЕ

Библиотеки, на протяжении веков являясь неотъемлемой частью культурной и научной жизни общества, претерпевают значительные изменения в эпоху цифровизации. Роль библиотек сегодня не ограничивается хранением и предоставлением доступа к книжным фондам. Они превращаются в многофункциональные информационные центры, предлагающие широкий спектр услуг, включая доступ к электронным ресурсам, организацию образовательных и культурных мероприятий, поддержку научных исследований. Однако, несмотря на расширение сферы деятельности, основой любой библиотеки остается ее фонд – собрание книг, журналов, газет и других печатных изданий. Эффективное управление библиотечным фондом, включающим тысячи, а иногда и миллионы экземпляров, представляет собой сложную и многогранную задачу. Традиционные методы учета, основанные на ручном ведении карточек, формуляров и журналов, становятся все менее эффективными в условиях растущих объемов информации и необходимости оперативного доступа к данным. Ручной учет книг – это трудоемкий процесс, подверженный ошибкам, который не позволяет в полной мере реализовать потенциал библиотеки как информационного центра. Развитие информационных технологий открывает новые возможности для автоматизации библиотечных процессов, что подтверждается работами исследователей в области библиотековедения и информационных технологий [3, 7, 15].

Автоматизация библиотечных процессов с помощью специализированного программного обеспечения открывает перед библиотеками широкие возможности для повышения эффективности своей работы. Внедрение таких систем позволяет:

* значительно упростить и ускорить процессы закупки, прихода, списания и учета книг, освободив сотрудников от рутинных операций;
* минимизировать риск ошибок, связанных с ручным вводом данных, обеспечив высокую точность учета библиотечного фонда;
* улучшить контроль за движением книг, отслеживая их местонахождение и историю перемещений;
* оперативно формировать отчеты о состоянии фонда, динамике его изменения, источниках пополнения, что позволяет принимать обоснованные управленческие решения.

Цель данной курсовой работы – разработка программного модуля для автоматизации учета книг в библиотеке, ориентированного на потребности небольших библиотек с ограниченным бюджетом. Модуль призван обеспечить эффективное управление библиотечным фондом, упростить учетные операции и повысить качество обслуживания читателей (в части учета библиотечного фонда).

**Объект исследования:**

* Процессы учета библиотечного фонда в библиотеке, включая закупку, приход, списание и инвентаризацию книг.

**Предмет исследования:**

* Программный модуль для автоматизации учета библиотечного фонда, реализованный на платформе 1С: Предприятие.

**Методы исследования:**

* Системный анализ.
* Функциональное моделирование.
* Объектно-ориентированное программирование.

В работе использовалась платформа 1С: Предприятие, которая широко распространена в России и обладает мощными возможностями для создания бизнес-приложений. Выбор данной платформы обусловлен ее доступностью, гибкостью и наличием широкого спектра инструментов для разработки учетных систем.

Информационную базу исследования составили официальные нормативно-правовые источники, регулирующие деятельность библиотек, научные статьи и публикации, посвященные проблемам автоматизации библиотек, методические рекомендации по учету библиотечных фондов, официальная документация по платформе 1С: Предприятие.

АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

**1. Общая характеристика предметной области**

Библиотека – это не просто хранилище книг, это важный социальный институт, осуществляющий сбор, систематизацию, хранение и предоставление в пользование литературы и других информационных материалов. Библиотеки играют ключевую роль в образовании, науке, культуре и информационном обеспечении общества.

Ключевой ресурс любой библиотеки – ее фонд. От эффективности управления фондом зависит успешность выполнения библиотекой своих функций и удовлетворение потребностей пользователей.

**2. Процессы учета книг в библиотеке**

Учет библиотечного фонда включает в себя комплекс взаимосвязанных процессов, направленных на обеспечение сохранности книг, контроль за их движением и эффективное использование.

* **Закупка книг:**
  + **Планирование закупок:** Процесс закупки начинается с анализа состава фонда, выявления недостающих изданий и определения потребностей пользователей. На основании этого анализа составляется план закупок, который утверждается заведующим библиотекой. [9, 14]
  + **Выбор поставщиков:** Выбор поставщиков книг осуществляется с учетом ценовой политики, ассортимента, сроков поставки и надежности компаний.
  + **Оформление заказа:** Заказ на книги оформляется в виде документа, содержащего перечень необходимых изданий, их количество, цену и условия поставки.
  + **Приемка книг:** При поступлении книг от поставщика проводится их приемка, в ходе которой проверяется соответствие поставки заказу, а также качество и комплектность изданий. Результаты приемки фиксируются в акте приема-передачи.
* **Приход книг:**
  + **Регистрация:** Каждая поступившая в библиотеку книга регистрируется в инвентарной книге и получает уникальный инвентарный номер. В каталожную карточку книги заносятся ее библиографические данные (название, автор, год издания, издательство и т.д.), а также информация об источнике получения.
  + **Инвентаризация:** Инвентаризация библиотечного фонда проводится периодически с целью выявления недостач, излишков и уточнения состояния книг. В ходе инвентаризации проводится сличение фактического наличия книг с данными учета.

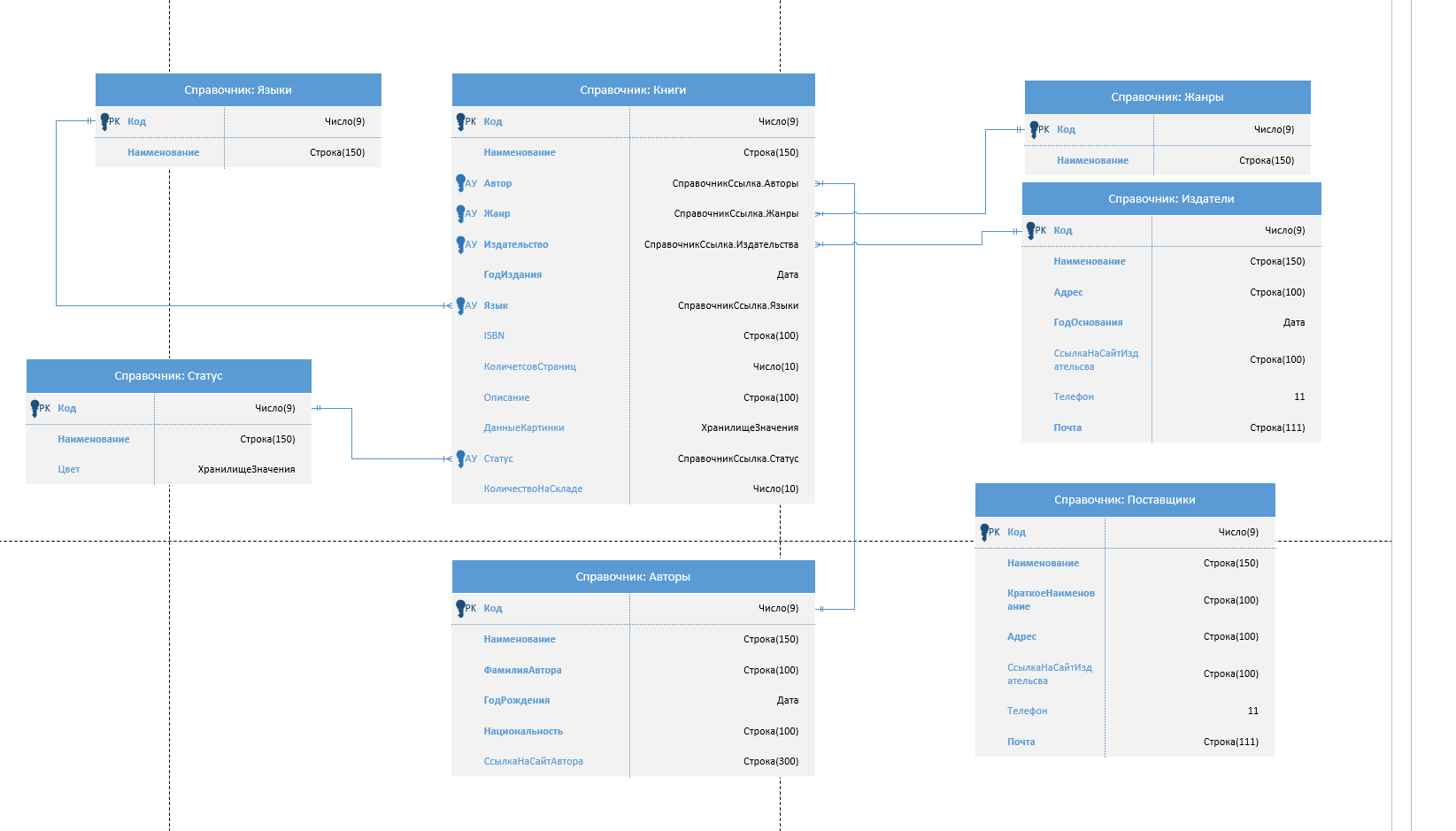


Рисунок 1 - ERD-Диаграмма справочников

* **Списание книг:**
  + **Причины списания:** Книги могут быть списаны из библиотеки по различным причинам: физический износ, моральное устаревание, утрата, порча, дублирование экземпляров.
  + **Процедура списания:** Списание книг осуществляется на основании решения специально созданной комиссии. Оформляется акт о списании, в котором указываются сведения о списываемых книгах и причина списания.
* **Учет движения книг:**
  + Движение книг отражает все изменения в составе библиотечного фонда: поступление, списание, перемещение внутри библиотеки, выдача и возврат книг читателями (если применимо).
  + Для учета движения книг используются различные журналы и карточки, в которых фиксируются сведения о книге, дате и причине движения.

**3. Существующие методы учета книг**

* **Традиционный (ручной) учет:** Основан на использовании бумажных носителей информации: карточек, формуляров, журналов. Этот метод является наиболее распространенным в небольших библиотеках с ограниченным фондом [10].
  + **Преимущества:** Простота и доступность.
  + **Недостатки:** Трудоемкость, высокий риск ошибок, сложность получения оперативной информации и анализа данных.
* **Автоматизированный учет:** Основан на использовании специализированного программного обеспечения. Позволяет вести электронный каталог книг, отслеживать их движение, формировать различную отчетность.
  + **Преимущества:** Повышение эффективности работы, точности учета, возможности анализа данных, улучшение качества обслуживания пользователей [7, 8].
  + **Недостатки:** Необходимость затрат на приобретение и внедрение программного обеспечения, обучение персонала.

**4. Проблемы ручного учета книг**

Ручной учет книг имеет ряд существенных недостатков, которые снижают эффективность работы библиотеки:

* **Трудоемкость и затраты времени:** Ручной ввод данных, поиск информации в карточках и журналах требуют значительных затрат времени сотрудников.
* **Высокий риск ошибок:** Ручной ввод данных подвержен ошибкам, которые могут привести к неточностям в учете и потере информации о книгах.
* **Сложность получения оперативной информации о состоянии фонда:** Поиск необходимой информации в бумажных носителях занимает много времени.
* **Ограниченные возможности для анализа данных:** Ручной учет не позволяет проводить глубокий анализ данных о фонде и его использовании.

**5. Обоснование необходимости автоматизации**

Автоматизация учета книг в библиотеке является необходимым условием для повышения эффективности ее работы, оптимизации управления библиотечным фондом и повышения качества обслуживания пользователей.

**Преимущества автоматизации:**

* **Повышение эффективности работы библиотеки:** Автоматизация учетных операций позволяет сократить время, затрачиваемое на рутинные процессы, и освободить сотрудников для выполнения более творческих задач.
* **Улучшение качества обслуживания пользователей:** Автоматизированный учет позволяет оперативно предоставлять пользователям информацию о наличии книг, их местонахождении, сроках выдачи и т.д.
* **Оптимизация управления библиотечным фондом:** Автоматизированная система позволяет проводить анализ данных о фонде, отслеживать его движение, планировать закупки и списания.
* **Снижение затрат времени и ресурсов:** Автоматизация учетных операций позволяет сократить затраты на бумажные носители, хранение и обработку информации.

**Обзор аналогов**

**1С: Библиотека** — комплексная программа для автоматизации библиотечных процессов. Она предоставляет широкий набор функций, включая учет фонда, каталогизацию, обслуживание читателей, формирование отчетов.

При работе с книгами программа использует справочники "Книги", "Авторы", "Издательства", а также регистры накопления для учета остатков и движения книг. Для управления закупками предусмотрен документ "Закупка книг" с возможностью выбора поставщика или издателя. Приход книг оформляется документом "Приход книг" с указанием источника получения. Списание книг осуществляется документом "Списание книг" с указанием причины списания. Для ведения информации о ценах используется регистр сведений "Цены".

**Плюсы:**

* Комплексный подход к автоматизации библиотечных процессов.
* Широкие возможности настройки и адаптации.
* Интеграция с другими продуктами 1С.

**Минусы:**

* Высокая стоимость.
* Требует специальных знаний для настройки и администрирования.

**АБИС "ИРБИС"** — автоматизированная библиотечно-информационная система с открытым исходным кодом. Она предоставляет базовый набор функций для учета фонда, каталогизации и поиска книг.

Программа работает с различными форматами данных, включая MARC и RUSMARC. Для хранения информации о книгах используются базы данных, структура которых может быть настроена пользователем.

**Плюсы:**

* Бесплатное программное обеспечение.
* Гибкость в настройке структуры данных.
* Наличие активного сообщества пользователей.

**Минусы:**

* Сложный интерфейс, требующий обучения.
* Ограниченные возможности по сравнению с коммерческими системами.

**6. Роли и ответственность в процессе учета книг:**

В зависимости от размера и структуры библиотеки, ответственность за учет книг может быть распределена по-разному.

В небольших библиотеках все функции учета, от закупки до списания, может выполнять один сотрудник, например, заведующий библиотекой. Это обусловлено ограниченными ресурсами и небольшим объемом фонда.

В крупных библиотеках обычно создаются отдельные структурные подразделения, ответственные за разные этапы учета: отдел комплектования (закупка, прием книг), отдел каталогизации (регистрация, классификация), отдел хранения (учет движения, списание). Такая специализация позволяет более эффективно управлять большими фондами и обеспечивает высокое качество учета.

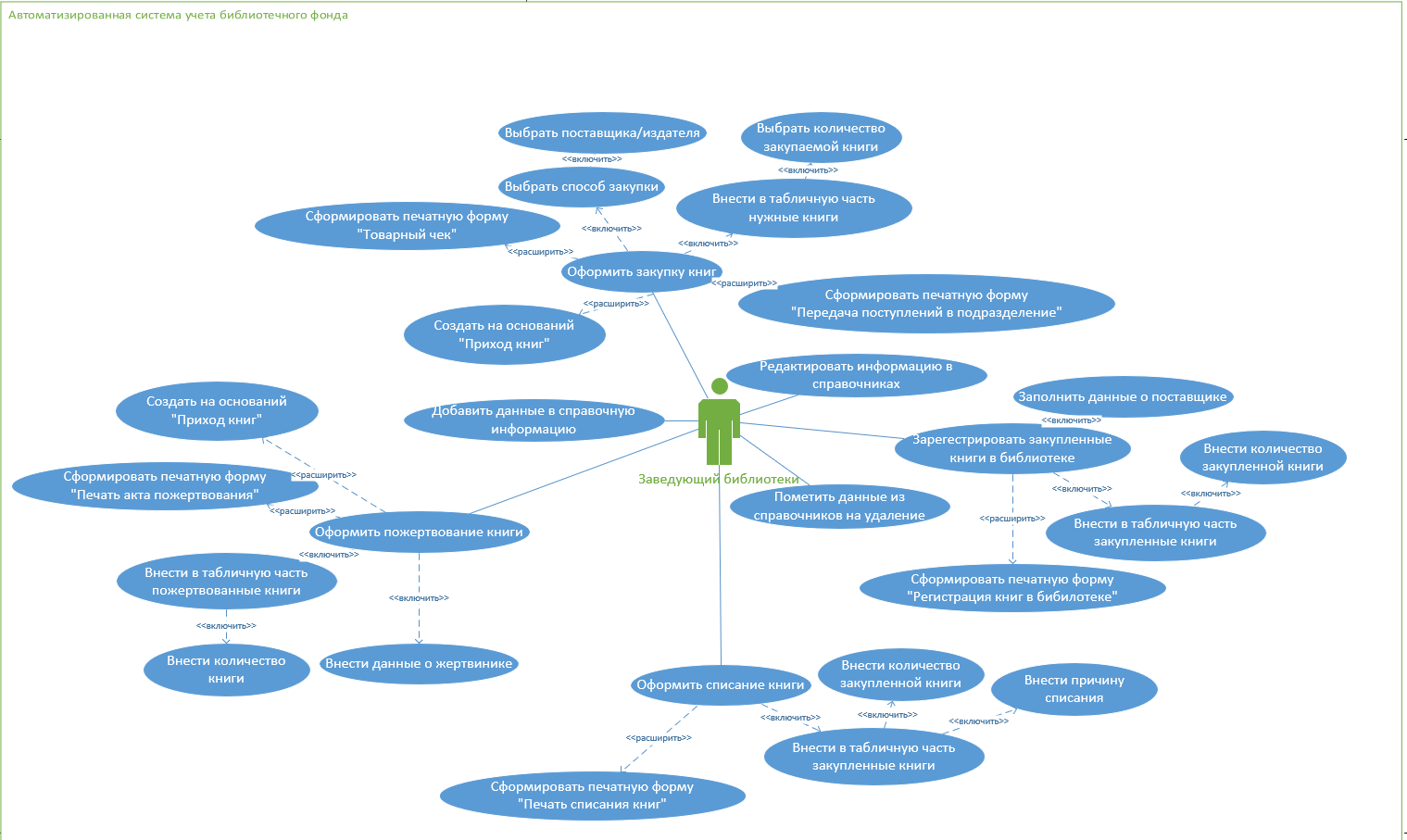


Рисунок 2 - Диаграмма вариантов использования роли "Заведующий библиотеки"

**7. Роль администратора системы:**

Внедрение автоматизированной системы учета предполагает наличие администратора, ответственного за ее настройку, поддержку и обеспечение безопасности.

Технический специалист (администратор): Этот пользователь отвечает за установку и настройку программного модуля, управление пользователями, резервное копирование данных, а также за решение технических проблем. Администратор имеет полный доступ ко всем функциям системы, включая возможность удаления данных из конфигурации. Важно отметить, что удаление данных из конфигурации – операция, требующая особой осторожности, и должна выполняться только квалифицированным специалистом.

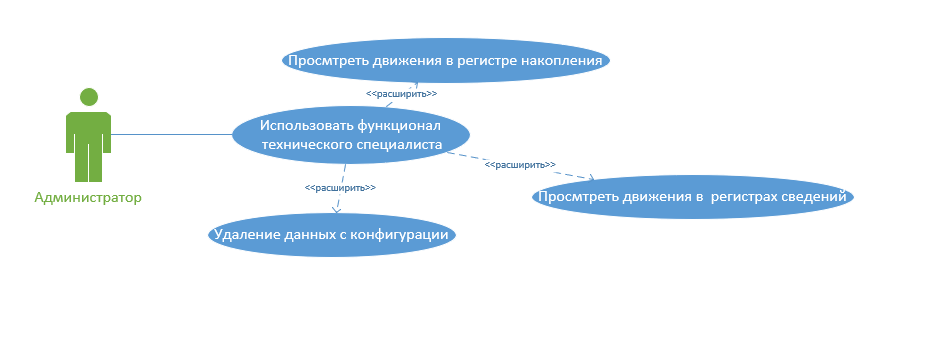


Рисунок 3 - Диаграмма вариантов использования роли "Администратор"

Мой программный модуль предлагает более простой и доступный вариант автоматизации учета библиотечного фонда, ориентированный на небольшие библиотеки и организации.

В данной курсовой работе будет разработан программный модуль для автоматизации учета книг, который позволит решить выявленные проблемы ручного учета и обеспечить эффективное управление библиотечным фондом.

РАЗРАБОТКА ТЕХНИЧЕСКОГО ЗАДАНИЯ

**1. Введение**

В данном разделе представлено техническое задание (ТЗ) на разработку программного модуля для автоматизации учета книг в библиотеке. ТЗ содержит подробное описание требований к модулю, его функциональным возможностям, а также технических характеристик. Техническое задание является основным документом, регламентирующим процесс разработки программного обеспечения и определяющим его соответствие потребностям заказчика.

**2. Назначение и область применения**

* **Назначение:** Разрабатываемый программный модуль предназначен для автоматизации учета книг в небольших библиотеках, предоставляя удобный инструмент для управления библиотечным фондом и оптимизации рабочих процессов.
* **Область применения:** Модуль может быть использован в различных типах библиотек с небольшим объемом фонда, таких как школьные, вузовские, публичные, а также в библиотеках небольших организаций, учебных центров, творческих студий и клубов.

**3. Функциональные требования**

Программный модуль должен обеспечивать выполнение следующих функций:

* **Учет книг:**
  + **Добавление новой книги в базу данных:** Модуль должен позволять добавлять новые книги в базу данных с указанием всех необходимых реквизитов, таких как название, автор, год издания, издательство, жанр, инвентарный номер, цена, количество экземпляров и т.д.
  + **Редактирование данных о книге:** Должна быть предусмотрена возможность изменения данных о книге, если это необходимо.
  + **Удаление книги из базы данных:** в случае утраты или списания книги, модуль должен позволять удалять ее из базы данных.
  + **Поиск книг по различным критериям:** Пользователь должен иметь возможность быстро находить нужные книги по различным критериям, таким как название, автор, жанр, инвентарный номер, ключевые слова и т.д.
* **Закупка книг:**
  + **Формирование заказа на книги:** Модуль должен позволять создавать заказы на книги с указанием поставщика, перечня заказываемых книг, количества экземпляров и цены.
  + **Учет поступления книг по заказу:** При поступлении книг от поставщика модуль должен регистрировать их приход и обновлять информацию о количестве экземпляров в базе данных.
  + **Контроль оплаты заказов:** Модуль должен отслеживать статус оплаты заказов на книги.
* **Приход книг:**
  + **Оформление поступления книг:** Модуль должен позволять регистрировать поступление книг из различных источников, таких как закупка, дар, обмен, с указанием соответствующих документов.
* **Списание книг:**
  + **Оформление списания книг:** Модуль должен позволять регистрировать списание книг с указанием причины списания, даты списания и других необходимых реквизитов.
* **Формирование отчетов:**
  + **Отчет о состоянии библиотечного фонда:** Модуль должен формировать отчет о текущем состоянии библиотечного фонда с указанием количества книг, их стоимости, жанровой структуры фонда и т.д.
  + **Отчет о движении книг за выбранный период:** Модуль должен формировать отчет о движении книг за выбранный период (поступление, списание, перемещение).
  + **Отчет о закупках книг:** Модуль должен формировать отчет о закупках книг за выбранный период с указанием поставщика, перечня книг, количества, цены, статуса оплаты.
  + **Отчет о списании книг:** Модуль должен формировать отчет о списанных книгах за выбранный период с указанием причин списания.

**4. Требования к пользовательскому интерфейсу**

* **Интерфейс программы должен быть интуитивно понятным и удобным в использовании:** Разработка пользовательского интерфейса должна быть ориентирована на простоту и удобство использования, чтобы даже неподготовленный пользователь мог легко освоить работу с модулем.
* **Должна быть предусмотрена система подсказок и справки для пользователей:** Для облегчения работы с модулем должна быть разработана система подсказок, которая будет предоставлять пользователю информацию о функциях программы и способах их использования.
* **Требования** к пользовательскому интерфейсу должны соответствовать принципам эргономичности и удобства использования, изложенным в ГОСТ [1]

**5. Требования к техническому обеспечению**

* **Операционная система:** Windows 7/8/10/11.
* **Платформа разработки:** 1С: Предприятие 8.3.
* **База данных:** Встроенная база данных 1С: Предприятие.

**6. Требования к надежности и безопасности**

* **Устойчивость к ошибкам:** Программа должна быть устойчива к ошибкам пользователей и сбоям в работе системы, обеспечивая целостность данных и предотвращая потерю информации.
* **Резервное копирование:** Должна быть предусмотрена система резервного копирования данных, позволяющая восстановить базу данных в случае сбоя или потери информации.
* **Авторизация и разграничение прав доступа:** Доступ к программе должен быть ограничен системой авторизации, которая будет идентифицировать пользователя и предоставлять ему доступ только к тем функциям, которые соответствуют его роли.

**7. Стадии и этапы разработки**

* **Анализ и проектирование:** 1 месяц.
* **Разработка программного кода:** 2 месяца.
* **Тестирование и отладка:** 1 месяц.
* **Внедрение и сопровождение:** по мере необходимости.

**Вывод:**

Разработанное техническое задание на создание программного модуля для автоматизации учета книг в библиотеке содержит подробное описание функциональных и технических требований к разрабатываемой системе. Данное ТЗ послужит основой для дальнейшей разработки, тестирования и внедрения программного модуля, обеспечивая его соответствие потребностям заказчика и эффективную автоматизацию процессов учета библиотечного фонда.

ОПИСАНИЕ АЛГОРИТМОВ И ФУНКЦИОНИРОВАНИЯ ПРОГРАММЫ

Разработанный программный модуль для автоматизации учета книг в библиотеке построен на платформе 1С: Предприятие 8.3, предоставляя удобный и интуитивно понятный интерфейс для взаимодействия с пользователем. Модуль состоит из совокупности объектов метаданных, включая справочники, документы, регистры, а также программные модули, определяющие логику работы системы.

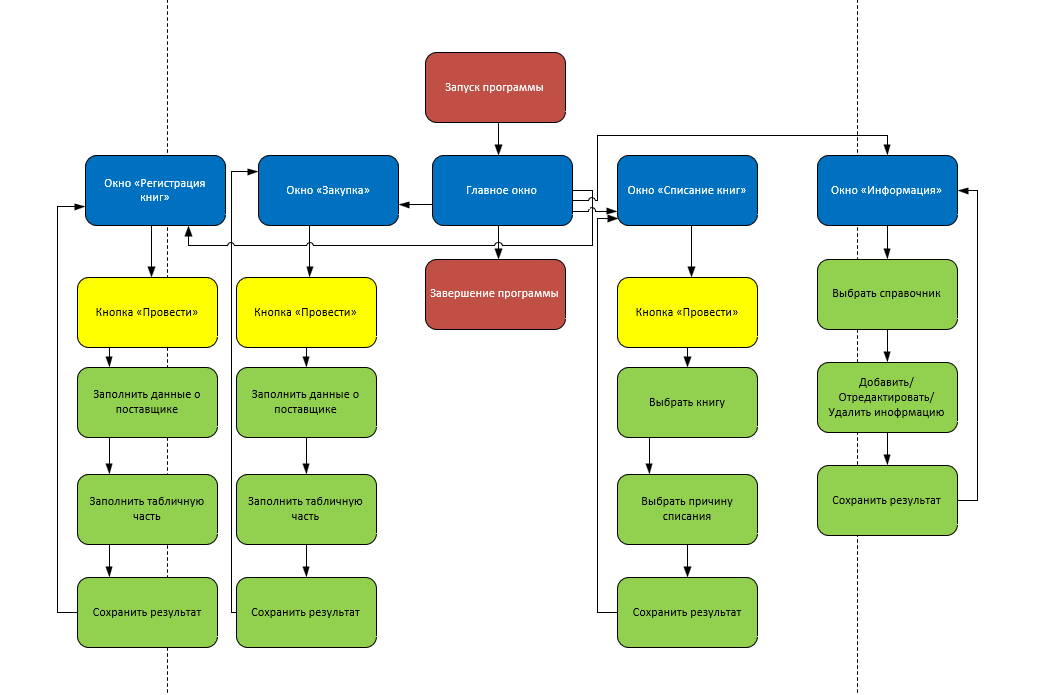


Рисунок 4 - Алгоритм выполнения программы

**Архитектура программного модуля:**

В основе архитектуры модуля лежит трехуровневая модель:

* **Уровень данных:** Встроенная база данных 1С: Предприятие служит хранилищем информации о книгах, закупках, приходах, списаниях и других сущностях, необходимых для учета библиотечного фонда.
* **Уровень бизнес-логики:** Реализуется в виде программных модулей, содержащих алгоритмы работы с данными. Здесь осуществляется проверка вводимой информации, формирование движений по регистрам, расчет необходимых показателей и т.д.
* **Уровень представления:** Представляет собой пользовательский интерфейс, реализованный в виде форм. Формы обеспечивают ввод, просмотр и редактирование данных, а также запуск различных функций модуля.

**Описание алгоритмов:**

В модуле реализованы алгоритмы для выполнения основных библиотечных операций:

* **Добавление новой книги:** При добавлении новой книги пользователь заполняет форму с реквизитами книги. Система проверяет корректность введенных данных и, в случае успешной проверки, создает новую запись в справочнике "Книги".

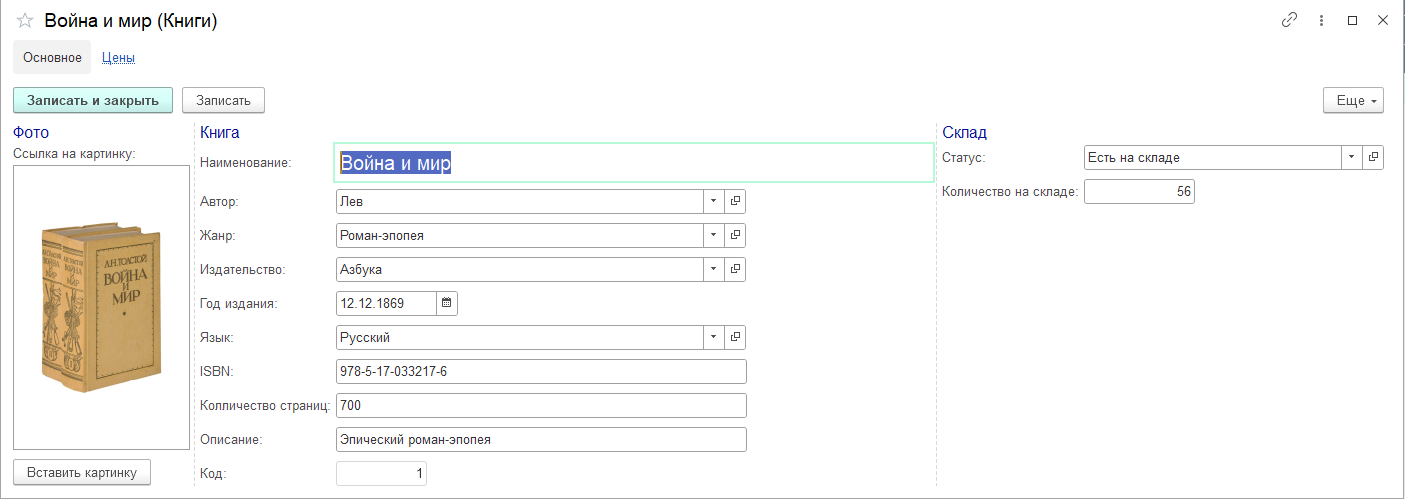


Рисунок 5- Окно добавления книги в систему

* **Редактирование и удаление книг:** Система предоставляет возможность изменения и удаления информации о книгах. При удалении книги модуль проверяет, не связана ли она с другими документами (например, закупками или списаниями).

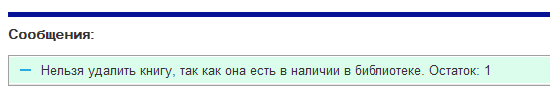


Рисунок 6- Сообщение-ошибка

|  |
| --- |
| &НаСервере  Процедура ПередУдалением(Отказ)  // Получаем остаток книги в регистре "ОстаткиКниг"  Остаток = ПолучитьОстатокКниги(ЭтотОбъект.Ссылка);  // Если остаток больше 0, запрещаем удаление  Если Остаток > 0 Тогда  Сообщить("Нельзя удалить книгу, так как она есть в наличии в библиотеке. Остаток: " + Остаток);  Отказ = Истина;  КонецЕсли;  КонецПроцедуры  &НаСервере  Функция ПолучитьОстатокКниги(Книга)  Запрос = Новый Запрос;  Запрос.Текст =  "ВЫБРАТЬ  | ОстаткиКниг.Количество  |ИЗ  | РегистрНакопления.ОстаткиКниг КАК ОстаткиКниг  |ГДЕ  | ОстаткиКниг.Книги = &Книга";    Запрос.УстановитьПараметр("Книга", Книга);    Результат = Запрос.Выполнить().Выбрать();    Если Результат.Следующий() Тогда  Возврат Результат.Количество;  Иначе  Возврат 0;  КонецЕсли;  КонецФункции |

Рисунок 7 - Процедура с проверкой перед удалением

* **Закупка книг:** Формирование заказа на покупку книг осуществляется с помощью специального документа "Закупка книг". В документе указывается поставщик, список заказываемых книг, количество, цена и другие условия. После поступления книг от поставщика информация о них регистрируется в модуле.

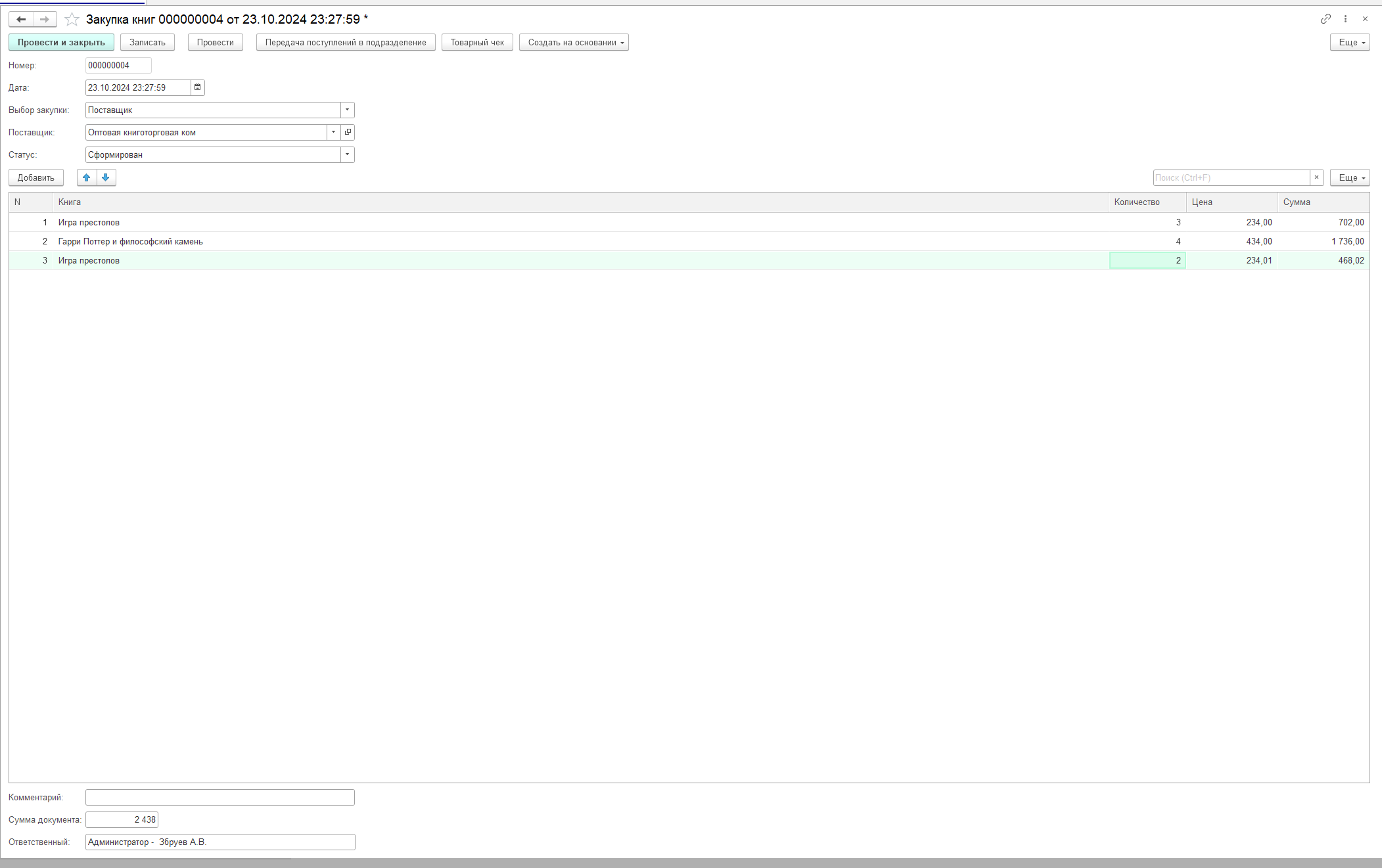


Рисунок 8 - Окно закупки книг

|  |
| --- |
| Процедура ПриЗаписи(Отказ)  Если Не ЗначениеЗаполнено(ЭтотОбъект.Ответственный) Тогда  ЭтотОбъект.Ответственный = ПользователиИнформационнойБазы.ТекущийПользователь();  КонецЕсли;  КонецПроцедуры  Процедура ПередЗаписью(Отказ, РежимЗаписи, РежимПроведения)  СуммаДокумента =0;  Для каждого ТабличнаяЧасть из Книги Цикл  СуммаДокумента = СуммаДокумента +ТабличнаяЧасть.Сумма;  КонецЦикла;  Если Не ЗначениеЗаполнено(ЭтотОбъект.Ответственный) Тогда  ЭтотОбъект.Ответственный = ПользователиИнформационнойБазы.ТекущийПользователь();  КонецЕсли;    КонецПроцедуры |

Рисунок 9 - Заполнение данных при/перед записью документа

* **Приход и списание книг:** Документы "Приход книг" и "Списание книг" позволяют отразить поступление и выбытие книг из библиотечного фонда с указанием источника получения или причины списания.

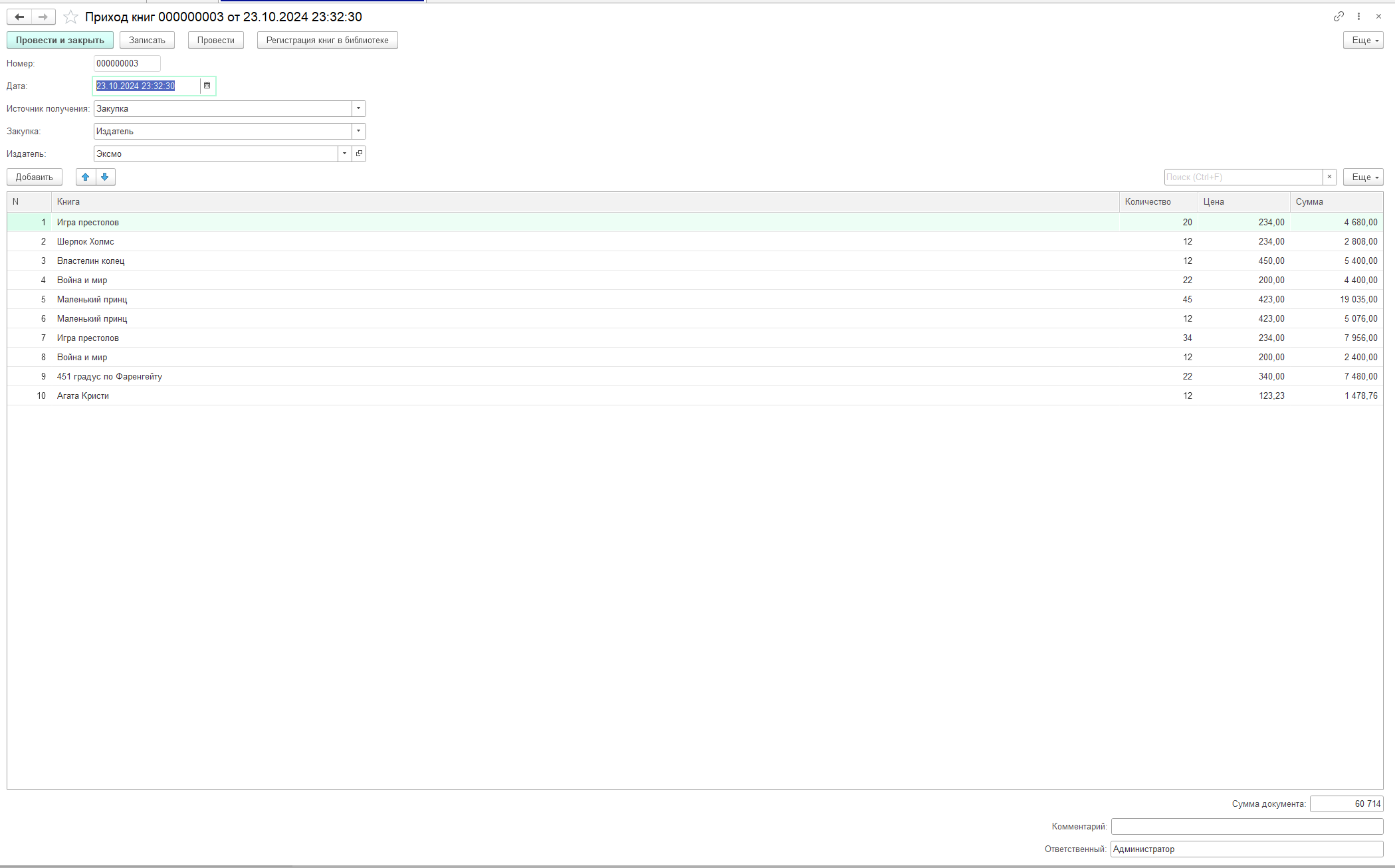


Рисунок 10 - Окно регистрации книг в библиотеку

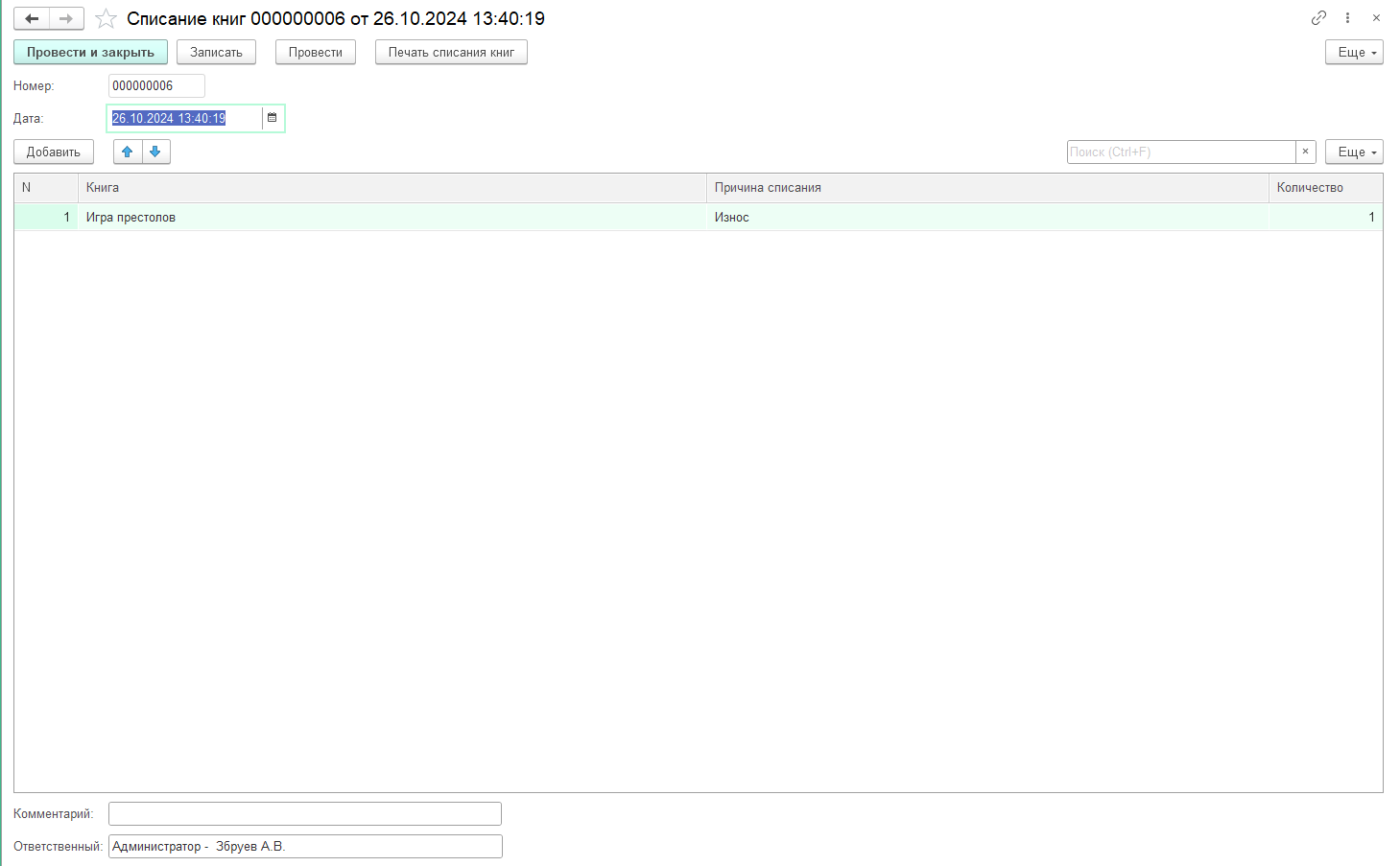


Рисунок 11 - Окно списание книги с библиотеки

|  |
| --- |
| Процедура ОбработкаПроведения(Отказ, Режим)    // регистр ОстаткиКниг Расход  Движения.ОстаткиКниг.Записывать = Истина;  Для Каждого ТекСтрокаКниги Из Книги Цикл  ОстатокКниг = ПолучитьОстаток(ТекСтрокаКниги.Книга);  Если ОстатокКниг<ТекСтрокаКниги.Количество Тогда  Сообщить(" Не достаточно книг "+ТекСтрокаКниги.Книга+" Доступный остаток: "+ОстатокКниг);  Отказ = Истина;  КонецЕсли;  Движение = Движения.ОстаткиКниг.Добавить();  Движение.ВидДвижения = ВидДвиженияНакопления.Расход;  Движение.Период = Дата;  Движение.Книги = ТекСтрокаКниги.Книга;  Движение.Количество = ТекСтрокаКниги.Количество;      КонецЦикла;  КонецПроцедуры  Функция ПолучитьОстаток(Книга)  Результат =0;  Отбор = Новый Структура;  Отбор.Вставить("Книги", Книга);  НайденныеЗначения = РегистрыНакопления.ОстаткиКниг.Остатки(Дата,Отбор);  Если НайденныеЗначения.Количество()>0 Тогда  Результат = НайденныеЗначения[0].Количество;  КонецЕсли;  Возврат Результат;  КонецФункции |

Рисунок 12 - Проверка количества книг перед внесением данных в регистр для исключения отрицательных значений

* **Формирование отчетов:** Модуль предоставляет возможность формирования различных отчетов, таких как отчет о состоянии фонда, отчет о движении книг, отчет о закупках и списаниях. Отчеты позволяют анализировать состояние библиотечного фонда, отслеживать динамику его изменения и принимать управленческие решения.

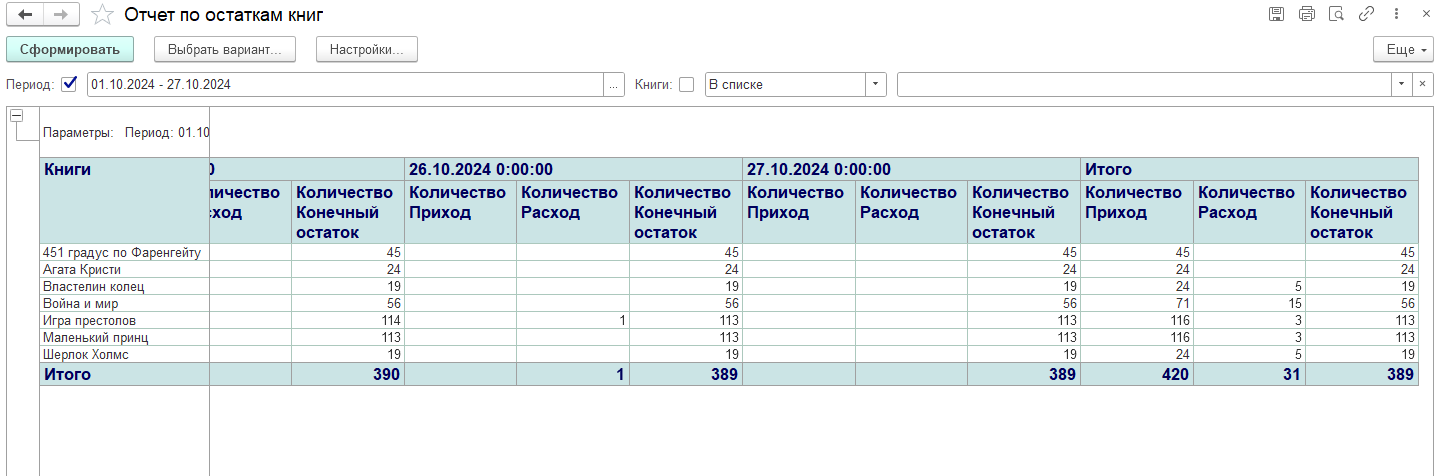


Рисунок 13 - Форма отчета с выбором периода, показатель оборота книг в фонде библиотеки

**Схема функционирования модуля:**

Взаимодействие пользователя с модулем осуществляется через интуитивно понятный интерфейс, который состоит из форм, меню, кнопок и других элементов управления. Пользователь может выполнять различные операции с книгами, создавать заказы, оформлять приходы и списания, формировать отчеты. Система обрабатывает действия пользователя, проверяет данные, выполняет необходимые расчеты и формирует результаты в виде отчетов или изменений в базе данных.

**Описание пользовательского интерфейса:**

Пользовательский интерфейс модуля разработан с учетом принципов эргономичности и удобства использования. Формы имеют логичную структуру, понятные названия полей и элементов управления. Система предоставляет подсказки и справочную информацию, что облегчает работу с модулем.

В данном разделе приведено описание алгоритмов и функционирования программного модуля для автоматизации учета книг в библиотеке. Разработанный модуль представляет собой удобный инструмент для эффективного управления библиотечным фондом и позволяет автоматизировать ключевые библиотечные процессы, упростить учетные операции и сформировать необходимую отчетность.

ТЕСТИРОВАНИЕ ПРОГРАММНОГО МОДУЛЯ

Тестирование программного обеспечения является важным этапом процесса разработки, позволяющим выявить ошибки и недочеты в работе программы, а также обеспечить ее соответствие заданным функциональным требованиям. В рамках тестирования разработанного программного модуля для автоматизации учета книг в библиотеке были проведены следующие тесты.

**Тестирование функции удаления книги**

В ходе тестирования функции удаления книги была проведена проверка корректности работы системы при попытке удалить книгу, которая присутствует в регистре накопления "ОстаткиКниг".

Для этого была создана новая книга в справочнике "Книги", затем оформлен документ "Приход книг" с данной книгой. После этого была предпринята попытка удалить книгу из справочника "Книги". Как и ожидалось, система выдала сообщение об ошибке, указывающее на невозможность удаления книги, так как она имеется в наличии в библиотеке.

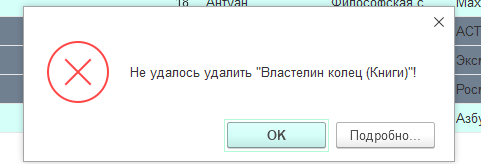


Рисунок 14 - Сообщение об ошибке при попытке удаления книги, присутствующей в регистре накопления "ОстаткиКниг"

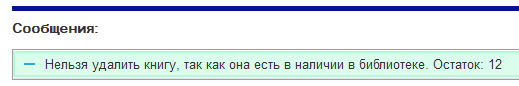


Рисунок 15 - Дополнительный вывод сообщения

Таким образом, тест подтвердил, что система корректно блокирует удаление книги, если ее остаток в библиотеке больше 0.

**Тестирование документа "Списание книг"**

Для проверки исключения отрицательных значений в регистре накопления "ОстаткиКниг" после списания книг был проведен тест, в ходе которого была предпринята попытка списать большее количество экземпляров книги, чем имеется в наличии.

В рамках теста была создана новая книга в справочнике "Книги", а затем оформлен документ "Приход книг" с данной книгой, указав количество 15 экземпляров. После этого был создан документ "Списание книг" с данной книгой, но указано количество 15 экземпляров. При попытке провести документ "Списание книг" система выдала сообщение об ошибке, информируя пользователя о невозможности списания большего количества экземпляров, чем имеется в наличии

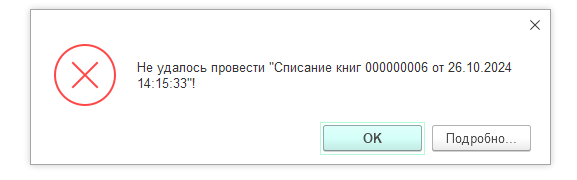


Рисунок 16 - Сообщение об ошибке при попытке списания большего количества книг, чем есть в наличии

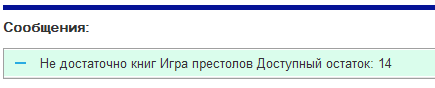


Рисунок 17 - Дополнительный вывод сообщения

Тест подтвердил, что система корректно контролирует остатки книг и не допускает возникновения отрицательных значений.

**Тестирование справочника "Авторы"**

Для проверки корректности работы механизма контроля заполнения обязательных полей в справочнике "Авторы" был проведен тест.

В рамках теста была открыта форма создания новой записи в справочнике "Авторы". Поле "Фамилия", являющееся обязательным для заполнения, было оставлено пустым. При попытке сохранить запись система выдала сообщение об ошибке, указывающее на необходимость заполнения обязательного поля

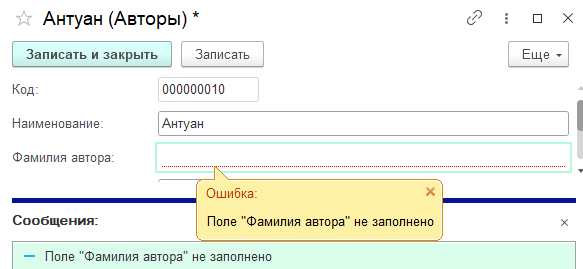


Рисунок 18 - Сообщение об ошибке при попытке сохранить запись в справочнике "Авторы" с незаполненным обязательным полем "Фамилия"

Тест подтвердил, что система корректно контролирует заполнение обязательных полей в справочнике "Авторы" и не позволяет сохранять записи с незаполненными обязательными полями.

Тестирование программы осуществлялось на персональном компьютере со следующими техническими характеристиками:

* Процессор – AMD Ryzen 5 3500 AM4
* Оперативная память – DDR3 16 ГБ
* Видеокарта – NVIDIA 1660 Super
* Операционная система – Windows 11

Проведенное тестирование программного модуля подтвердило его корректную работу и соответствие заданным функциональным требованиям. Все выявленные в ходе тестирования недочеты были исправлены, что позволяет утверждать, что модуль готов к использованию в реальных условиях.

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

**1. Введение**

Данное руководство предназначено для пользователей программного модуля "Автоматизация учета книг в библиотеке". В нем приводится информация о назначении модуля, его функциональных возможностях, а также инструкции по использованию основных функций.

**2. Назначение программы**

Программный модуль "Автоматизация учета книг в библиотеке" предназначен для упрощения и автоматизации процессов учета книг в небольших библиотеках. Модуль позволяет вести электронный каталог книг, отслеживать их движение, формировать заказы на покупку книг, оформлять приходы и списания, а также получать различную отчетность о состоянии библиотечного фонда.

**3. Запуск программы**

Для запуска программного модуля необходимо иметь установленную платформу 1С: Предприятие 8.3. После запуска платформы необходимо выбрать базу данных, в которой установлен модуль, и ввести имя пользователя и пароль.



Рисунок 19 - Окно запуска 1С: Предприятие

**4. Работа с модулем**

После успешной авторизации откроется главное окно программы. В главном окне доступны различные разделы и функции модуля.

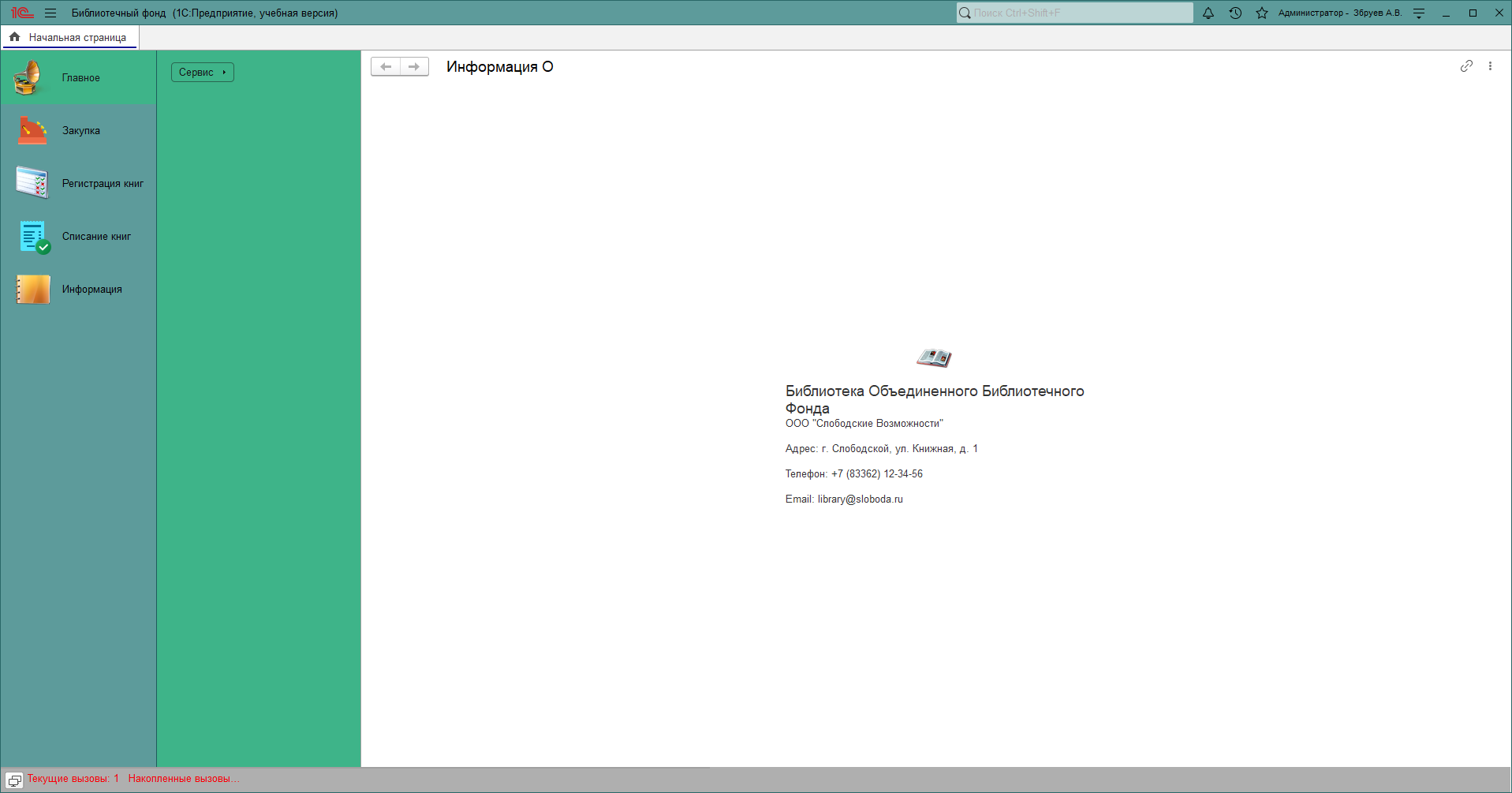


Рисунок 20 - Главное окно программного модуля

**4.1. Учет книг**

Для работы с книгами необходимо перейти в раздел "Информация" –‘’Книг’’. В этом разделе доступны следующие функции:

* **Добавление новой книги:** Для добавления новой книги нажмите кнопку "Создать". Откроется форма для ввода данных о книге. Заполните все необходимые поля и нажмите кнопку "Записать".

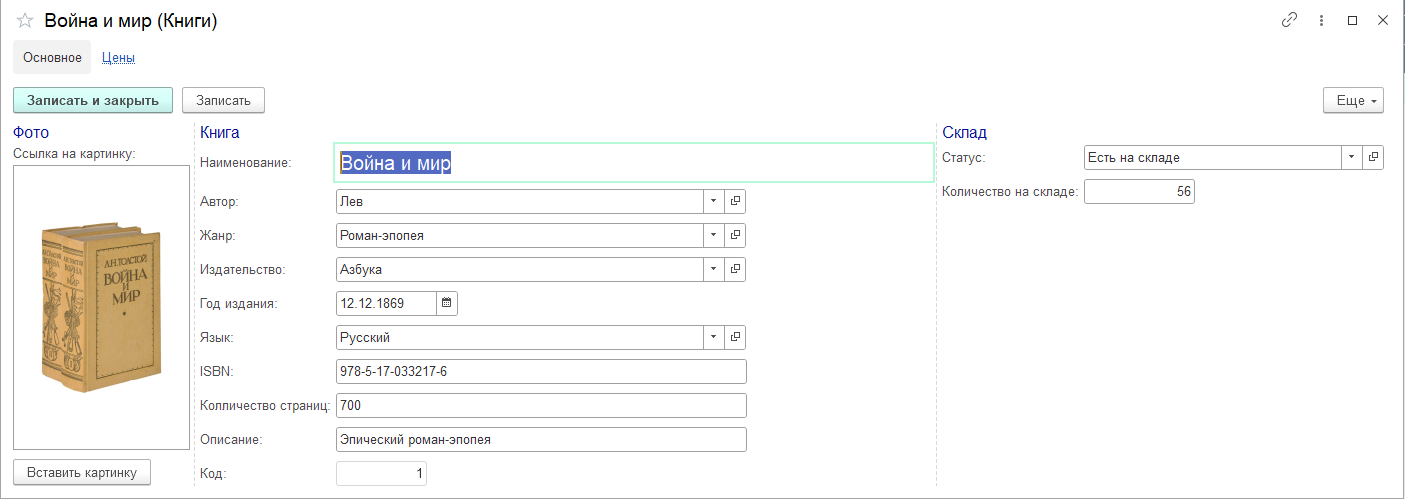


Рисунок 21 - Форма добавления новой книги

* **Редактирование книги:** Для редактирования данных о книге откройте ее карточку и внесите необходимые изменения. Затем нажмите кнопку "Записать".
* **Удаление книги:** Для удаления книги выделите ее в списке и нажмите кнопку "Удалить". **Внимание!** Перед удалением книги убедитесь, что она не используется в других документах (закупках, списаниях и т.д.).
* **Поиск книг:** Для поиска книг используйте форму поиска. Вы можете искать книги по названию, автору, жанру, инвентарному номеру и другим критериям.

**4.2. Закупка книг**

Для оформления закупки книг необходимо перейти в раздел "Закупки". В этом разделе можно:

* **Создать новый заказ на покупку:** Нажмите кнопку "Создать" и заполните форму заказа, указав поставщика, список заказываемых книг, количество, цену и другие условия.

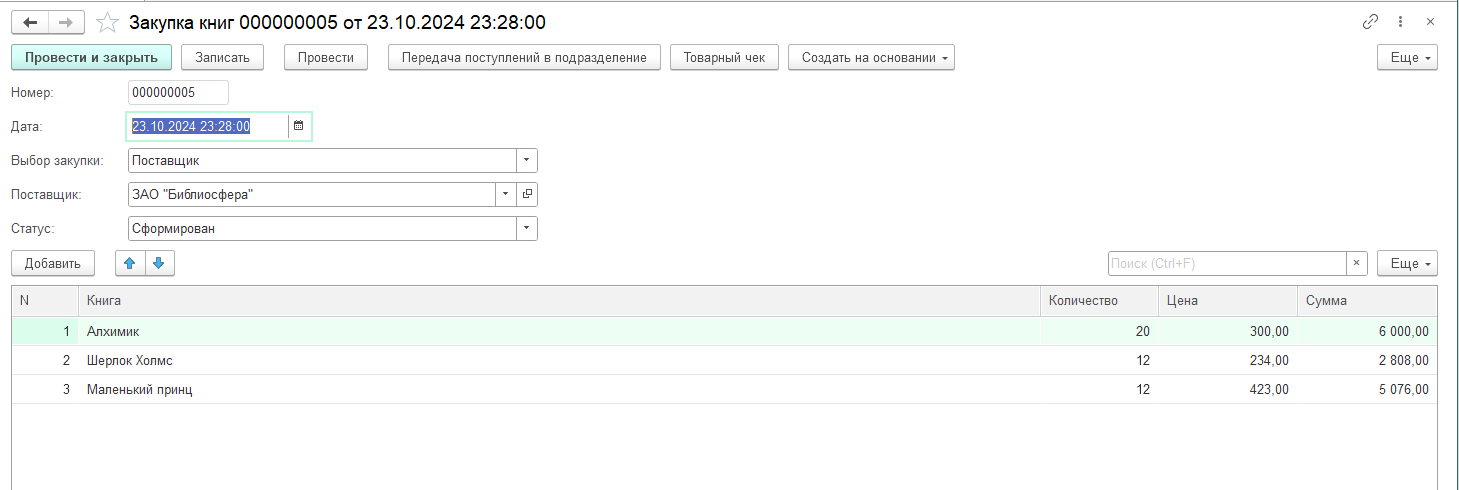


Рисунок 22 - Форма создания заказа на книги

* **Отслеживать статус заказов:** Просматривайте список заказов и информацию о статусе их выполнения (сформирован, оплачен, получен).
* **Формировать отчеты о закупках:** в разделе "Отчеты" можно сформировать отчет о закупках книг за выбранный период.

**4.3. Регистрация поступления книг**

При поступлении книг в библиотеку необходимо оформить документ "Приход книг". Для этого:

* Перейдите в раздел "Приход книг" и нажмите кнопку "Создать".
* Выберите источник получения книг: "Закупка" или "Пожертвование".
* **Если источник - "Закупка":** Выберите соответствующий заказ из списка. Данные о книгах из заказа будут автоматически добавлены в табличную часть документа.
* **Если источник - "Пожертвование":** Заполните информацию о дарителе и добавьте данные о книгах вручную.

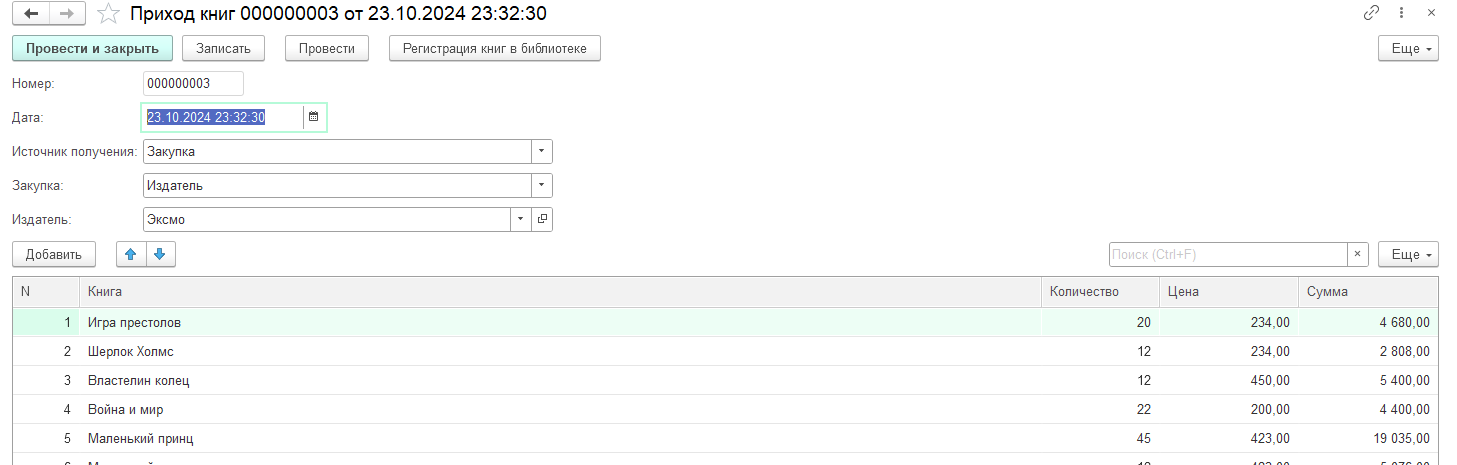
****

Рисунок 23 - Форма документа "Приход книг"

**4.4. Списание книг**

Для оформления списания книг из библиотеки:

* Перейдите в раздел "Списание книг" и нажмите кнопку "Создать".
* Добавьте в табличную часть документа книги, которые подлежат списанию.
* Для каждой книги укажите причину списания.

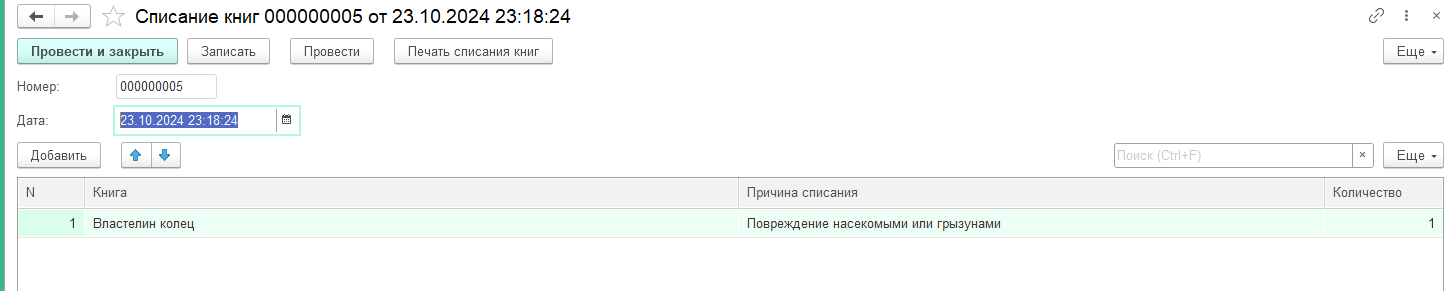
****

Рисунок 24 - Форма документа "Списание книг"

**5. Завершение работы**

Для завершения работы с программным модулем выберите поле активного пользователя и нажмите на кнопку “Завершить работу”

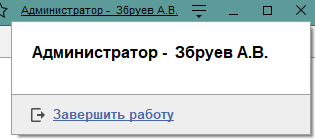


Рисунок 25 - Окно завершения работы

ЗАКЛЮЧЕНИЕ.

В ходе выполнения курсовой работы была разработана программная система для автоматизации учета книг в библиотеке. Данная система позволяет эффективно управлять библиотечным фондом, оптимизировать процессы закупки, прихода, списания и учета книг, а также формировать необходимую отчетность.

Разработанный программный модуль прост и удобен в использовании благодаря интуитивно понятному интерфейсу и логичной структуре. Он автоматизирует множество рутинных операций, таких как ввод данных о книгах, формирование заказов и оформление приходов и списаний, что экономит время и снижает риск ошибок. Система обеспечивает высокую точность учета библиотечного фонда, исключая ошибки, связанные с ручным вводом данных. Кроме того, модуль улучшает контроль за движением книг, отслеживая их перемещения, и позволяет формировать различные отчеты для анализа состояния фонда.

Разработанный программный модуль может быть использован в небольших библиотеках различных типов для оптимизации процессов управления библиотечным фондом и повышения эффективности их работы.

В дальнейшем планируется расширить функциональность модуля и включить в него возможности для учета читателей, выдачи и возврата книг, а также продления сроков пользования. Это позволит создать полноценную автоматизированную систему управления библиотекой, которая будет отвечать потребностям широкого круга пользователей. Также планируется усовершенствовать пользовательский интерфейс, сделав его еще более интуитивно понятным и удобным в использовании, и добавить возможность интеграции с другими информационными системами, такими как электронные каталоги и базы данных.

СПИСОК ЛИТЕРАТУРЫ

1. ГОСТ 7.1–2003. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание. Общие требования и правила составления.
2. ГОСТ 7.82–2001. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления.
3. Асеев, Г.Г. Автоматизация библиотечных процессов: учебное пособие / Г.Г. Асеев. – Москва: Либерея-Бибинформ, 2018. – 288 с.
4. Балахонов, В.А. 1С:Предприятие 8.3. Практическое руководство разработчика / В.А. Балахонов. – Москва: 1С-Паблишинг, 2019. – 800 с.
5. Гусева, Е.Н. Автоматизированные библиотечные информационные системы / Е.Н. Гусева. – Санкт-Петербург: Профессия, 2017. – 352 с.
6. Евстигнеева, Г.А. Технологии автоматизации библиотечных процессов: учебное пособие / Г.А. Евстигнеева. – Москва: Флинта, 2020. – 256 с.
7. Зупарова, Л.Б., Соколов, А.В. Автоматизированные библиотечно-информационные системы: учебное пособие / Л.Б. Зупарова, А.В. Соколов. – Москва: Юрайт, 2021. – 384 с.
8. Иванова, Е.А. Организация и технология библиотечного дела: учебное пособие / Е.А. Иванова. – Москва: КноРус, 2020. – 272 с.
9. Каталогизация и классификация документов: учебник для студентов вузов культуры и искусств / под ред. Р.С. Гиляревского. – Москва: Либерея-Бибинформ, 2019. – 400 с.
10. Книговедение: учебник для студентов вузов культуры и искусств / под ред. Н.М. Сикорского. – Москва: ФАИР-ПРЕСС, 2018. – 592 с.
11. Мезенцев, К.А. Энциклопедия библиотечного дела / К.А. Мезенцев. – Москва: Либерея-Бибинформ, 2016. – 784 с.
12. Радченко, М.Г. 1С: Предприятие 8.3. Программирование для начинающих / М.Г. Радченко. – Москва: 1С-Паблишинг, 2020. – 432 с.
13. Справочник библиотекаря / под ред. А.Н. Ванеева. – Санкт-Петербург: Профессия, 2019. – 640 с.
14. Хахаев, И.А. Информационные технологии в библиотечном деле: учебное пособие / И.А. Хахаев. – Москва: Либерея-Бибинформ, 2022. – 320 с.
15. Чат-бот Bard (Google). Помощник в формировании тезисов, определений и поиском информации для предметной области. – [Электронный ресурс]. – Режим доступа: [https://bard.google.com/](https://www.google.com/url?sa=E&q=https%3A%2F%2Fbard.google.com%2F)

ПРИЛОЖЕНИЕ

**Приложение 1**

**КОД ПРОГРАМ**

**Код формы элемента справочника «Статус»**

#Область ОбработчикиСобытийФормы

&НаСервере

Процедура ПередЗаписьюНаСервере(Отказ, ТекущийОбъект, ПараметрыЗаписи)

ЗначениеЦвета = Цвет;

Если ЗначениеЦвета = Новый Цвет(0,0,0) Тогда

ЗначениеЦвета = Неопределено;

КонецЕсли;

ТекущийОбъект.Цвет=Новый ХранилищеЗначения(ЗначениеЦвета);

КонецПроцедуры

&НаСервере

Процедура ПриЧтенииНаСервере(ТекущийОбъект)

Цвет = ТекущийОбъект.Цвет.Получить();

КонецПроцедуры

#КонецОбласти

**Код модуля объекта справочника «Книги»**

&НаСервере

Процедура ПередУдалением(Отказ)

// Получаем остаток книги в регистре "ОстаткиКниг"

Остаток = ПолучитьОстатокКниги(ЭтотОбъект.Ссылка);

// Если остаток больше 0, запрещаем удаление

Если Остаток > 0 Тогда

Сообщить("Нельзя удалить книгу, так как она есть в наличии в библиотеке. Остаток: " + Остаток);

Отказ = Истина;

КонецЕсли;

КонецПроцедуры

&НаСервере

Функция ПолучитьОстатокКниги(Книга)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ОстаткиКниг.Количество

|ИЗ

| РегистрНакопления.ОстаткиКниг КАК ОстаткиКниг

|ГДЕ

| ОстаткиКниг.Книги = &Книга";

Запрос.УстановитьПараметр("Книга", Книга);

Результат = Запрос.Выполнить().Выбрать();

Если Результат.Следующий() Тогда

Возврат Результат.Количество;

Иначе

Возврат 0;

КонецЕсли;

КонецФункции

**Код формы элемента справочника «Книги»**

&НаКлиенте

Процедура ВставитьКартинку(Команда)

Оповещение = Новый ОписаниеОповещения("ОбработатьВыборФайла",ЭтотОбъект);

НачатьПомещениеФайла(Оповещение,,,Истина, УникальныйИдентификатор);

КонецПроцедуры

&НаКлиенте

Процедура ОбработатьВыборФайла(Результат, Адрес,

ВыбранноеИмяФайла,ДополнительныйПараметр) Экспорт

Если НЕ Результат Тогда

Возврат;

КонецЕсли;

СсылкаНаКартинку = Адрес;

КонецПроцедуры

&НаСервере

Процедура ПередЗаписьюНаСервере(Отказ, ТекущийОбъект, ПараметрыЗаписи)

Если ЭтоАдресВременногоХранилища(СсылкаНаКартинку) Тогда

ТекущийОбъект.ДанныеКартинки = Новый

ХранилищеЗначения(ПолучитьИзВременногоХранилища(СсылкаНаКартинку));

КонецЕсли;

КонецПроцедуры

&НаСервере

Процедура ПриСозданииНаСервере(Отказ, СтандартнаяОбработка)

СсылкаНаКартинку = ПолучитьНавигационнуюСсылку(Объект.Ссылка,

"ДанныеКартинки");

КонецПроцедуры

&НаСервере

Процедура ПриЧтенииНаСервере(ТекущийОбъект)

// Получаем количество книг на складе

КоличествоНаСкладе = ПолучитьОстаток(ТекущийОбъект.Ссылка);

// Устанавливаем количество в поле на форме

Объект.КоличествоНаСкладе = КоличествоНаСкладе;

Если КоличествоНаСкладе > 0 Тогда

Объект.Статус = Справочники.Статус.НайтиПоНаименованию("Есть на складе");

Иначе

Объект.Статус = Справочники.Статус.НайтиПоНаименованию("Отсутствует на складе");

КонецЕсли;

КонецПроцедуры

&НаСервере

Функция ПолучитьОстаток(Книга)

Результат = 0;

Отбор = Новый Структура;

Отбор.Вставить("Книги", Книга);

НайденныеЗначения = РегистрыНакопления.ОстаткиКниг.Остатки(ТекущаяДата(), Отбор);

Если НайденныеЗначения.Количество() > 0 Тогда

Результат = НайденныеЗначения[0].Количество;

КонецЕсли;

Возврат Результат;

КонецФункции

**Код формы списка справочника «Книги»**

#Область ОбработкиСобытийФормы

&НаСервере

Процедура ПриСозданииНаСервере(Отказ, СтандартнаяОбработка)

УстановитьУсловноеОформление();

#КонецОбласти

#Область СлужебныеПроцедурыИФункции

КонецПроцедуры

&НаСервере

Процедура УстановитьУсловноеОформление()

УсловноеОформлениеСписка=Список.КомпоновщикНастроек.Настройки.УсловноеОформление;

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| Статус.Ссылка КАК Ссылка,

| Статус.Цвет КАК Цвет

|ИЗ

| Справочник.Статус КАК Статус";

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

ЭлементОформления = УсловноеОформлениеСписка.Элементы.Добавить();

ЭлементОформления.Оформление.УстановитьЗначениеПараметра("ЦветФона", ВыборкаДетальныеЗаписи.Цвет.Получить());

ЭлементОтбора = ЭлементОформления.Отбор.Элементы.Добавить(Тип("ЭлементОтбораКомпоновкиДанных"));

ЭлементОтбора.ЛевоеЗначение=Новый ПолеКомпоновкиДанных("Статус");

ЭлементОтбора.ВидСравнения = ВидСравненияКомпоновкиДанных.Равно;

ЭлементОтбора.ПравоеЗначение = ВыборкаДетальныеЗаписи.Ссылка;

КонецЦикла;

КонецПроцедуры

#КонецОбласти

**Код формы документа «Приход книг»**

&НаКлиенте

Перем Пожертвование;

Перем Закупка;

&НаКлиенте

Процедура КнигиЦенаЗаШтПриИзменении(Элемент)

СтрокаТабличнойЧасти = Элементы.Книги.ТекущиеДанные;

СтрокаТабличнойЧасти.Сумма = СтрокаТабличнойЧасти.Количество\*СтрокаТабличнойЧасти.Цена;

КонецПроцедуры

&НаКлиенте

Процедура КнигиСуммаЦенаПриИзменении(Элемент)

СтрокаТабличнойЧасти = Элементы.Книги.ТекущиеДанные;

Если СтрокаТабличнойЧасти.Количество = 0 тогда

СтрокаТабличнойЧасти.ЦенаЗаШ = 0;

Иначе

СтрокаТабличнойЧасти.Цена = СтрокаТабличнойЧасти.Сумма / СтрокаТабличнойЧасти.Количество;

КонецЕсли;

КонецПроцедуры

&НаКлиенте

Процедура КнигиКоличествоПриИзменении(Элемент)

СтрокаТабличнойЧасти = Элементы.Книги.ТекущиеДанные;

СтрокаТабличнойЧасти.Сумма = СтрокаТабличнойЧасти.Количество\*СтрокаТабличнойЧасти.Цена;

КонецПроцедуры

&НаСервере

Процедура ИсточникПолученияПриИзмененииНаСервере()

Если Объект.ИсточникПолучения = Перечисления.ИсточникПолучения.Пожертвование тогда

Элементы.ДанныеОпожертвований.Видимость = Истина;

Элементы.Закупка.Видимость = Ложь;

ИначеЕсли Объект.ИсточникПолучения = Перечисления.ИсточникПолучения.Закупка тогда

Элементы.Закупка.Видимость = Истина;

Элементы.ДанныеОпожертвований.Видимость = Ложь;

КонецЕсли;

КонецПроцедуры

&НаКлиенте

Процедура ИсточникПолученияПриИзменении(Элемент)

ИсточникПолученияПриИзмененииНаСервере();

КонецПроцедуры

&НаСервере

Процедура ПриОткрытииНаСервере()

Если Объект.ИсточникПолучения = Перечисления.ИсточникПолучения.Пожертвование тогда

Элементы.ДанныеОпожертвований.Видимость = Истина;

Элементы.Закупка.Видимость = Ложь;

ИначеЕсли Объект.ИсточникПолучения = Перечисления.ИсточникПолучения.Закупка тогда

Элементы.Закупка.Видимость = Истина;

Элементы.ДанныеОпожертвований.Видимость = Ложь;

КонецЕсли;

Если Объект.Закупка = Перечисления.ВыборЗакупки.Издатель тогда

Элементы.Издатель.Видимость = Истина;

Элементы.Поставщик.Видимость = Ложь;

ИначеЕсли Объект.Закупка = Перечисления.ВыборЗакупки.Поставщик тогда

Элементы.Издатель.Видимость = Ложь;

Элементы.Поставщик.Видимость = Истина;

КонецЕсли;

КонецПроцедуры

&НаКлиенте

Процедура ПриОткрытии(Отказ)

ПриОткрытииНаСервере();

КонецПроцедуры

&НаСервере

Процедура ЗакупкаПриИзмененииНаСервере()

Если Объект.Закупка = Перечисления.ВыборЗакупки.Издатель тогда

Элементы.Издатель.Видимость = Истина;

Элементы.Поставщик.Видимость = Ложь;

ИначеЕсли Объект.Закупка = Перечисления.ВыборЗакупки.Поставщик тогда

Элементы.Издатель.Видимость = Ложь;

Элементы.Поставщик.Видимость = Истина;

КонецЕсли;

КонецПроцедуры

&НаКлиенте

Процедура ЗакупкаПриИзменении(Элемент)

ЗакупкаПриИзмененииНаСервере();

КонецПроцедуры

&НаКлиенте

Процедура КнигиКнигаПриИзменении(Элемент)

СтрокаТЧ = Элементы.Книги.ТекущиеДанные;

СтрокаТЧ.Цена=ОбщийМодульФункции.ЦеныИзРегистраСведений(Объект.Дата,СтрокаТЧ.Книга);

КонецПроцедуры

&НаСервере

Процедура ДанныеОпожертвованийПриИзмененииНаСервере()

// Проверяем, выбран ли документ пожертвования

Если ЗначениеЗаполнено(Объект.ДанныеОПожертвований) Тогда

// Получаем документ пожертвования

Пожертвование = Объект.ДанныеОПожертвований.ПолучитьОбъект();

// Очищаем табличную часть "Книги" в документе "Приход книг"

Объект.Книги.Очистить();

// Копируем данные из табличной части "Книги" документа "Пожертвование"

Для Каждого СтрокаПожертвования Из Пожертвование.Книги Цикл

НоваяСтрока = Объект.Книги.Добавить();

НоваяСтрока.Книга = СтрокаПожертвования.Книга;

НоваяСтрока.Количество = СтрокаПожертвования.Количество;

НоваяСтрока.Цена = СтрокаПожертвования.Цена;

НоваяСтрока.Сумма = СтрокаПожертвования.Сумма;

КонецЦикла;

КонецЕсли;

КонецПроцедуры

&НаКлиенте

Процедура ДанныеОпожертвованийПриИзменении(Элемент)

ДанныеОпожертвованийПриИзмененииНаСервере();

КонецПроцедуры

**Код модуля объекта документа «Списание книг»**

Процедура ОбработкаПроведения(Отказ, Режим)

// регистр ОстаткиКниг Расход

Движения.ОстаткиКниг.Записывать = Истина;

Для Каждого ТекСтрокаКниги Из Книги Цикл

ОстатокКниг = ПолучитьОстаток(ТекСтрокаКниги.Книга);

Если ОстатокКниг<ТекСтрокаКниги.Количество Тогда

Сообщить(" Не достаточно книг "+ТекСтрокаКниги.Книга+" Доступный остаток: "+ОстатокКниг);

Отказ = Истина;

КонецЕсли;

Движение = Движения.ОстаткиКниг.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Расход;

Движение.Период = Дата;

Движение.Книги = ТекСтрокаКниги.Книга;

Движение.Количество = ТекСтрокаКниги.Количество;

КонецЦикла;

КонецПроцедуры

Функция ПолучитьОстаток(Книга)

Результат =0;

Отбор = Новый Структура;

Отбор.Вставить("Книги", Книга);

НайденныеЗначения = РегистрыНакопления.ОстаткиКниг.Остатки(Дата,Отбор);

Если НайденныеЗначения.Количество()>0 Тогда

Результат = НайденныеЗначения[0].Количество;

КонецЕсли;

Возврат Результат;

КонецФункции

Процедура ПриЗаписи(Отказ)

Если Не ЗначениеЗаполнено(ЭтотОбъект.Ответственный) Тогда

ЭтотОбъект.Ответственный = ПользователиИнформационнойБазы.ТекущийПользователь();

КонецЕсли;

КонецПроцедуры

**Код модуля документа «Закупка книг»**

Процедура ПриЗаписи(Отказ)

Если Не ЗначениеЗаполнено(ЭтотОбъект.Ответственный) Тогда

ЭтотОбъект.Ответственный = ПользователиИнформационнойБазы.ТекущийПользователь();

КонецЕсли;

КонецПроцедуры

Процедура ПередЗаписью(Отказ, РежимЗаписи, РежимПроведения)

СуммаДокумента =0;

Для каждого ТабличнаяЧасть из Книги Цикл

СуммаДокумента = СуммаДокумента +ТабличнаяЧасть.Сумма;

КонецЦикла;

Если Не ЗначениеЗаполнено(ЭтотОбъект.Ответственный) Тогда

ЭтотОбъект.Ответственный = ПользователиИнформационнойБазы.ТекущийПользователь();

КонецЕсли;

КонецПроцедуры

Процедура ОбработкаПроведения(Отказ, Режим)

// регистр ОборотЗаказовКниг Приход

Движения.ОборотЗаказовКниг.Записывать = Истина;

Для Каждого ТекСтрокаКниги Из Книги Цикл

Движение = Движения.ОборотЗаказовКниг.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Приход;

Движение.Период = Дата;

Движение.Книга = ТекСтрокаКниги.Книга;

Движение.ВыборЗакупки = ВыборЗакупки;

Движение.Поставщик = Поставщик;

Движение.Издатель = Издатель;

Движение.Количество = ТекСтрокаКниги.Количество;

КонецЦикла;

КонецПроцедуры