АННОТАЦИЯ

Пояснительная записка к курсовому проекту содержит постановку и программу решения задачи «Автоматизация учета канцелярских товаров на складе оптовой базы».

Программа «Канцелярик.exe» написана на языке C# в среде программирования Visual Studio 2022 с использованием сервера баз данных MS SQL Server 2019, предназначена для работы в операционной системе Windows 10 и выше, отлажена на данных контрольного примера

СОДЕРЖАНИЕ

|  |  |
| --- | --- |
|  | лист |
| Введение | 4 |
| 1 Проектирование информационной системы | 6 |
| 1.1 Техническое задание  1.2 Описание предметной области | 6  13 |
| 1.3 Функциональная модель предметной области | 16 |
| 1.4 Диаграмма прецедентов  1.5 Описание входной информации | 18  18 |
| 1.6 Описание выходной информации | 18 |
| 1.7 Концептуальное проектирование | 20 |
| 1.8 Логическое моделирование  1.9 Описание структуры базы данных | 21  21 |
| 1.10 Контрольный пример  1.11 Общие требования к разрабатываемому продукту | 24  29 |
| 2 Экспериментальный раздел | 31 |
| 2.1 Описание программы  2.2 Протокол тестирования программного продукта  2.3 Руководство пользователя  2.4 Меры по обеспечению защиты информации  Заключение  Список сокращений  Список использованных источников | 31  38  47  53  54  55  56 |

# ВВЕДЕНИЕ

Необходимость автоматизации информационных процессов вызвана возрастанием объемов информации в информационной системе организаций, потребностью в ускорении и применением более сложных способов ее обработки.

Автоматизация информационных процессов проводится с целью повышения производительности и эффективности труда работающих, улучшения качества информационных услуг и продукции, повышения сервиса и оперативности обслуживания клиентов. Автоматизация основана на использовании средств компьютерной техники и необходимого программного обеспечения.

Учет канцтоваров на складе является сложной задачей и при работе, работники сталкивается с целым рядом проблем. Во-первых, низкая скорость обработки клиентских заказов. Эта проблема связана с большой затратой времени на ввод данных по учету движения и количества товара. Во-вторых, это зависимость от конкретных людей. А именно, когда склад не автоматизирован, одним из ключевых факторов является зависимость от узкого круга сотрудников, которым известны принципы размещения товаров, количество этих же товаров, места нахождения запасов, и прочие особенности. Кроме того, довольно частой проблемой является человеческий фактор, ошибки персонала. При плохой организации складского учета, часто возникает ситуация, когда товары размещаются не по складским правилам, а по логике определенного сотрудника что приводит к лишней затрате времени и сил. Таким образом, автоматизация системы склада, которая решает все перечисленные проблемы, в настоящее время просто необходима.

Целью курсового проекта является облегчение и ускорения ведения учета канцтоваров складскими работниками, уменьшения ошибок допущенными человеческим фактором и подготовка документации для выдачи заказчику, путем разработки программного обеспечения.

Для выполнения этой цели требуется выполнить следующие задачи:

* описать предметную область;
* выполнить функциональное проектирование;
* создать структуру базы данных;
* составить контрольный пример;
* разработать приложение для решения задач;
* протестировать приложение АИС;
* составить техническую документацию;
* создать приложение для решения задачи;
* создать справочники;
* создать документ для внесения данных по продаже товаров;
* создать отчет по продажам за период.

Альтернативные программы: для учёта канцелярских товаров используются такие программы как 1C: предприятие, ТОРГСОФТ.

1С: предприятие включает в себя платформу и прикладные решения, разработанные на ее основе, для автоматизации деятельности организаций и частных лиц. Сама платформа не является программным продуктом для использования конечными пользователями, которые обычно работают с одним из многих прикладных решений (конфигураций), разработанных на данной платформе. Такой подход позволяет автоматизировать различные виды деятельности, используя единую технологическую платформу. Единственный минус, в такой программе совершенно невозможно работать с графикой. ТОРГСОФТ - простая и функциональная компьютерная программа для торговли и складского учета с использованием штрих-кодов.

Полезная программа для ведения учета, которая наладит реальный учет и контроль в торговле разных форматов: розничный магазин, сеть магазинов, мелкооптовая торговля, интернет-магазин. Минус программы состоит в том, что не всегда можно внедрить доработки программы в оперативном режиме.

# Проектирование информационной системы

1.1 Техническое задание

Разработать техническое задание на разработку при«Автоматизация учета канцелярских товаров на складе оптовой базы».

* + 1. Общие сведение

Работа выполняется в рамках проекта «Автоматизация учета канцелярских товаров на складе оптовой базы».

1.1.1.1 Наименование системы

Полное наименование системы: «Автоматизированная система учета канцелярских товаров на складе оптовой базы».

Краткое наименование: АСУКТНСОБ, Система.

1.1.1.2 Номер договора №1 от 26.01.22

1.1.1.3 Наименование разработчика и заказчика работ, их реквизиты

Заказчик: Уфимский Колледж Статистики, Информатики и Вычислительной Техники.

Юридический адрес заказчика: 450005, г.Уфа, ул.Кирова, 65.

Почтовый адрес заказчика: 450005, г.Уфа, ул.Кирова, 65.

Телефон заказчика: +7 (347) 228-67-72, 228-67-74, 252-40-84.

Факс заказчика: 8 (347) 228-67-72.

Разработчик: Теплых Сергей Александрович.

Телефон разработчика: +7 (986) 965-46-21.

Электронная почта разработчика: teplykh-26032004@mail.ru

1.1.1.4 Основание для проведения работ

Договора №1 от 26.01.2022 года и Рабочая программа модуля МДК.06.01 Внедрение информационных систем.

1.1.1.5 Плановые сроки начала и окончания работ

Плановые срок начало работ: 17.01.2022

Плановые срок окончание работ: 20.05.2022

1.1.1.6 Источники и порядок финансирования

Финансирование проекта не планируется.

1.1.1.7 Порядок оформления и предъявления заказчику результатов работ

Работы по созданию приложения для формирования расходной накладной и автоматизации учета канцелярских товаров на складе оптовой базы сдаются разработчиком поэтапно в соответствии с календарным планом проекта. По окончании каждого из этапов работ Разработчик сдает Заказчику соответствующую отчетную документацию по данному этапу.

1.1.2 Назначение и цели создания системы

1.1.2.1 Виды автоматизируемой деятельности

Формирование оптовой или розничной цены исходя из количества и стоимости итоговой суммы приобретаемых товаров, учёт продажи товаров.

1.1.2.2 Назначение системы

Система АСУКТНСОБ предназначена для упрощения и улучшения работы склада в плане корректировки имеющихся товаров и учёта проданных товаров.

Настоящее техническое задание распространяется на разработку системы автоматизации учета канцелярских товаров на складе оптовой базы. Учёт проданных товаров за определённый период времени. Предполагается, что использовать данную систему будут сотрудники склада: менеджер по продажам и администратор отдела продаж.

1.1.2.3 Цели создания системы

Целями информационной системы являются:

* избавление сотрудников от вычислений цен при оптовой или розничной продаже;
* предоставить функцию автоматического формирования ограничений по товарам, которые есть в наличие;
* предоставить функцию автоматического создания отчётов по результатам проданных товаров за определённый период времени;
* автоматизация составления расходных накладных.
  + 1. Характеристика объекта автоматизации

Объектом автоматизации является система формирования расходной накладной из учёта количества имеющихся на складе товаров. Автоматизируется процесс отбора доступных для покупки товаров. Автоматизируется процесс формирования отчёта проданных товаров за определенный период времени. Выходными документами являются: «Расходная накладная» и «Отчёт по проданным товарам за период времени».

Предполагается, что система будет эксплуатироваться в помещении, доступном для сотрудников.

* + 1. Требования к системе
       1. Требования к системе в целом
       2. Требования к структуре и функционированию системы

В структуре системы должны находиться:

* прайс-лист товаров;
* список сотрудников;
* список поставщиков;

Предполагается следующие режимы функционирования:

* режим администратора отдела продаж;
* режим менеджера по продажам.

Разрабатываемая система предназначена для работы в непрерывном и круглосуточном режиме. Должны быть предусмотрены технологические перерывы в работе системы в соответствии с режимом работы персонала.

В дальнейшем развитие системы не предполагается

1.1.4.2.1 Требование к персоналу

Численность пользователей системы и необходимого обслуживающего

персонала уточняется на этапе разработки технического проекта. Изначально требуются:

* администратор отдела продаж;
* менеджер по продажам.

1.1.4.1.3 Показатели назначения

Система должна быть способна адаптироваться к изменению количества товаров, поставщиков.

1.1.4.1.4 Требования к надёжности

Необходимо, чтобы система обладала устойчивостью к отказам оборудования и программных систем, а также электропитания. Для надежной работы системы необходимы высоконадежные аппаратные и программные средства, а также возможность сохранения резервной копии администратором системы. Требования надежности должны быть регламентированы для следующих аварийных ситуаций:

* выход из строя аппаратных средств системы;
* отсутствие электроэнергии;
* выход из строя программных средств системы;
* неверные действия пользователей системы;
* форс мажорные обстоятельства.

1.1.4.1.5 Требования к безопасности

Все технические решения, использованные при создании системы, а также при определении требований к аппаратному обеспечению, должны соответствовать действующим нормам и правилам техники безопасности, пожарной безопасности и взрывобезопасности, а также охраны окружающей среды при эксплуатации.

Доступ к системе не смогут получить пользователи, которые не являются зарегистрированными, так как будет использоваться система авторизации.

1.1.4.1.6 Требования к эргономике и технической эстетике

Доступ к ИС должен осуществляться из мест недоступных для не авторизированных пользователей.

1.1.4.1.7 Требования к транспортабельности

Для осуществления перемещения системы на другие устройства необходимо будет осуществить новое подключение к базе данных, которая будет установлена на новом сервере, если таковой предоставится.

1.1.4.1.8 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы.

Сеть энергоснабжения должна иметь следующие параметры: напряжение 20В; частота — 50Гц.

Для обслуживания и профилактики аппаратных элементов системы необходимо привлечение инженера-электронщика либо специалиста по сетевым технологиям. Его образование должно быть исключительно высшее техническое, связанное с отладкой локальных или структурированных кабельных сетей. Специалист по плану должен уделять 1 день в месяц обслуживанию аппаратных элементов системы, либо в случае непредвиденного выхода аппаратных элементов из строя по заявке персонала компании.

Специалист по сетевым технологиям с высшим образованием должен проводить обслуживание программных элементов системы в следующих случаях:

* выход из строя программных элементов;
* при неправильном использовании программных элементов;
* по плану 1 день в месяц для проведения тестирования программных элементов.

1.1.4.1.9 Требования к защите от несанкционированного доступа

При работе с системой, необходимо, чтобы она была защищена от попыток изменения и разрушения. Система нуждается в защите информации от несанкционированного доступа. ИС защищается логином и паролем.

Существует два вида доступа:

* доступ администратора отдела продаж;
* доступ менеджера по продажам.

1.1.4.1.10 Требования по сохранности информации

Сохранность информации должна быть обеспечена в следующих случаях:

* выход из строя аппаратных систем комплекса;
* стихийные бедствия (пожар, наводнение, взрыв, землетрясение и т.п.);
* хищение носителей информации, других систем комплекса;
* ошибки в программных средствах;
* неверные действия пользователей.

Для сохранности информации необходимо предусмотреть использование блоков бесперебойного питания для защиты данных от повреждения в случае отключения питания, для надёжного хранения данных необходимо производить ежедневное резервное копирование БД на несколько дисков администратором системы.

Для выполнения операции отката и повышения надёжности хранения базы данных предусмотреть раздельное хранение двух дополнительных копий (с возможностью сохранения на различных физических носителях).

1.1.4.2 Требования к функциям и задачам

1.1.4.2.1 Функции пользователей системы

Администратор отдела продаж:

* регистрация, корректировка и удаление пользователей АИС;
* ввод, корректировка и удаление данных о поставщиках;
* ввод, корректировка и удаление данных о товарах;
* ввод, корректировка и удаление данных о товарных группах;
* создание отчета по проданным товарам.

Менеджер по продажам:

* ввод, корректировка и удаление данных о товарах;
* ввод, корректировка и удаление данных о товарных группах;
* ввод, корректировка и удаление данных о поставщиках;
* составление расходной накладной;

1.1.4.3 Требования к видам обеспечения

1.1.4.3.1 Математическое обеспечение

Математическое обеспечение не предусматривается системой.

1.1.4.3.2 Информационное обеспечение

Должно включать:

* прайс-лист товаров;
* список поставщиков;
* список сотрудников.

1.1.4.3.3 Лингвистическое обеспечение

* шрифт ввода-вывода данных – латиница;
* для создания данной системы предполагается использовать СУБД MS

SQL-сервер;

* язык программирования – С#.

Пользовательский интерфейс должен соответствовать следующим требованиям:

* эффективные интерфейсы должны быть очевидными и внушать своему пользователю чувство контроля. Необходимо, чтобы пользователь мог одним взглядом окинуть весь спектр своих возможностей, понять, как достичь своих целей и выполнить работу;
* эффективные интерфейсы не должны беспокоить пользователя внутренним взаимодействием с системой. Необходимо бережное и непрерывное сохранение работы, с предоставлением пользователю возможности отменять любые действия в любое время.

1.1.4.3.4 Программное обеспечение

Система должна работать под управление операционной системы Windows10. На компьютер должны быть установлены программные продукты MS Office, в частности, MS Word 2019 и MS Excel 2019.

1.1.4.3.5 Техническое обеспечение

* объем оперативного запоминающего устройств 8 ГБ или более;
* HDD объёмом минимум 80 GB без учета ОС;
* монитор с разрешением не ниже 1600х900;
* клавиатура - 101/102 клавиши;
* манипулятор типа «мышь».

1.1.4.3.6 Организационное обеспечение

Программа должна работать под управлением 2-х видов пользователей, которые должны иметь доступ только к определенным видам деятельности:

* администратор отдела продаж;
* менеджер по продажам.

1.2 Описание предметной области

Для разработки автоматизированного учёта продажи товаров розничным и оптовым способом необходимо разработать информационную систему, которая будет содержать следующие параметры:

* данные поставщика;
* данные сотрудников;
* прайс-лист товаров.

В сведения о поставщике заносятся следующие данные:

* наименование;
* описание;
* адрес;
* номер телефона.

В сведения о сотруднике заносятся следующие данные:

* фамилия, имя, отчество;
* дата рождения;
* паспортные данные;
* номер телефона;
* логин;
* пароль;
* должность.

В сведения о товарах заносятся следующие данные:

* наименование товара;
* цена;
* количество;
* фотография;
* товарная группа;
* наименование поставщика.

В сведения расходной накладной заносятся следующие данные:

* данные о товаре;
* данные о сотруднике;
* количество товаров;
* цена;
* цена со скидкой.

Один и тот же клиент может сделать несколько заявок на покупку. Каждый клиент имеет свой уникальный номер, поэтому на каждую заявку создаётся новая расходная накладная.

К входным данным относятся:

* список сотрудников;
* список поставщиков;
* прайс-лист товаров;

К выходным документам относятся:

* расходная накладная;
* отчет по проданным товарам за период времени.

Ограничения для АИС:

* в системе может быть зарегистрирован только один администратор;
* сотрудники входят в программу по своим логину и паролю;
* добавлять и редактировать список товаров могут только администратор и менеджер;
* фамилия, имя и отчество сотрудников вносятся в базу данных только с русской раскладки;
* серия и номер паспорта и номер телефона сотрудников и собственников вносятся в базу данных только цифрами;
* минимальный возраст сотрудника – 18 лет, максимальный – 65 лет;

Допуск к АИС имеет:

* менеджер по продажам;
* администратор отдела продаж.

При работе с АИС менеджер по продажам имеет возможность:

* вводить, корректировать и удалять данные о товарах;
* вводить, корректировать и удалять данные о товарных группах;
* вводить, корректировать и удалять данные о поставщиках;
* составлять расходную накладную.

Администратор имеет возможность:

* вводить, корректировать и удалять данные о сотрудниках;
* добавлять, удалять, изменять данные о сотрудниках;
* вводить, корректировать и удалять данные о товарах;
* вводить, корректировать и удалять данные о товарных группах;
* вводить, корректировать и удалять данные о поставщиках;
* создавать отчет по проданным товарам за период времени;

Требования к программному продукту:

* надежность работы;
* удобный интерфейс;
* конфиденциальность работы;
* сохранность данных.

1.3 Функциональная модель предметной области

Функциональная модель – это модель инфологического уровня представления, в которой акцентируется функциональный аспект моделирования предметной области.

[Функциональная декомпозиция](https://translated.turbopages.org/proxy_u/en-ru.ru.02cffed1-62b1450c-c283cd10-74722d776562/https/en.wikipedia.org/wiki/Functional_decomposition) в широком смысле относится к процессу разрешения [функциональной](https://translated.turbopages.org/proxy_u/en-ru.ru.02cffed1-62b1450c-c283cd10-74722d776562/https/en.wikipedia.org/wiki/Function_(mathematics)) связи на ее составные части таким образом, что исходная функция может быть восстановлена из этих частей с помощью [композиции функций](https://translated.turbopages.org/proxy_u/en-ru.ru.02cffed1-62b1450c-c283cd10-74722d776562/https/en.wikipedia.org/wiki/Function_composition). В общем случае этот процесс декомпозиции осуществляется либо с целью получения представления о тождестве составляющих компонентов, либо с целью получения сжатого представления глобальной функции, задача, которая выполнима только тогда, когда составляющие процессы обладают определенным уровнем модульности.

На рисунке 1.3.1 представлена функциональная модель предметной области.

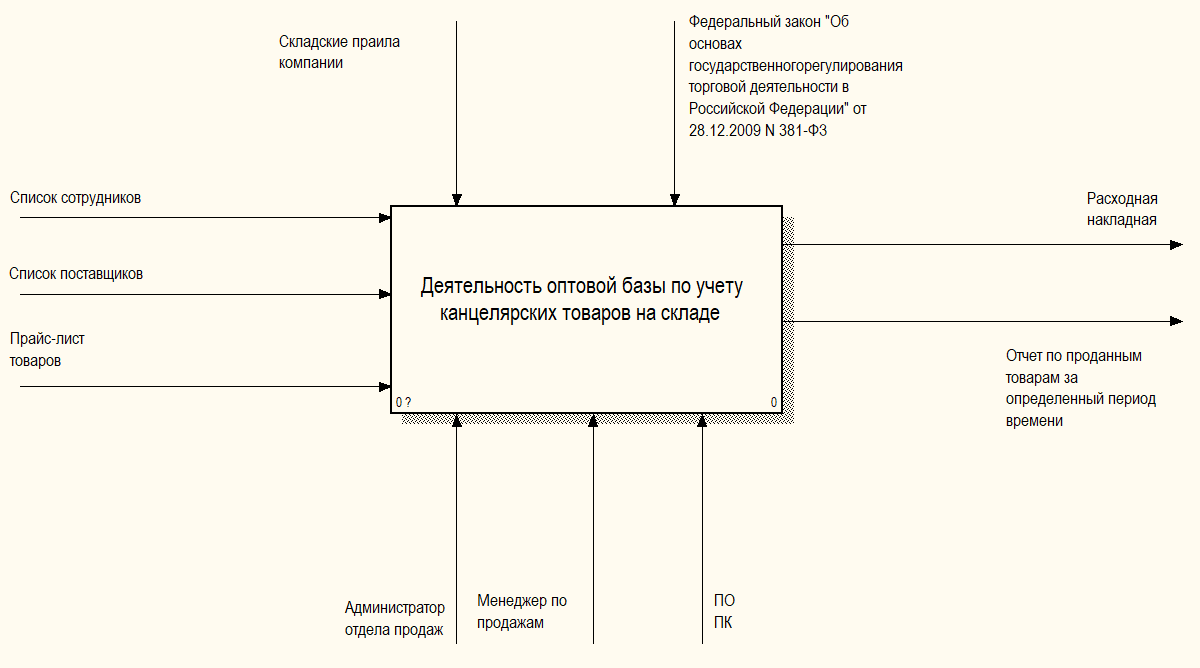


Рисунок 1.3.1 – Функциональная модель

На рисунке 1.3.2 представлена модель декомпозиции.

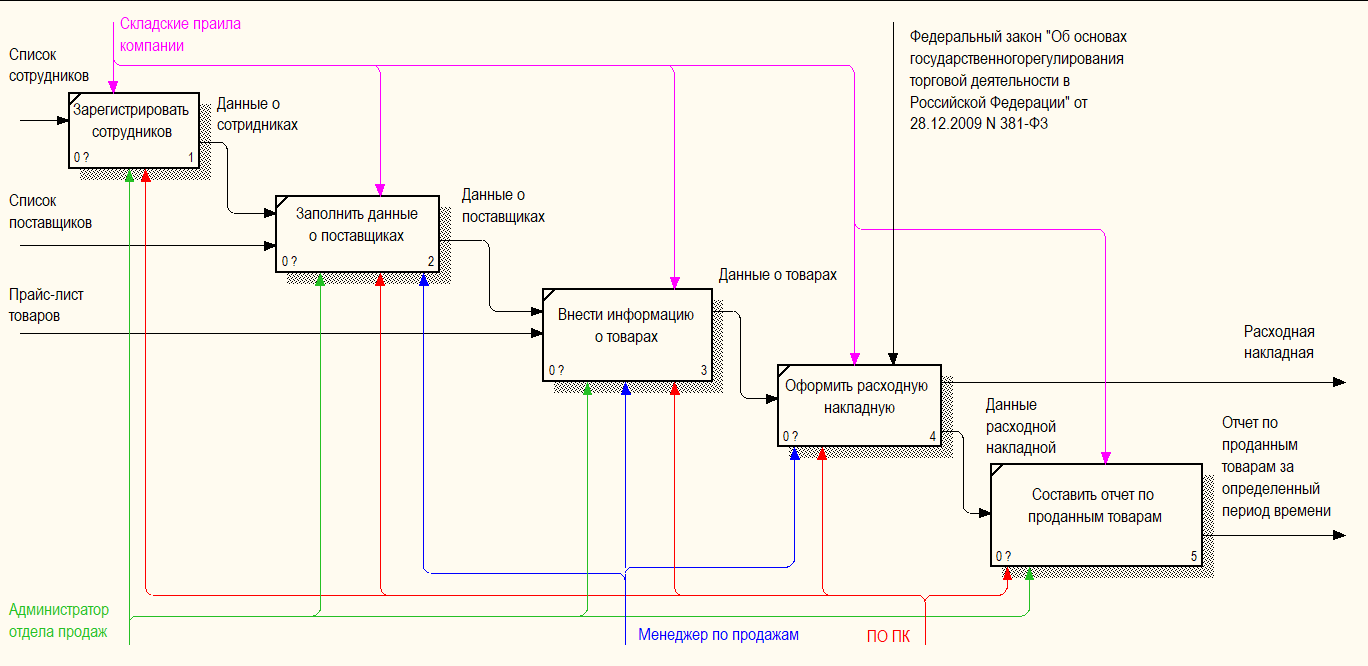


Рисунок 1.3.2 – Модель декомпозиции

* 1. Диаграмма прецедентов

Диаграмма прецедентов (диаграмма вариантов использования) в UML – диаграмма, отражающая отношения между актёрами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Прецедент – возможность моделируемой системы (часть её функциональности), благодаря которой пользователь может получить конкретный, измеримый и нужный ему результат. Прецедент соответствует отдельному сервису системы, определяет один из вариантов её использования и описывает типичный способ взаимодействия пользователя с системой. Варианты использования обычно применяются для спецификации внешних [требований](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B5%D0%B1%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%BC%D1%83_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D1%8E) к системе.

Основное назначение диаграммы – описание функциональности и поведения, позволяющее заказчику, конечному пользователю и разработчику совместно обсуждать проектируемую или существующую систему.

Диаграмма прецедентов приведена на рисунке 1.4.1

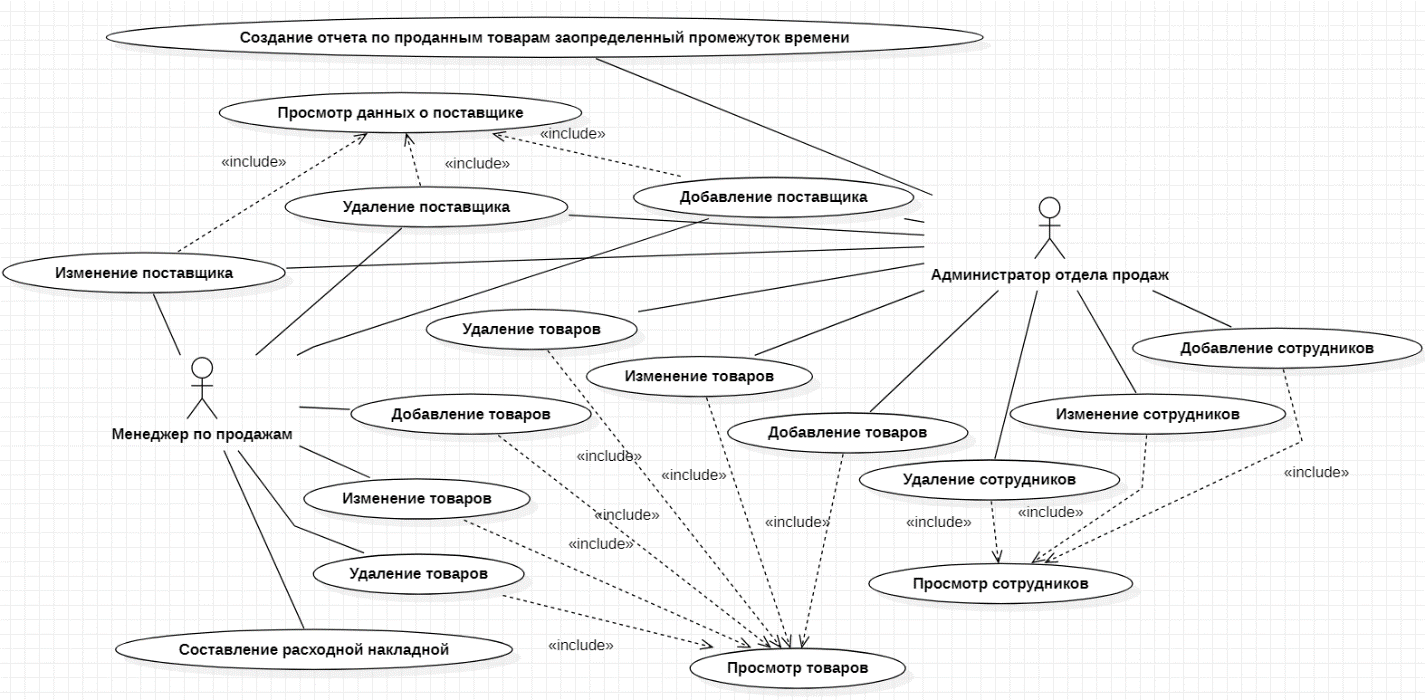


Рисунок 1.4.1 – Диаграмма прецедентов

## Описание входной информации

Входными данными для создания системы являются:

* список сотрудников;
* список поставщиков;
* прайс-лист товаров.

На основании входных документов создаются таблицы базы данных, описание которых приведено в таблице 1.5.1.

Таблица 1.5.1 – Описание входных документов

|  |  |  |
| --- | --- | --- |
| Наименование документа (шифр) | Периодичность поступления | Откуда поступает документ |
| Прайс-лист товаров | По мере поступления товара | Администратор отдела продаж |

1.6 Описание выходной информации

Результат автоматизации системы формирования расходной накладной из учёта количества имеющихся на складе товаров и составления отчёта приведено в таблице 1.6.1.

Таблица 1.6.1 – Описание выходных документов

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование документа | Периодичность выдачи документа | Кол-во экз. | Куда передаются |
| 1 | 2 | 3 | 4 |
| Расходная накладная | По мере необходимости | 1 | Менеджер по продажам |
| Отчёт по проданным товарам | По мере необходимости | 1 | Администратору отдела продаж |

Продолжение таблицы 1.6.1

|  |  |  |
| --- | --- | --- |
| Поля сортировки | Поля группировки | Итог |
| 5 | 6 | 7 |
|  | - | Сумма |
| Дата | - | Сумма |

Форма выходных документов представлена на рисунке 1.6.2 – 1.6.3

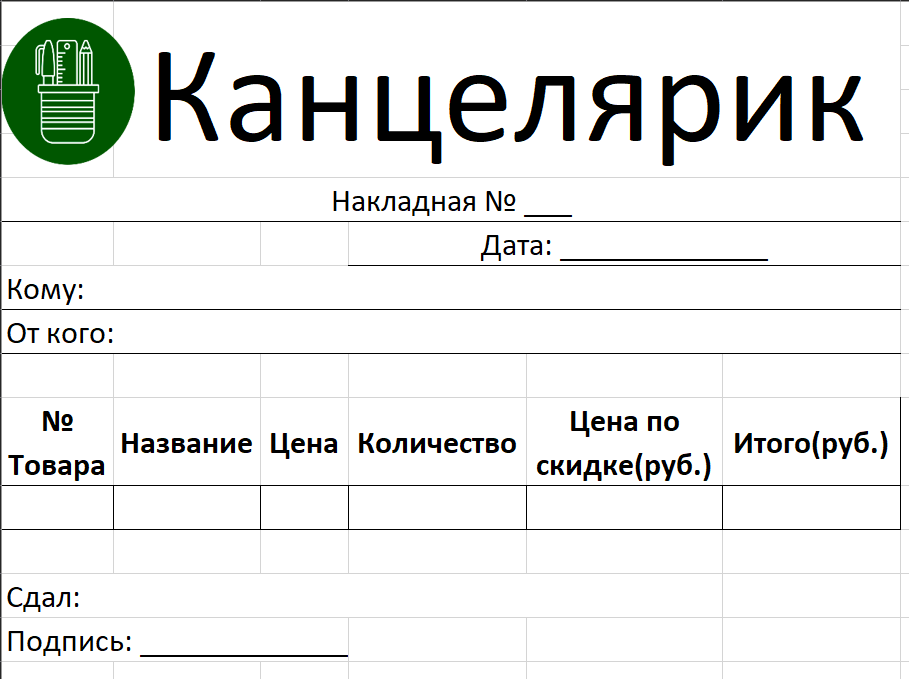
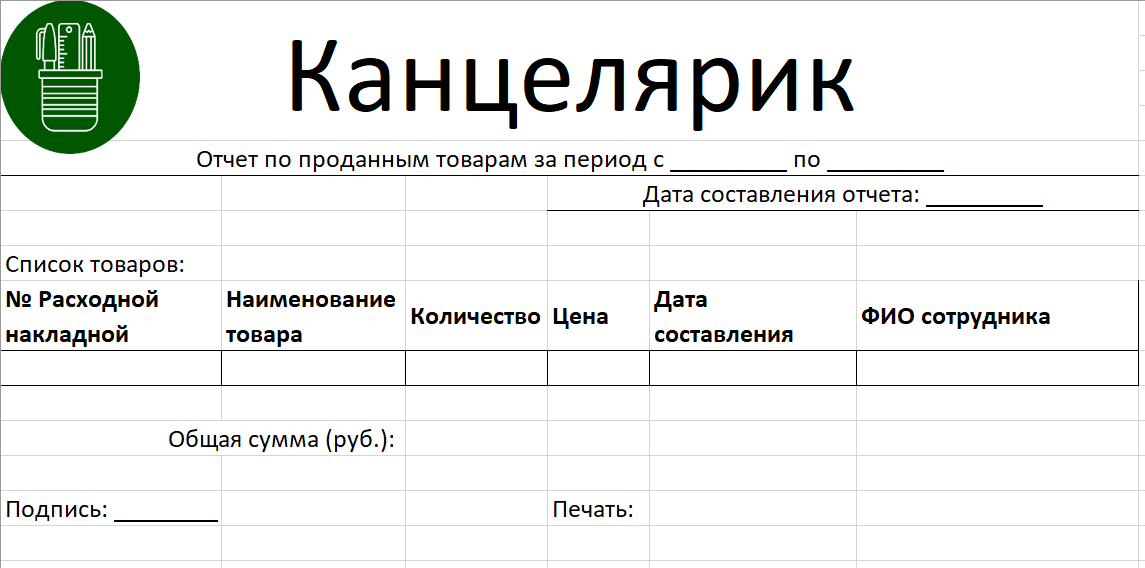


Рисунок 1.6.2 – Форма выходного документа «Расходная накладная»

Рисунок 1.6.3 – Отчет по проданным товарам за определенный период времени

1.7 Концептуальное моделирование

Концептуальное моделирование предметной области представлено на рисунке 1.7.1.

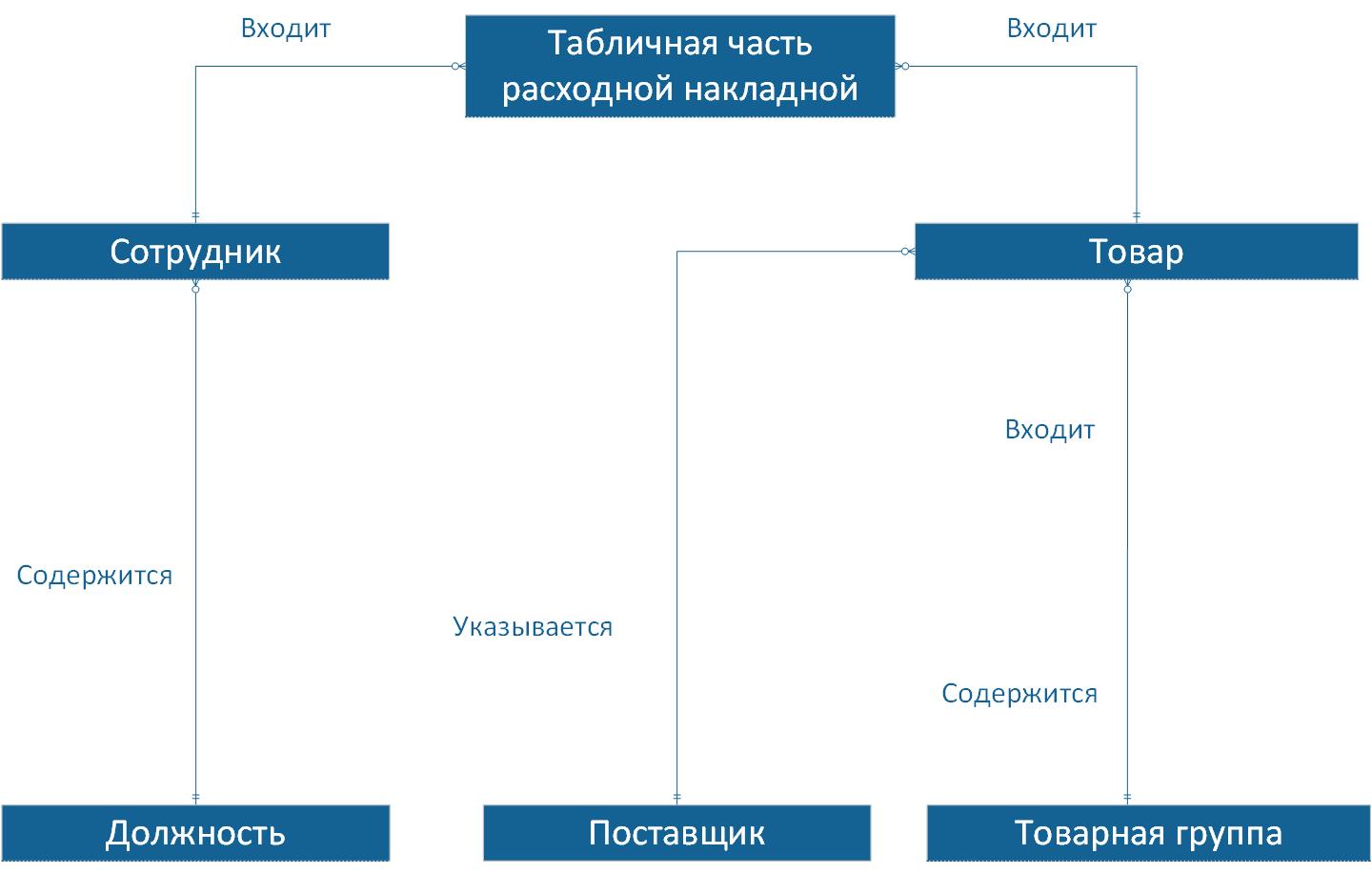


Рисунок 1.7.1 – Концептуальная модель

1.8 Логическое моделирование

Схема реляционной базы данных представлена на рисунке 1.8.1.

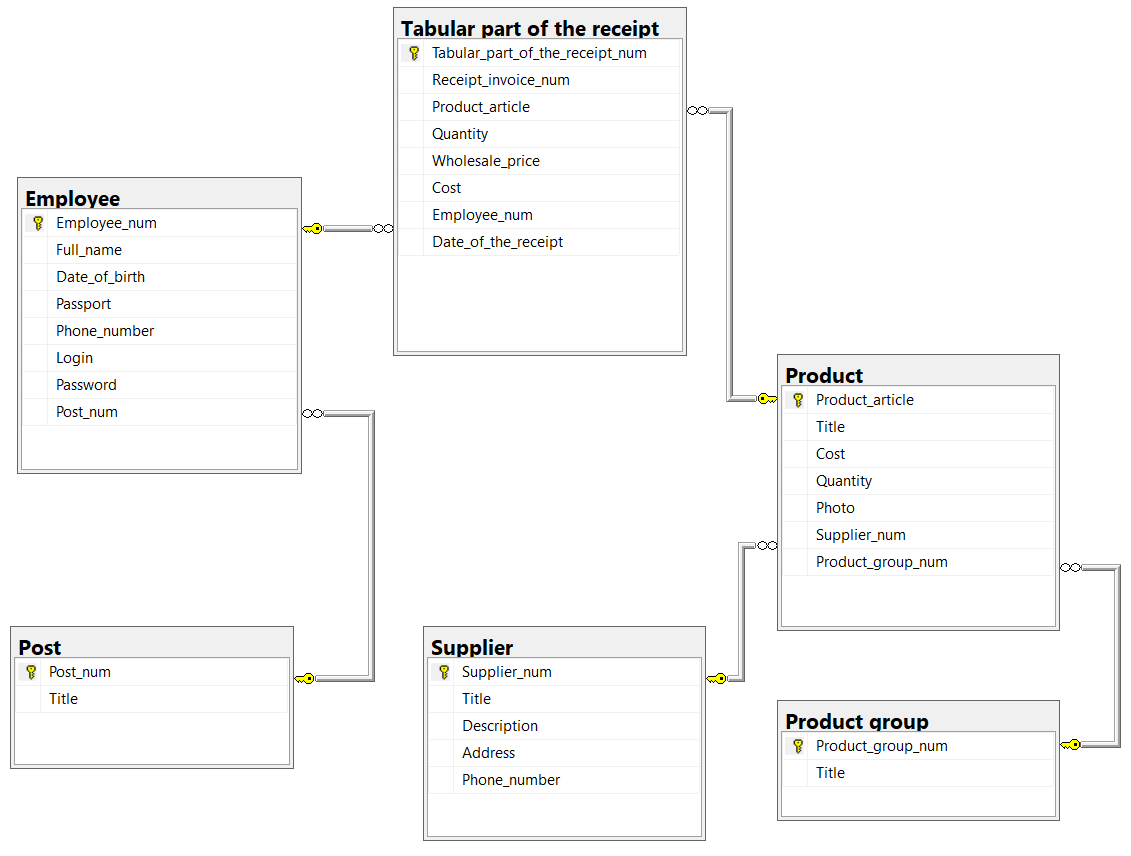


Рисунок 1.8.1 – ERD-диаграмма

1.9 Описание структуры базы данных

На основании входных документов создаются таблицы базы данных, описание которых приведено в таблицах 1.9.1 – 1.9.6.

Таблица 1.9.1 – Post (Должности)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя поля | Описание поля | Тип данных | Размер поля | Тип ключа  (РК-первичный, FK- внешний) |
| Post\_num | Код должности | INT | 4 | PK |
| Title | Название должности | VARCHAR | 50 |  |

Таблица 1.9.2 – Employee (Сотрудники)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя поля | Описание поля | Тип данных | Размер поля | Тип ключа  (РК-первичный, FK- внешний) |
| Employee\_num | Код сотрудника | INT | 4 | РК |
| Full\_name | ФИО сотрудника | VARCHAR | 150 |  |
| Date\_of\_birth | Дата рождения сотрудника | DATE | 3 |  |
| Passport | Паспортные данные | BIGINT | 8 |  |
| Phone\_number | Номер телефона | BIGINT | 8 |  |
| Login | Логин | VARCHAR | 50 |  |
| Password | Пароль | VARCHAR | 50 |  |
| Post\_num | Код должности | INT | 4 | FK |

Таблица 1.9.3 – Supplier (Поставщики)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя поля | Описание поля | Тип данных | Размер поля | Тип ключа  (РК-первичный, FK- внешний) |
| Supplier\_num | Код поставщика | INT | 4 | РК |
| Title | Наименование | VARCHAR | 100 |  |
| Description | Описание | VARCHAR | 250 |  |
| Address | Адрес | VARCHAR | 100 |  |
| Phone\_number | Номер телефона | BIGINT | 8 |  |

Таблица 1.9.4 – Product group (Товарные группы)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя поля | Описание поля | Тип данных | Размер поля | Тип ключа  (РК-первичный, FK- внешний) |
| Product\_group\_num | Код товарной группы | INT | 4 | РК |
| Title | Наименование | VARCHAR | 50 |  |

Таблица 1.9.5 – Product (Товары)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя поля | Описание поля | Тип данных | Размер поля | Тип ключа  (РК-первичный, FK- внешний) |
| Product\_article | Артикул продукта | INT | 4 | PK |
| Title | Наименование | VARCHAR | 100 |  |
| Cost | Стоимость | INT | 4 |  |
| Quantity | Количество | INT | 4 |  |
| Photo | Фото | VARCHAR | 150 |  |
| Supplier\_num | Код поставщика | INT | 4 | FK |
| Product\_group\_num | Код товарной группы | INT | 4 | FK |

Таблица 1.9.6 – Tabular part of expenditure invoice (Табличная часть расходной накладной)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя поля | Описание поля | Тип данных | Размер поля | Тип ключа  (РК-первичный, FK- внешний) |
| Tabular\_part\_of\_peceipt invoice\_num | Код табличной части расходной накладной | INT | 4 | PK |
| Receipt\_invoice\_num | Код расходной накладной | INT | 4 |  |
| Product\_article | Артикул товара | INT | 4 | FK |
| Quantity | Количество товаров | INT | 4 |  |
| Wholesale\_price | Оптовая цена | INT | 4 |  |
| Cost | Стоимость | INT | 4 |  |
| Employee\_num | Код сотрудника | INT | 4 | FK |
| Date\_of\_the\_receipt | Дата составления | DATE | 3 |  |

1.10 Контрольный пример

Контрольный пример является документом, позволяющий осуществить полную проверку функционирования разработанного программного обеспечения.

По составленной программе обрабатываются исходные данные контрольного примера. Полученные результаты сравниваются с известными результатами контрольного примера.

В таблице 1.10.1 представлены входные данные «Должности» для контрольного примера.

Таблица 1.10.1 – Контрольный пример «Должности»

|  |  |
| --- | --- |
| № должности | Наименование должности |
| 1 | Администратор |
| 2 | Менеджер |

В таблице 1.10.2 представлены входные данные «Сотрудники» для контрольного примера.

Таблица 1.10.2 – Контрольный пример «Сотрудники»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № сотрудника | Фамилия, имя, отчество | Дата рождения | Серия и номер паспорта | Номер телефона |
| 1 | 2 | 3 | 4 | 5 |
| 1 | Теплых Сергей Александрович | 26.03.2004 | 8080102456 | 89869654621 |
| 2 | Валеев Аскар Аликович | 02.12.2002 | 8080123456 | 89174578963 |
| 29 | Хасанова Лиана Ильфировна | 02.07.2003 | 8080987654 | 89871212789 |
| 30 | Коновалов Тимофей Александрович | 12.05.2003 | 8080147895 | 89271425023 |

Продолжение таблицы 1.10.2

|  |  |  |
| --- | --- | --- |
| Логин | Пароль | Должность |
| 6 | 7 | 8 |
| qw | wq | Администратор |
| ячс | счя | Менеджер |
| йц | цй | Менеджер |
| 123 | 321 | Менеджер |

В таблице 1.10.3 представлены входные данные «Товарная группа» для контрольного примера.

Таблица 1.10.3 – Контрольный пример «Товарная группа»

|  |  |
| --- | --- |
| № товарной группы | Название |
| 1 | Карандаши |
| 2 | Ручки |
| 3 | Тетради |
| 4 | Кисти |
| 5 | Альбомы |
| 6 | Фломастеры |

В таблице 1.10.4 представлены входные данные «Поставщик» для контрольного примера.

Таблица 1.10.4 – Контрольный пример «Поставщик»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № товарной группы | Наименование | Описание | Адрес | Номер телефона |
| 1 | BIC |  | Уфа, ул. Коммунистическая, 80 | 83472461719 |
| 2 | КанцМастер | Компания "КанцМастер" крупный поставщик канцелярских товаров. | Уфа, ул. Первомайская, 34 | 83476756709 |

Продолжение 1.10.4.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 3 | ОфисМаг | ОФИСМАГ — компания-лидер по комплекснУому обеспечению организаций товарами для офиса на территории Российской Федерации | Уфа, ул. Бакалинская, 27 | 83472262727 |

В таблице 1.10.5 представлены входные данные «Товар» для контрольного примера.

Таблица 1.10.5 – Контрольный пример «Товар»

|  |  |  |  |
| --- | --- | --- | --- |
| Артикул товара | Наименование | Цена | Количество |
| 1 | Фломастер | 20 | 111 |
| 3 | Карандаш | 25 | 1 |
| 4 | Ручка | 17 | 37 |
| 28 | Кисть | 55 | 750 |
| 55 | Блокнот | 90 | 69 |
| 56 | Альбом | 1000 | 800 |

Продолжение таблицы 1.10.5

|  |  |  |
| --- | --- | --- |
| Фото | Код поставщика | Код товарной группы |
| C:\Users\furuh\Pictures\Канцелярик\Товары\pngwing.com (1).png | 2 | 14 |

Продолжение таблицы 1.10.5

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| C:\Users\furuh\Pictures\Канцелярик\Товары\MitsubishiPencil.png | 7 | 3 |
| NULL | 2 | 1 |
| C:\Users\furuh\Pictures\Канцелярик\Товары\pngwing.com (3).png | 2 | 2 |
| C:\Users\furuh\Pictures\Канцелярик\Товары\pngwing.com (2).png | 3 | 5 |
| NULL | 2 | 8 |

Таблица 1.10.6 содержит выходные данные расходной накладной, который должен формироваться в ходе выполнения программы.

Таблица 1.10.6 – Выходные данные контрольного примера «Расходной накладной»

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № табличной части расходной накладной | № расходной накладной | Код товара | Количество | Цена | Итоговая стоимость со скидкой | № сотрудника |
| 51 | 1 | 1 | 20 | 20 | 400 | 29 |
| 52 | 1 | 55 | 11 | 90 | 990 | 29 |
| 53 | 1 | 56 | 105 | 850 | 89250 | 29 |

Таблица 1.10.7 содержит выходные данные контрольного примера «Расходная накладная», который должен формироваться в ходе выполнения программы.

Таблица 1.10.7 – Выходные данные контрольного примера «Расходная накладная»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Накладная № 1 | | | | | |
|  |  |  | Дата: 05.06.2022 | | |
| Кому: | | | | | |
| От кого: | | | | | |
| **№ Товара** | **Название** | **Цена** | **Количество** | **Цена со скидкой(руб.)** | **Итого(руб.)** |
| 1 | Фломастер | 20 | 20 | 20 | 400 |
| 2 | Блокнот | 90 | 11 | 90 | 990 |
| 3 | Альбом | 1000 | 105 | 850 | 89250 |
|  |  |  |  |  | 90640(руб.) |
| Сдал: Хасанова Лиана Ильфировна | | | |  |  |
| Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | | |  |  |

Таблица 1.10.8 содержит выходные данные контрольного примера «Отчет по проданным товарам за период времени», который должен формироваться в ходе выполнения программы.

Таблица 1.10.8 – Выходные данные контрольного примера «Отчет по проданным товарам за период времени»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Отчет по проданным товарам за период с 05.06.2022 по 05.06.2022 | | | | | |
|  |  |  | Дата составления отчета: 06.06.2022 | | |
| Список товаров: |  |  |  |  |  |
| **№ Расходной накладной** | **Наименование товара** | **Количество** | **Цена** | **Дата составления** | **ФИО сотрудника** |
| 1 | Фломастер | 20 | 400 | 05.06.2022 | Хасанова Лиана Ильфировна |
| 1 | Блокнот | 11 | 990 | 05.06.2022 | Хасанова Лиана Ильфировна |
| 1 | Альбом | 105 | 89250 | 05.06.2022 | Хасанова Лиана Ильфировна |
|  | Общая сумма (руб.): | | 90640 |  |  |
| Подпись: \_\_\_\_\_\_\_\_ |  |  | Печать: |  |  |

1.11 Общие требования к программному продукту

Программный продукт должен обладать следующими требованиями:

* понятность и удобство интерфейса;
* корректная реакция на ошибки;
* защита от несанкционированного доступа;
* надежное хранение и добавление данных;
* успешное выполнение основных функций программы.

Технические средства, необходимые для ввода продукта в эксплуатацию, включая наименования изготовителей:

* операционная система Windows 10 или выше;
* тип системы: x32-x64(x86);
* процессор AMD A4 или выше;
* объем оперативного запоминающего устройств 8 ГБ или более;
* HDD объёмом минимум 20 GB без учета ОС;
* монитор с разрешением не ниже 1500х800;
* клавиатура - 101/102 клавиши;
* манипулятор типа «мышь».

Для обеспечения надежности и безопасности программы были сгенерированы логин и пароль для сотрудников и администрации.

При невыполнении минимальных требований (в том числе и ОС) возможна некорректная работа программы.

Для предотвращения несанкционированного доступа к программе, используется парольная аутентификация.

2 Экспериментальный раздел

2.1 Описание программы

Программа ТуRист.exe для сотрудников отеля написана на языке C# и работает с базой данных SQL Server. Программа имеет модульную схему. Модульная схема представлена на рисунке 2.1.1.

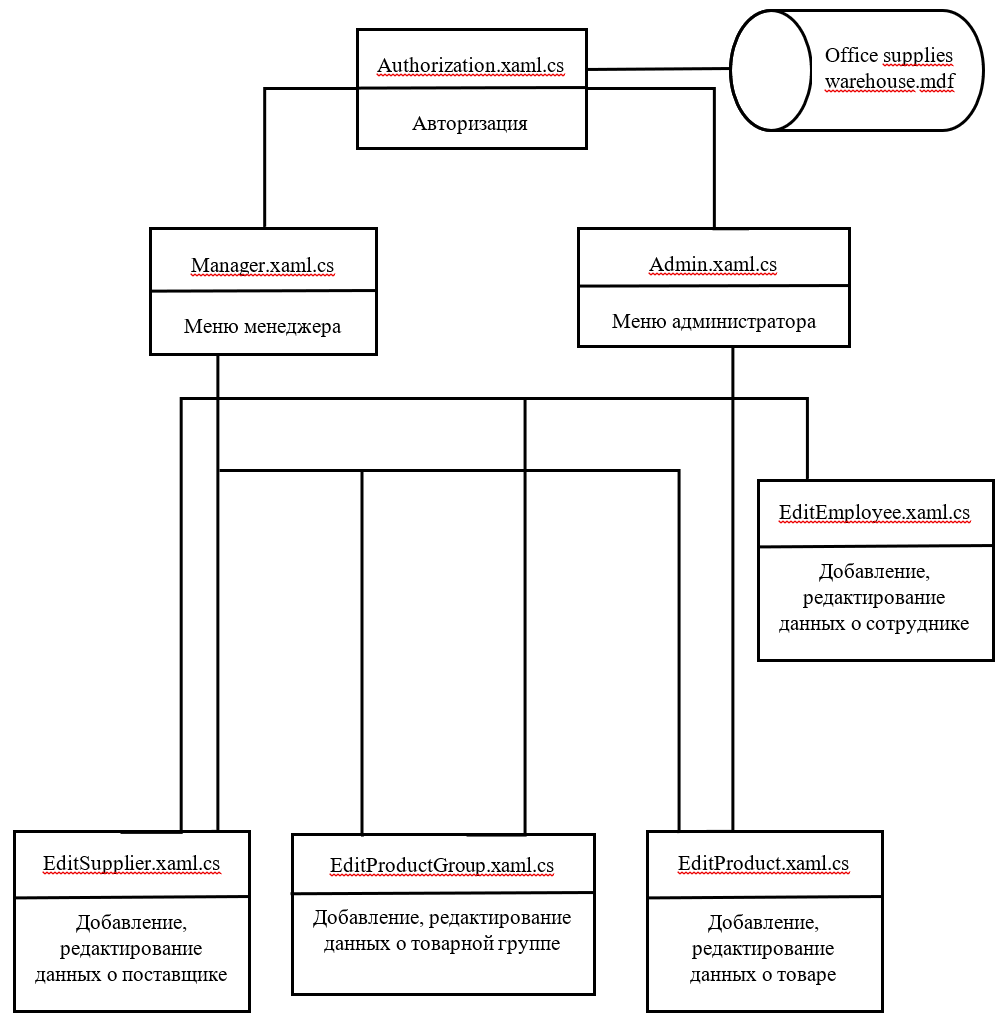


Рисунок 2.1.1 – Модульная схема программы

В таблице 2.1.1 описывается каждый модуль, перечисляются все процедуры, входящие в модуль, описываются все действия, выполняемые в каждой процедуре модуля.

Таблица 2.1.1 – Описание процедур

|  |  |
| --- | --- |
| Процедуры | Описание процедур |
| 1 | 2 |
| Authorization.xaml.cs | |
| private void Button\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку происходит проверка логина и пароля на соответствие и существование в базе данных. При неправильном вводе логина и пароля будет выведено сообщение. |
| FormAdmin.xaml.cs | |
| private void btnExitEmployee\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку, открытое окно закрывается и открывается окно авторизации |
| private void Button\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку открывается окно изменения данных о сотруднике |
| private void btnAdd\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку открывается окно добавления данных о сотруднике |
| private void tbSearch\_TextChanged(object sender, TextChangedEventArgs e) | При введении текста производится поиск сотрудников |
| private void rbWithout\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку сбрасывается сортировка |
| private void rbAsc\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о сотрудниках сортируются алфавитном порядке |
| private void rbDesc\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о сотрудниках сортируются по убыванию |
| private void btnDelete\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки удаляется выбранный сотрудник |
| private void BtnExt\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку окно пользователя закрывается и открывается авторизация. |
| private void btnExitProduct\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку окно пользователя закрывается и открывается авторизация. |
| private void tbSearchProduct\_TextChanged(object sender, TextChangedEventArgs e) | При введении текста производится поиск товаров |

Продолжение таблицы 2.1.1.

|  |  |
| --- | --- |
| 1 | 2 |
| private void rbWithoutProduct\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку сбрасывается сортировка |
| private void rbAscProduct\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку товары сортируются по возрастанию |
| private void rbDescProduct\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку товары сортируются по убыванию |
| private void btnAddProduct\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку добавления товара, открывается окно добавления товара |
| private void btnDeleteProduct\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку удаляются данные о товаре |
| private void btnEditProduct\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку изменения данных о товаре, открывается окно изменения данных |
| private void cbProduct\_SelectionChanged(object sender, SelectionChangedEventArgs e) | При нажатии можно выбрать, каким образом отфильтровать данные по товарам |
| private void btnExitSupplier\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку окно пользователя закрывается и открывается авторизация. |
| private void tbSearchSupplier\_TextChanged(object sender, TextChangedEventArgs e) | Поиск клиентов в таблице по введенным в поле для ввода символам. |
| private void rbWithoutSupplier\_Checked(object sender, RoutedEventArgs e) | При нажатии кнопки сбрасывается сортировка |
| private void rbAscSupplier\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о сотрудниках сортируются по возрастанию |
| private void rbDescSupplier\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о сотрудниках сортируются по убыванию |
| private void btnDeleteSupplier\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки удаляются данные о поставщике |
| private void btnEditSupplier\_Click(object sender, RoutedEventArgs e) | Кнопка изменения данных постащика |
| private void btnAddSupplier\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки добавления, открывается окно добавления данных о поставщике |
| private void btnExitProductGroup\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку окно пользователя закрывается и открывается авторизация. |

Продолжение таблицы 2.1.1.

|  |  |
| --- | --- |
| 1 | 2 |
| private void tbSearchProductGroup\_TextChanged(object sender, TextChangedEventArgs e) | Поиск клиентов в таблице по введенным в поле для ввода символам. |
| private void rbWithoutProductGroup\_Checked(object sender, RoutedEventArgs e) | При нажатии кнопки сбрасывается сортировка |
| private void rbAscProductGroup\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о товарной группе сортируются по возрастанию |
| private void rbDescProductGroup\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о товарной группе сортируются по убыванию |
| private void btnAddProduct\_ClickGroup(object sender, RoutedEventArgs e) | При нажатии кнопки добавления, открывается окно добавления данных о товарной группе |
| private void btnDeleteProductGroup\_Click(object sender, RoutedEventArgs e) | Кнопка нажатии на кнопку, удаляются данные о товарной группе |
| private void btnEditProductGroup\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки изменения, открывается окно изменения данных о товарной группе |
| private void Date1\_SelectedDateChanged(object sender, SelectionChangedEventArgs e) | При нажатии, выбираем дату начала расходных накладных |
| private void Date2\_SelectedDateChanged(object sender, SelectionChangedEventArgs e) | При нажатии, выбираем дату конца расходных накладных |
| private void btnExitReport\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку окно пользователя закрывается и открывается авторизация. |
| private void btnPrint\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки, формируется отчет для печати |
| private void btnSaveBackup\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку сохраняется резервная копия данных |
| private void btnBackupPath\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку сохраняется путь к резервной копии данных |

Продолжение таблицы 2.1.1.

|  |  |  |
| --- | --- | --- |
| 1 | 2 | |
| private void btnRecover\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку восстанавливаются данные из резервной копии | |
| FormManager.xaml.cs | | |
| private void btnExitProduct\_Click(object sender, RoutedEventArgs e) | | При нажатии на кнопку окно пользователя закрывается и открывается авторизация. |
| private void tbSearchProduct\_TextChanged(object sender, TextChangedEventArgs e) | | При введении текста производится поиск товаров |
| private void rbWithoutProduct\_Checked(object sender, RoutedEventArgs e) | | При нажатии на кнопку сбрасывается сортировка |
| private void rbAscProduct\_Checked(object sender, RoutedEventArgs e) | | При нажатии на кнопку товары сортируются по возрастанию |
| private void rbDescProduct\_Checked(object sender, RoutedEventArgs e) | | При нажатии на кнопку товары сортируются по убыванию |
| private void btnAddProduct\_Click(object sender, RoutedEventArgs e) | | При нажатии на кнопку добавления товара, открывается окно добавления товара |
| private void btnDeleteProduct\_Click(object sender, RoutedEventArgs e) | | При нажатии на кнопку удаляются данные о товаре |
| private void btnEditProduct\_Click(object sender, RoutedEventArgs e) | | При нажатии на кнопку изменения данных о товаре, открывается окно изменения данных |
| private void cbProduct\_SelectionChanged(object sender, SelectionChangedEventArgs e) | | При нажатии можно выбрать, каким образом отфильтровать данные по товарам |
| private void btnExitSupplier\_Click(object sender, RoutedEventArgs e) | | При нажатии на кнопку окно пользователя закрывается и открывается авторизация. |
| private void tbSearchSupplier\_TextChanged(object sender, TextChangedEventArgs e) | | Поиск клиентов в таблице по введенным в поле для ввода символам. |

Продолжение таблицы 2.1.1.

|  |  |
| --- | --- |
| 1 | 2 |
| private void rbWithoutSupplier\_Checked(object sender, RoutedEventArgs e) | При нажатии кнопки сбрасывается сортировка |
| private void rbAscSupplier\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о сотрудниках сортируются по возрастанию |
| private void rbDescSupplier\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о сотрудниках сортируются по убыванию |
| private void btnDeleteSupplier\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки удаляются данные о поставщике |
| private void btnEditSupplier\_Click(object sender, RoutedEventArgs e) | Кнопка изменения данных постащика |
| private void btnAddSupplier\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки добавления, открывается окно добавления данных о поставщике |
| private void btnExitProductGroup\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку окно пользователя закрывается и открывается авторизация. |
| private void tbSearchProductGroup\_TextChanged(object sender, TextChangedEventArgs e) | Поиск клиентов в таблице по введенным в поле для ввода символам. |
| private void rbWithoutProductGroup\_Checked(object sender, RoutedEventArgs e) | При нажатии кнопки сбрасывается сортировка |
| private void rbAscProductGroup\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о товарной группе сортируются по возрастанию |
| private void rbDescProductGroup\_Checked(object sender, RoutedEventArgs e) | При нажатии на кнопку данные о товарной группе сортируются по убыванию |
| private void btnAddProduct\_ClickGroup(object sender, RoutedEventArgs e) | При нажатии кнопки добавления, открывается окно добавления данных о товарной группе |
| private void btnDeleteProductGroup\_Click(object sender, RoutedEventArgs e) | Кнопка нажатии на кнопку, удаляются данные о товарной группе |

Продолжение таблицы 2.1.1.

|  |  |
| --- | --- |
| 1 | 2 |
| private void btnEditProductGroup\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки изменения, открывается окно изменения данных о товарной группе |
| private void btnAddReportProduct1\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки добавляется 1 товар в корзину |
| private void btnAddReportProduct10\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки добавляется 10 товаров в корзину |
| private void btnAddReportProduct100\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки добавляется 100 товаров в корзину |
| private void btnDeleteReportProduct1\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки из корзины удаляется 1 товар |
| private void btnDeleteReportProduct10\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки из корзины удаляется 10 товаров |
| private void btnDeleteReportProduct100\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки из корзины удаляется 100 товаров |
| private void btnClear\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку корзина полностью очищается |
| private void btnPrint\_Click(object sender, RoutedEventArgs e) | При нажатии на кнопку, составляется расходная накладная для печати |
| private void btnDeleteReportRow\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки полностью удаляет строку из корзины |
| FormEditEmployee.xaml.cs | |
| private void Button\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки данные о сотруднике сохраняются |
| private void Button\_Click\_1(object sender, RoutedEventArgs e) | При нажатии кнопки окно редактирования закрывается |

Продолжение таблицы 2.1.1.

|  |  |
| --- | --- |
| 1 | 2 |
| FormEditProduct.xaml.cs | |
| private void Button\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки данные о сотруднике сохраняются |
| private void Button\_Click\_1(object sender, RoutedEventArgs e) | При нажатии кнопки окно редактирования закрывается |
| private void btnSelectPhoto\_Click(object sender, RoutedEventArgs e) | При нажатии открывается проводник для выбора фотографии |
| private void Button\_Click(object sender, RoutedEventArgs e) | При нажатии удаляет фотографию |
| FormEditProductGroup.xaml.cs | |
| private void Button\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки данные о сотруднике сохраняются |
| private void Button\_Click\_1(object sender, RoutedEventArgs e) | При нажатии кнопки окно редактирования закрывается |
| FormEditSupplier.xaml.cs | |
| private void Button\_Click(object sender, RoutedEventArgs e) | При нажатии кнопки данные о сотруднике сохраняются |
| private void Button\_Click\_1(object sender, RoutedEventArgs e) | При нажатии кнопки окно редактирования закрывается |

2.2 Протокол тестирования программного продукта

В ходе тестирования программного продукта на корректных и некорректных данных не было обнаружено ошибок, которые влияли бы на работу самого программного продукта и всей системы.

Данный программный продукт удовлетворяет всем предъявленным требованиям, имеет комфортный интерфейс и интуитивно понятный функционал, исключает появления системных ошибок.

В таблице 2.2.1 представлена общая информация о тестировании. В таблицах 2.2.2 – 2.2.6 представлены протоколы тестирования: тестирование авторизации на корректных данных, тестирование авторизации на некорректных данных, авторизации с незаполненными данными.

Таблица 2.2.1 – Общая информация о тестировании

|  |  |
| --- | --- |
| Название проекта | Канцелярик |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |

В таблице 2.2.2 представлен протокол тестирования «Проверка формы авторизации».

Таблица 2.2.2 – Протокол тестирования «Проверка формы авторизации».

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Канцелярик |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Наименование проекта | Проверка формы авторизации |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |
| Test Case # | testcase #1 |
| Приоритет тестирования *(Малый/Средний/высокий)* | Высокий |
| Наименование тестирования/ Имя | Проверка формы авторизации с неправильным логином и паролем |

Продолжение таблицы 2.2.2.

|  |  |
| --- | --- |
| 1 | 2 |
| Резюме испытания | Необходимо добиться корректного поведения программы при вводе неправильных данных |
| Шаги тестирования | Вводим неверные данные для того, чтобы убедиться, что программа корректно отреагирует при авторизации |
| Данные тестирования | Логин – 123  Пароль – 456 |
| Ожидаемый результат | Вывод сообщения: «Пользователь не найден» |
| Фактический результат | Вывод сообщения: «Пользователь не найден» |
| Предпосылки | Запуск программы |
| Постусловия | Форма становится доступной для ввода логина и пароля |
| Статус *(Pass/Fail)* | Pass |
| Комментарии | Работает |

На рисунке 2.2.3 показан ввод неверных данных в поля формы авторизации при тестировании «Проверка формы авторизации».

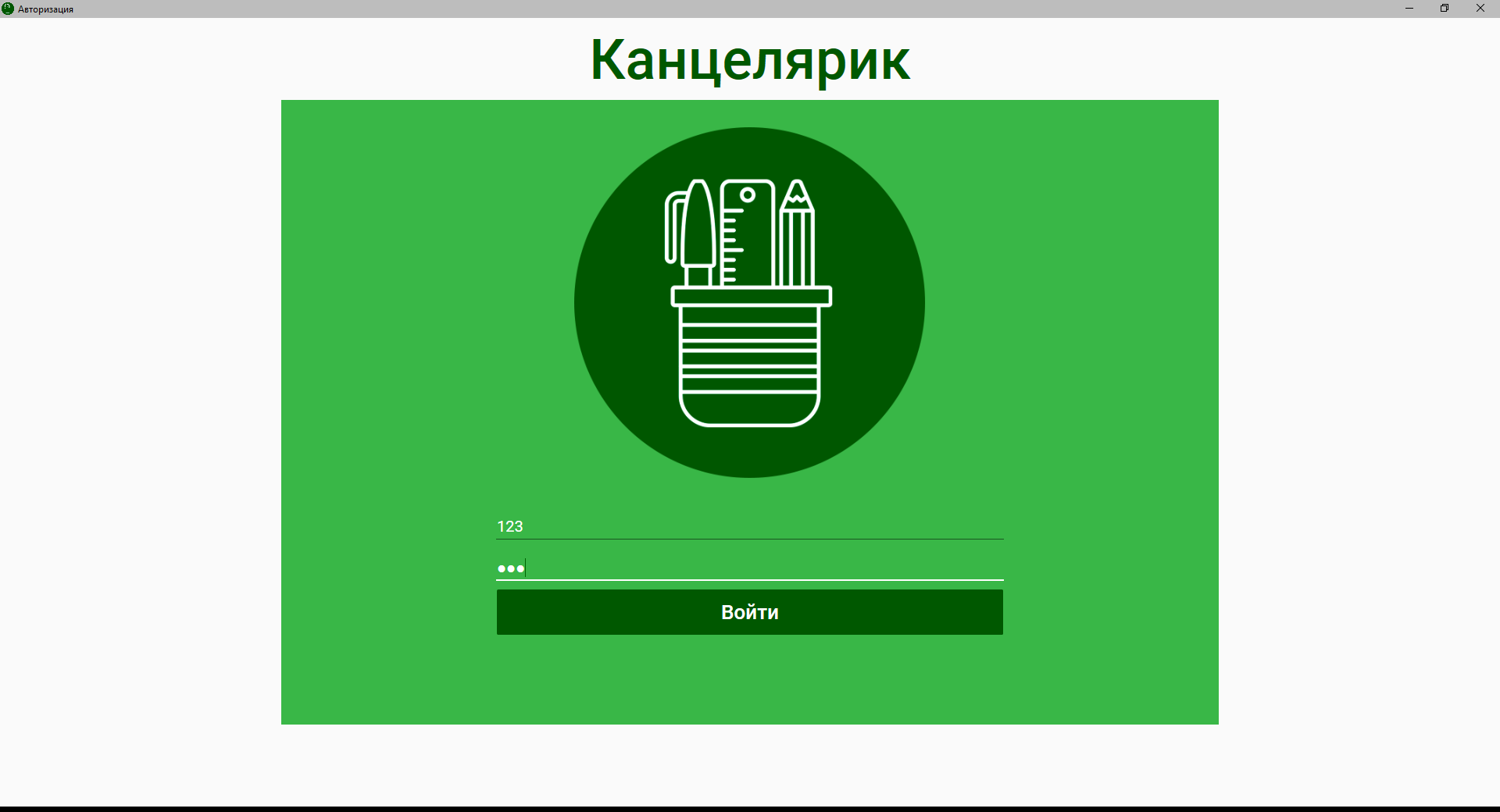


Рисунок 2.2.1 – Неверные данные в форме входа

На рисунке 2.2.2 показано сообщение, при вводе неверных данных в поля формы входа при тестировании «Проверка формы авторизации».

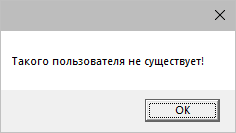


Рисунок 2.2.2 – Вывод сообщения при вводе неверных данных в форме входа

В таблице 2.2.3 представлен протокол тестирования «Проверка формы авторизации».

Таблица 2.2.3 – Протокол тестирования «Проверка формы авторизации»

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| 1 | 2 |
| Название проекта | Канцелярик |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Наименование проекта | Проверка формы авторизации |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |
| Test Case # | testcase #2 |
| Приоритет тестирования *(Малый/Средний/высокий)* | Высокий |
| Наименование тестирования/ Имя | Проверка формы авторизации с верными логином и паролем |
| Резюме испытания | Необходимо добиться корректного поведения программы при вводе неправильных данных |
| Шаги тестирования | Вводим действительные данные для того, чтобы убедиться, что программа корректно отреагирует при авторизации |

Продолжение таблицы 2.2.3.

|  |  |
| --- | --- |
| 1 | 2 |
| Данные тестирования | Логин - qw  Пароль - wq |
| Ожидаемый результат | Открытие окна администратора отдела продаж |
| Фактический результат | Открытие окна администратора отдела продаж |
| Предпосылки | Запуск программы |
| Постусловия | Открывается окно администратора |
| Статус *(Pass/Fail)* | Pass |
| Комментарии | Работает |

На рисунке 2.2.3 показан ввод верных данных в поля формы авторизации при тестировании «Проверка формы авторизации».

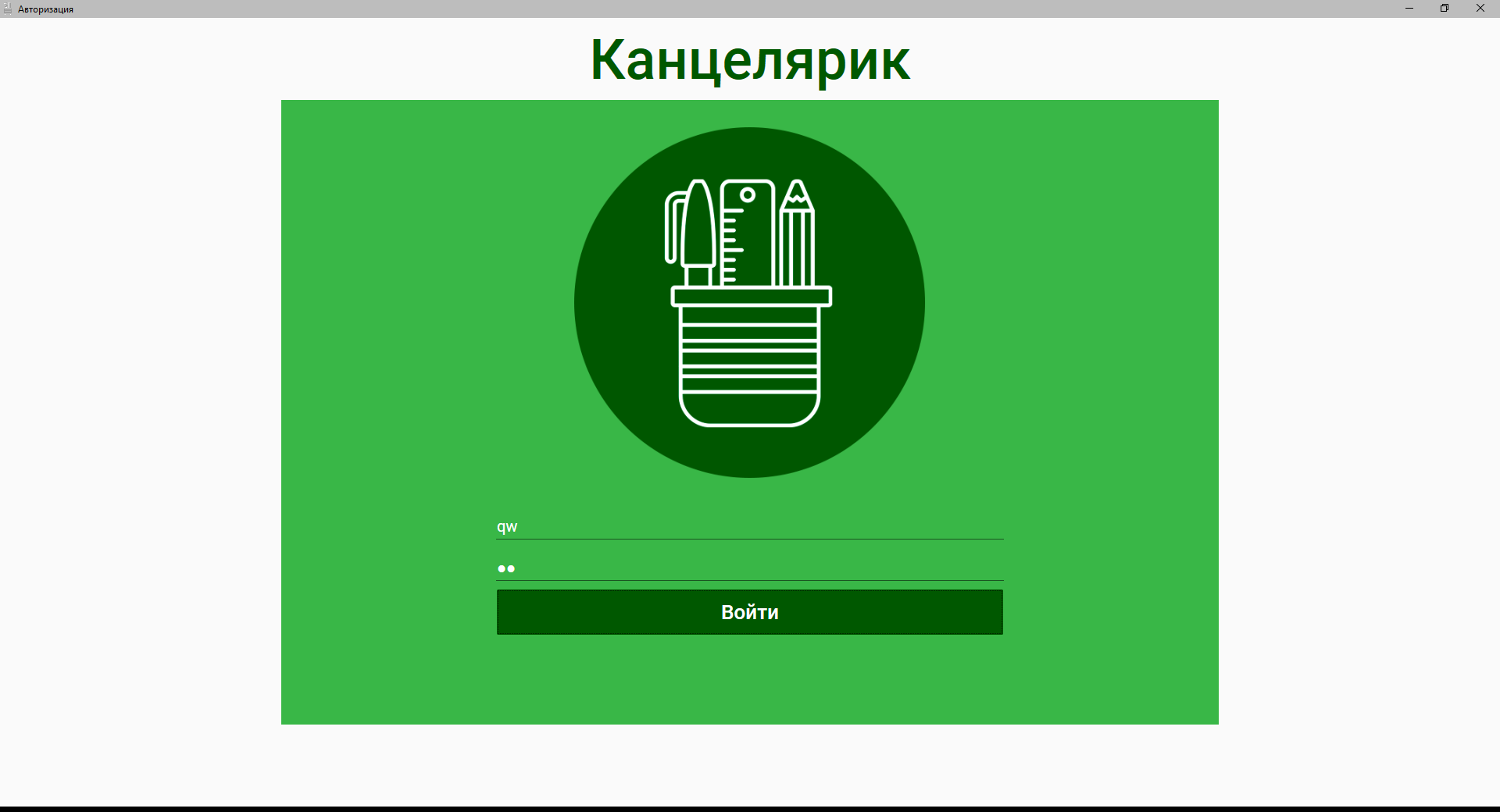


Рисунок 2.2.3 – Верные данные в форме входа

На рисунке 2.2.4 показано сообщение, при вводе верных данных в поля формы входа при тестировании «Проверка формы авторизации».

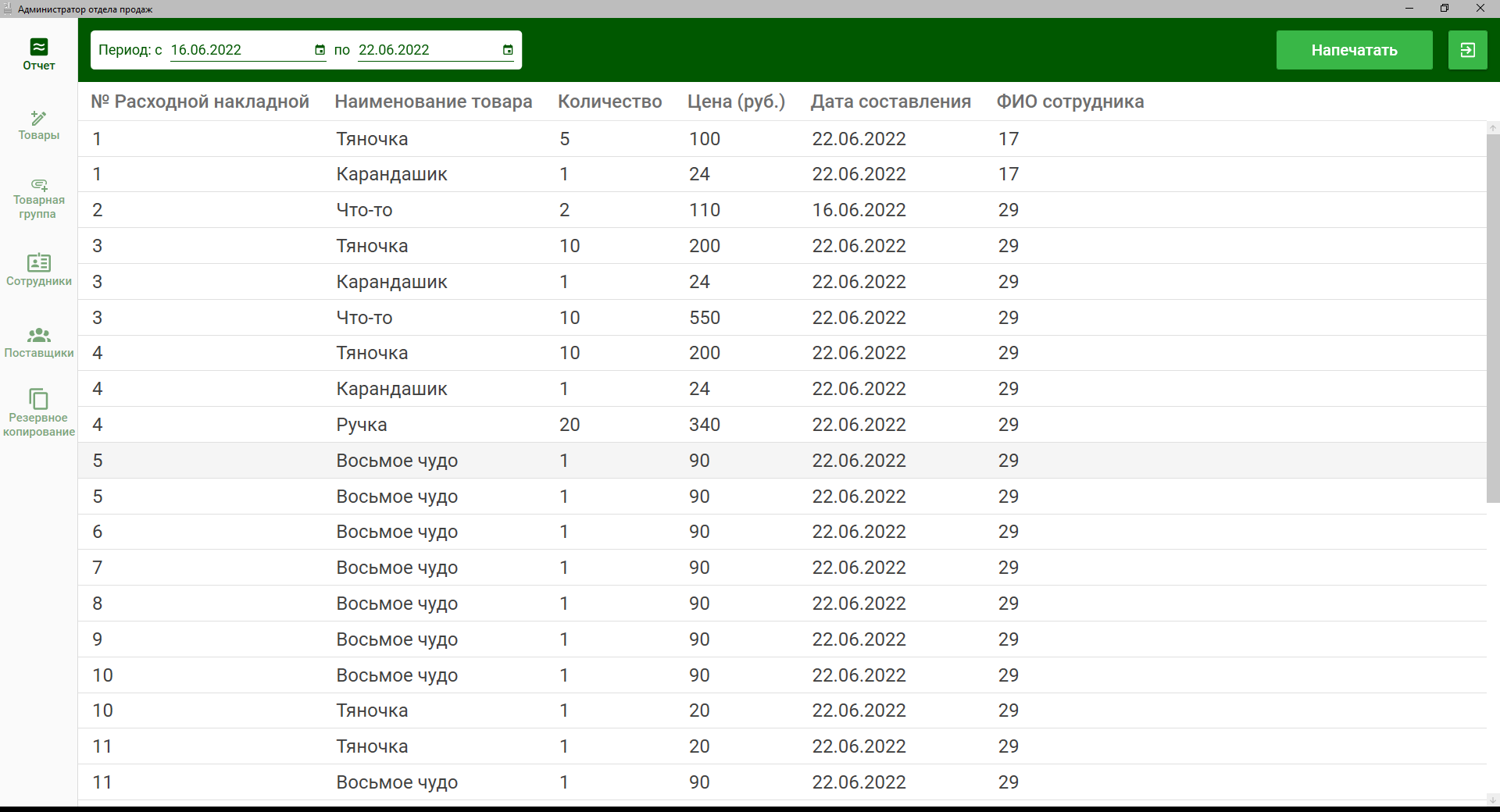


Рисунок 2.2.4 – Окно администратора отдела продаж

В таблице 2.2.4 представлен протокол тестирования «Удаление администратором самого себя».

Таблица 2.2.4 – Протокол тестирования «Удаление администратором самого себя»

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Канцелярик |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Наименование проекта | Удаление администратором самого себя |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |
| Test Case # | testcase #3 |
| Приоритет тестирования *(Малый/Средний/высокий)* | Высокий |
| Название тестирования/ Имя | Проверка возможности удаления администратором отдела продаж самого себя |
| Резюме испытания | Проверить корректность работы программы при попытке администратора удалить самого себя |

Продолжение таблицы 2.2.4.

|  |  |
| --- | --- |
| 1 | 2 |
| Шаги тестирования | Входим в систему как администратор отдела продаж. Ищем в списке сотрудников администратора отдела продаж. Пытаемся удалить его |
| Данные тестирования | Удаление |
| Ожидаемый результат | Вывод сообщения: «Вы не можете удалить самого себя!» |
| Фактический результат | Вывод сообщения: «Вы не можете удалить самого себя!» |
| Предпосылки | Запуск программы |
| Постусловия | Система работает корректно |
| Статус *(Pass/Fail)* | pass |
| Комментарии | Работает |

На рисунке 2.2.5 показан вывод сообщения

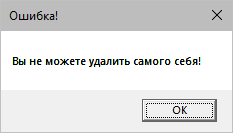


Рисунок 2.2.5 – Удаление администратором самого себя

В таблице 2.2.5 представлен протокол тестирования «Создание отчёта по проданным товарам».

Таблица 2.2.5 – Протокол тестирования «Создание отчёта по проданным товарам»

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Канцелярик |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Наименование проекта | Создание отчёта по проданным товарах |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |

Продолжение таблицы 2.2.5.

|  |  |
| --- | --- |
| 1 | 2 |
| Test Case # | testcase #4 |
| Приоритет тестирования *(Малый/Средний/высокий)* | Высокий |
| Название тестирования/ Имя | Проверка создание отчёта по проданным товарам |
| Резюме испытания | Проверить правильность создания отчета по проданным товарам |
| Шаги тестирования | 1. Выбрать период  2. Нажатие на кнопку «Напечатать» |
| Данные тестирования | Кнопка «Напечатать» |
| Ожидаемый результат | Отчёт по проданным товарам |
| Фактический результат | Отчёт по проданным товарам |
| Предпосылки | Выполнение программы |
| Постусловия | Система работает корректно |
| Статус *(Pass/Fail)* | pass |
| Комментарии | Работает |

На рисунке 2.2.6 представлен предварительный просмотр «Отчёт по проданным товарам».

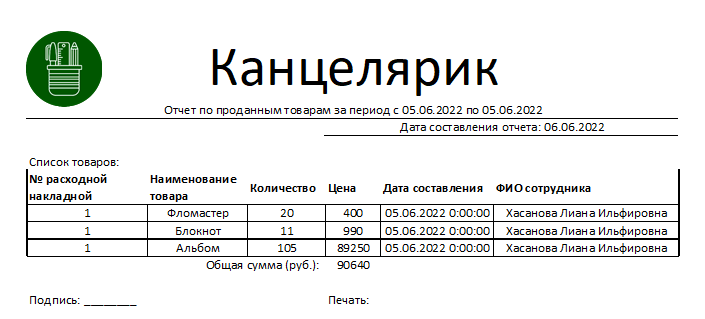


Рисунок 2.2.6 – Предварительный просмотр «Отчёт по проданным товарам»

В таблице 2.2.6 представлен протокол тестирования «Формирование расходной накладной».

Таблица 2.2.6 – Протокол тестирования «Формирование расходной накладной»

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Канцелярик |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Наименование проекта | Формирование расходной накладной |
| Номер версии | 1.0 |
| Имя тестера | Теплых Сергей Александрович |
| Даты тестирования | 19.05.2022 |
| Test Case # | testcase #5 |
| Приоритет тестирования *(Малый/Средний/высокий)* | Высокий |
| Название тестирования/ Имя | Проверка формирования расходной накладной |
| Резюме испытания | Проверить правильность создания расходной накладной |
| Шаги тестирования | 1. Добавить товары в корзину  2. Нажатие на кнопку «Напечатать» |
| Данные тестирования | Кнопка «Напечатать» |
| Ожидаемый результат | Расходная накладная |
| Фактический результат | Расходная накладная |
| Предпосылки | Выполнение программы |
| Постусловия | Система работает корректно |
| Статус *(Pass/Fail)* | pass |
| Комментарии | Работает |

На рисунке 2.2.7 представлен предварительный просмотр «Расходная накладная».



Рисунок 2.2.7 – Предварительный просмотр «Расходная накладная»

2.3 Руководство пользователя

При запуске программы будет показана форма входа, где пользователю необходимо ввести логин и пароль.

На рисунке 2.3.1 показано окно входа.

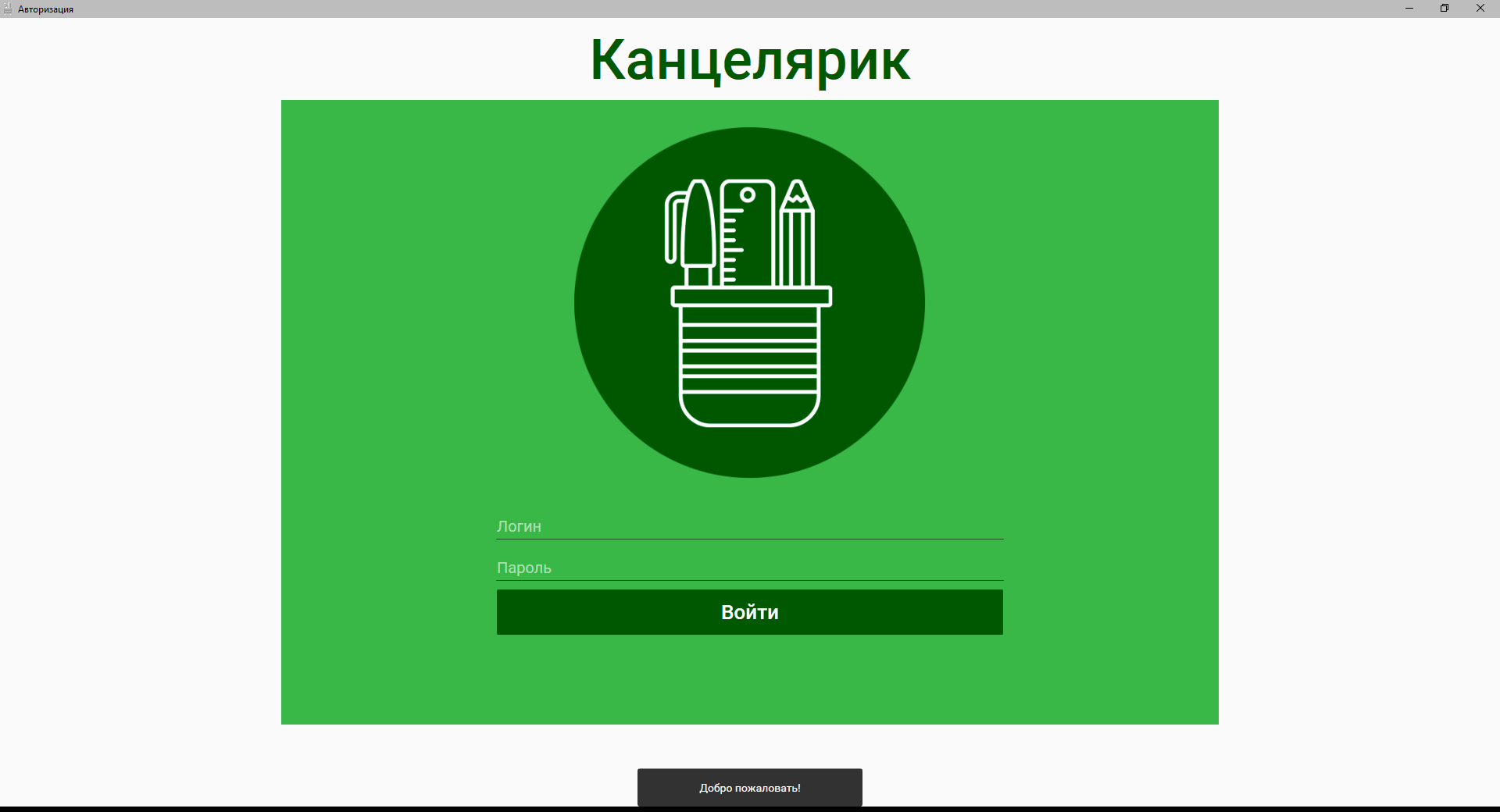


Рисунок 2.3.1 – Окно входа

При вводе логина и пароля менеджера по продажам, будет выполнен вход в программу и будет запущено окно менеджера по продажам. В левой части расположены такие кнопки, как «Расходная накладная», «Товары», «Товарная группа», «Поставщики». Изначально окно открывается во вкладке «Расходная накладная», в верхней части которой расположен поиск товаров и три кнопки: «Очистить корзину», которая удаляет все добавленные товары в корзину, «Напечатать», нужна для того, чтобы составлять расходную накладную и «Сменить пользователя», необходимая для смены вида пользователя.

На рисунке 2.3.2 показано окно менеджера по продажам.

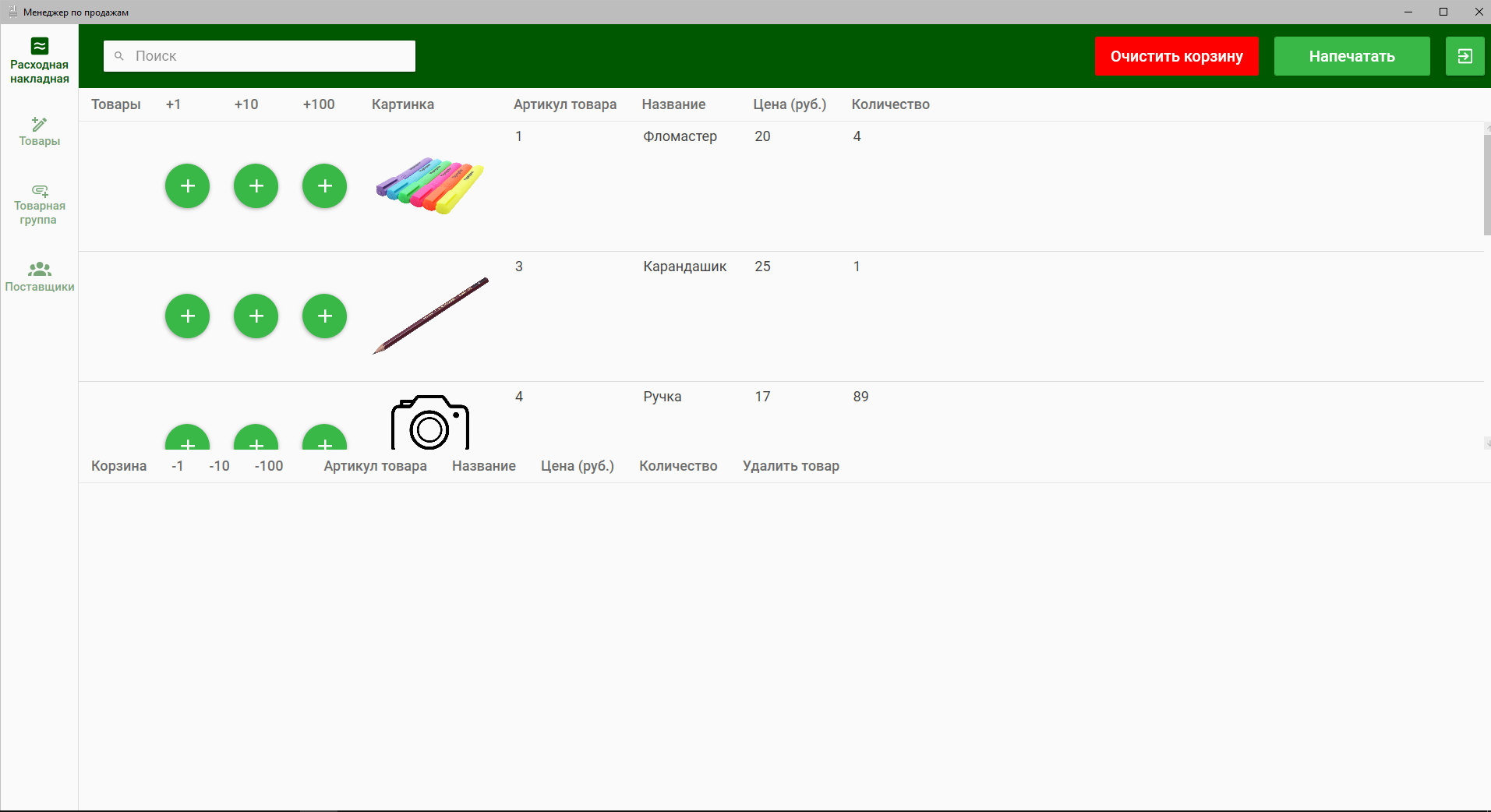


Рисунок 2.3.2 – Окно менеджера по продажам

Ниже расположен список товаров. Эти товары мы можем добавлять в корзину с помощтю специальных кнопок, которые находятся рядом с картинкой товара. Для того, чтобы добавить товар в корзину вам необходимо нажимать на кнопки, которых есть три вида. Первая – добавляет один товар в корзину, вторая – добавляет сразу десять, и третья – добавляет сто товаров. После того, как вы добавите данные в корзину, вы можете сформировать расходную накладную. Если вы хотите убрать какое-то количество товаров, то нужно воспользоваться кнопками, похожими на добавление, но которые удаляют.

Следующее окно - «Товары»

На рисунке 2.3.3 показано окно «Товары».

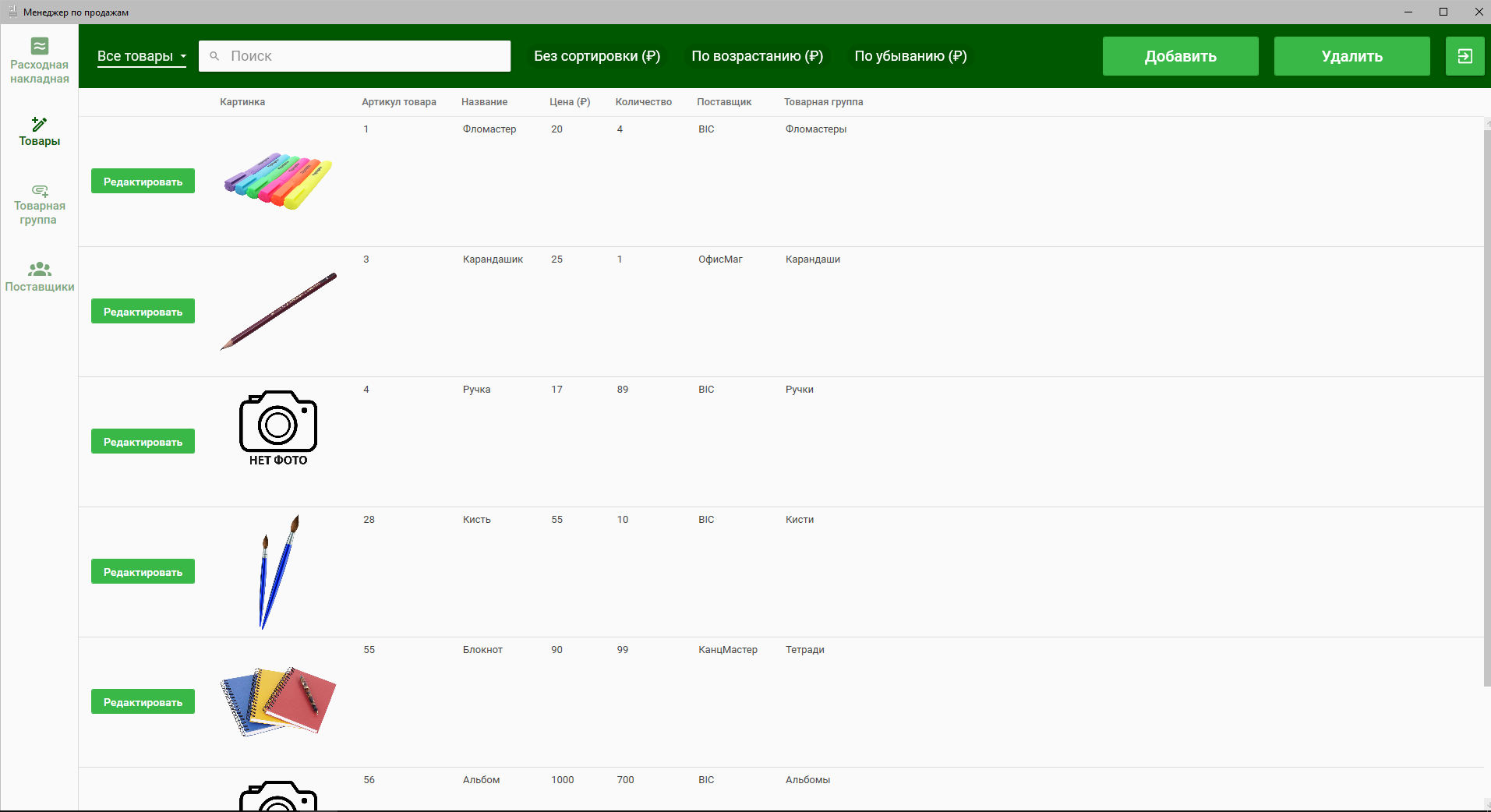


Рисунок 2.3.3 – Окно «Товары»

Расскрывающийся список с надписью «Все товары» позволяет нам отфильтровать товары по их товарной группе. Также мы можем найти нужный нам товар по его названию. Сортировка может переключаться между тремя видами: «Без сортировки», «По возрастанию» и «По убыванию». Сортировка выполняется по цене товара. Кнопка «Добавить», расположенная в верхнем-правом углу, необходима для добавления новых товаров. Кнопка «Удалить» нужна для удаления уже имеющихся товаров. Кнопка «Редактировать», находящаяся рядом с картинкой товаров позволяет изменить данные товара, который уже имеется на складе. В окне редактирования мы можем поменять данные товара, в том числе поставщика и товарную группу. Также можно выбрать фотографию для товара. Если у товара нету фотографии, то вместо нее ставится заглушка. Кнопка «Выбрать фото» необходима для выбора фотографии, кнопка «Сохранить» нужна для сохранения измененных данных, а кнопка «Назад» закрывает окно. Красная круглая кнопка рядом с путем нужна для удаления фотографии.

На рисунке 2.3.4 показано окно редактирования товара.

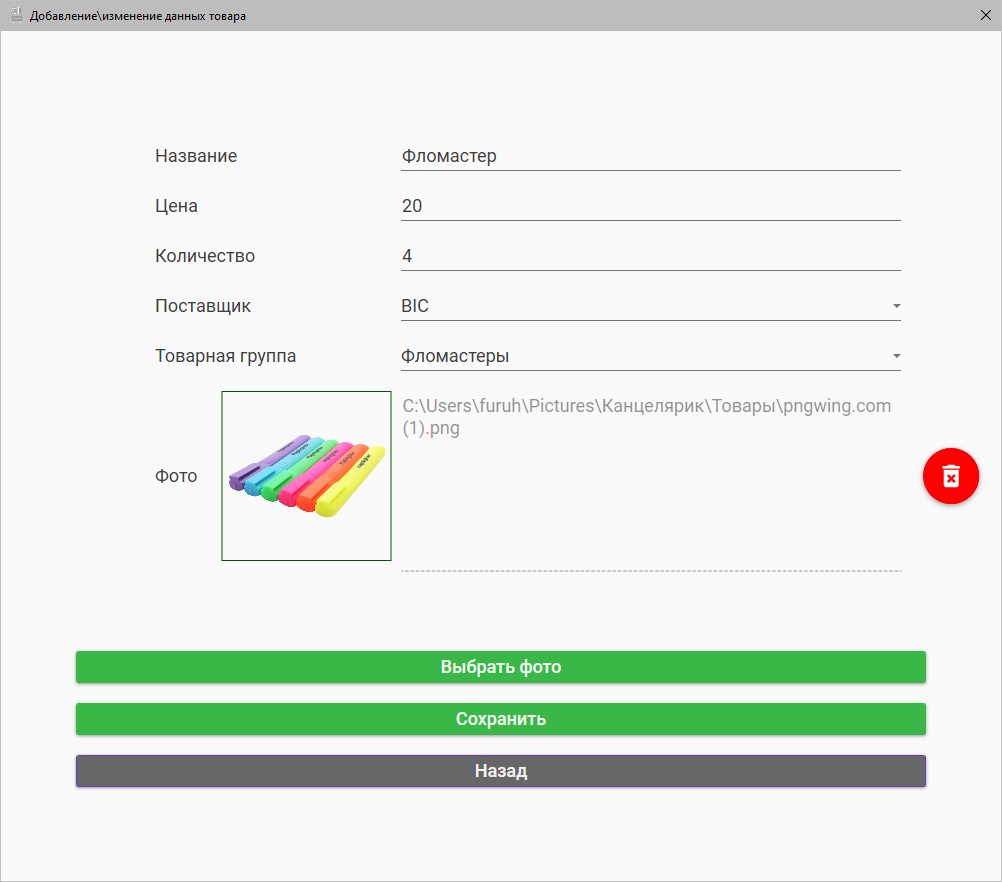


Рисунок 2.3.4 – Окно редактирования товара

На рисунке 2.3.5 показан товар без фотографии.

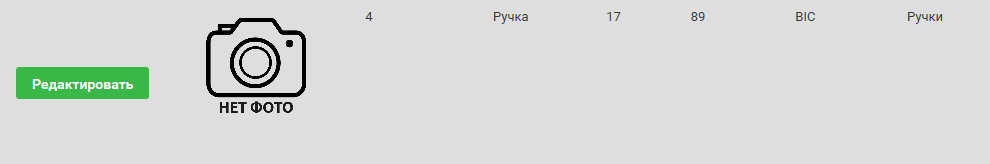


Рисунок 2.3.5 – Товар без фотографии

Следующее окно – «Товарная группа»

На рисунке 2.3.6. показано окно «Товарная группа»

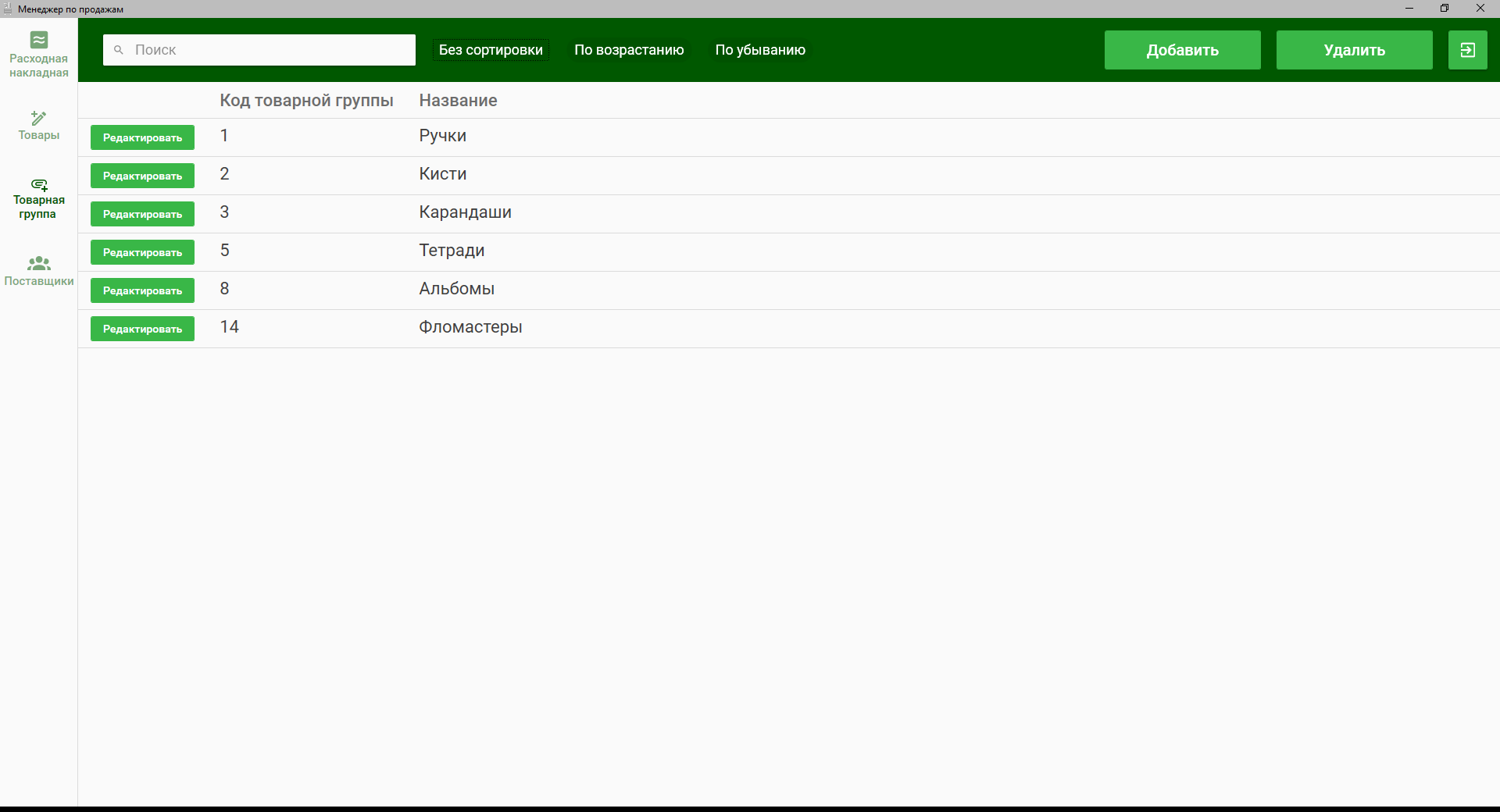


Рисунок 2.3.6 – Окно «Товарная группа»

В этом окне, аналогично и окну с товарами мы можем сортировать товарные группы, существлять поиск, добавлять, изменять и удалять товарные группы.

Следующее окно – «Поставщики»

На рисунке 2.3.7 изображено окно «Поставщики»

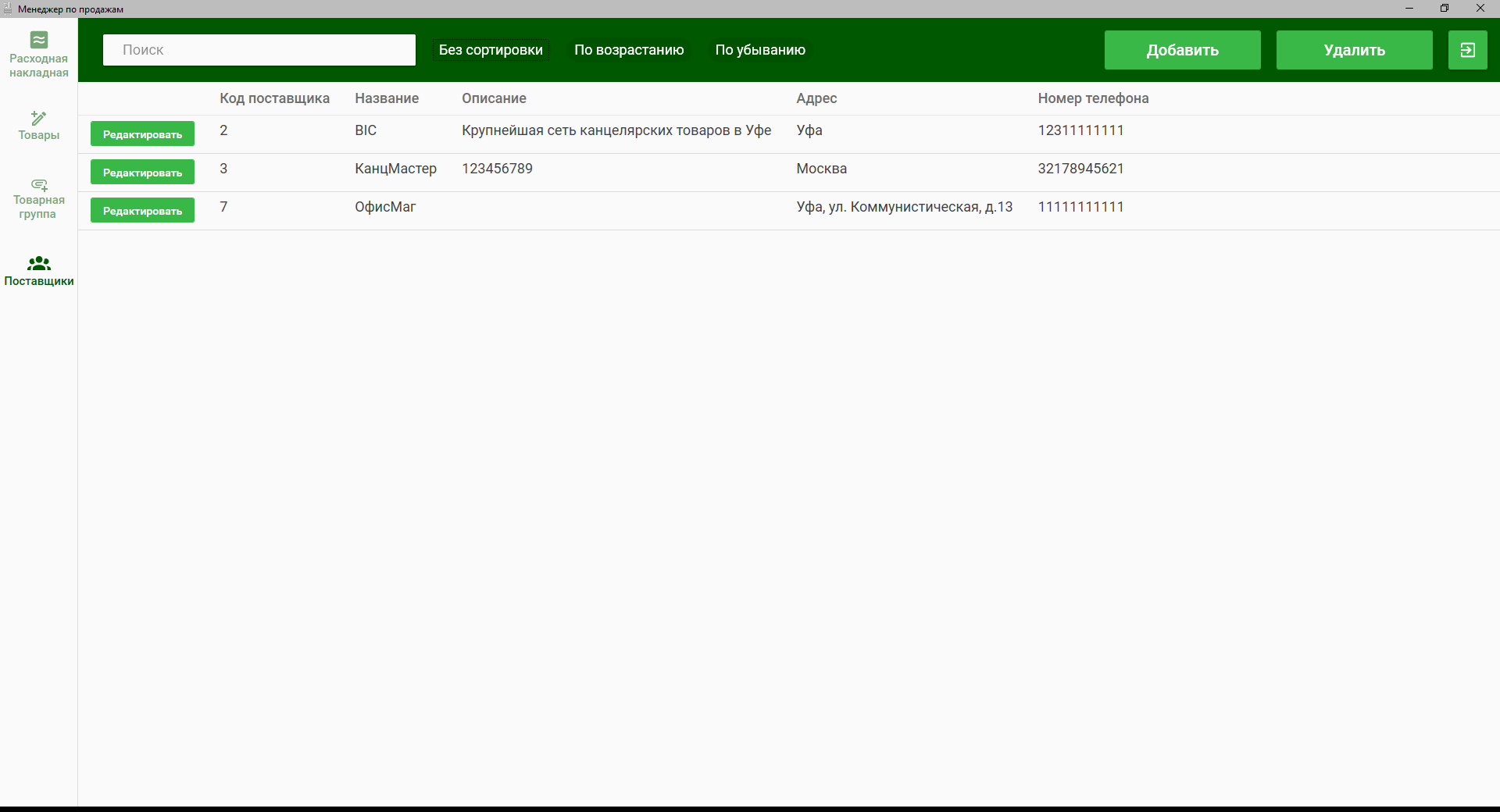


Рисунок 2.3.7 – Окно «Поставщики»

В данном окне, как и в прошлых мы можем добавлять, редактировать и удалять данные, выполнять поиск и сортировку.

На этом, функионал менеджера по продажам внутри программы заканчивается. Перейдем к администратору отдела продаж.

В отличие от менеджера по продажам, администратор отдела продаж формирует отчет по проданным товарам за период времени.

На рисунке 2.3.8 показано окно админитратора отдела продаж

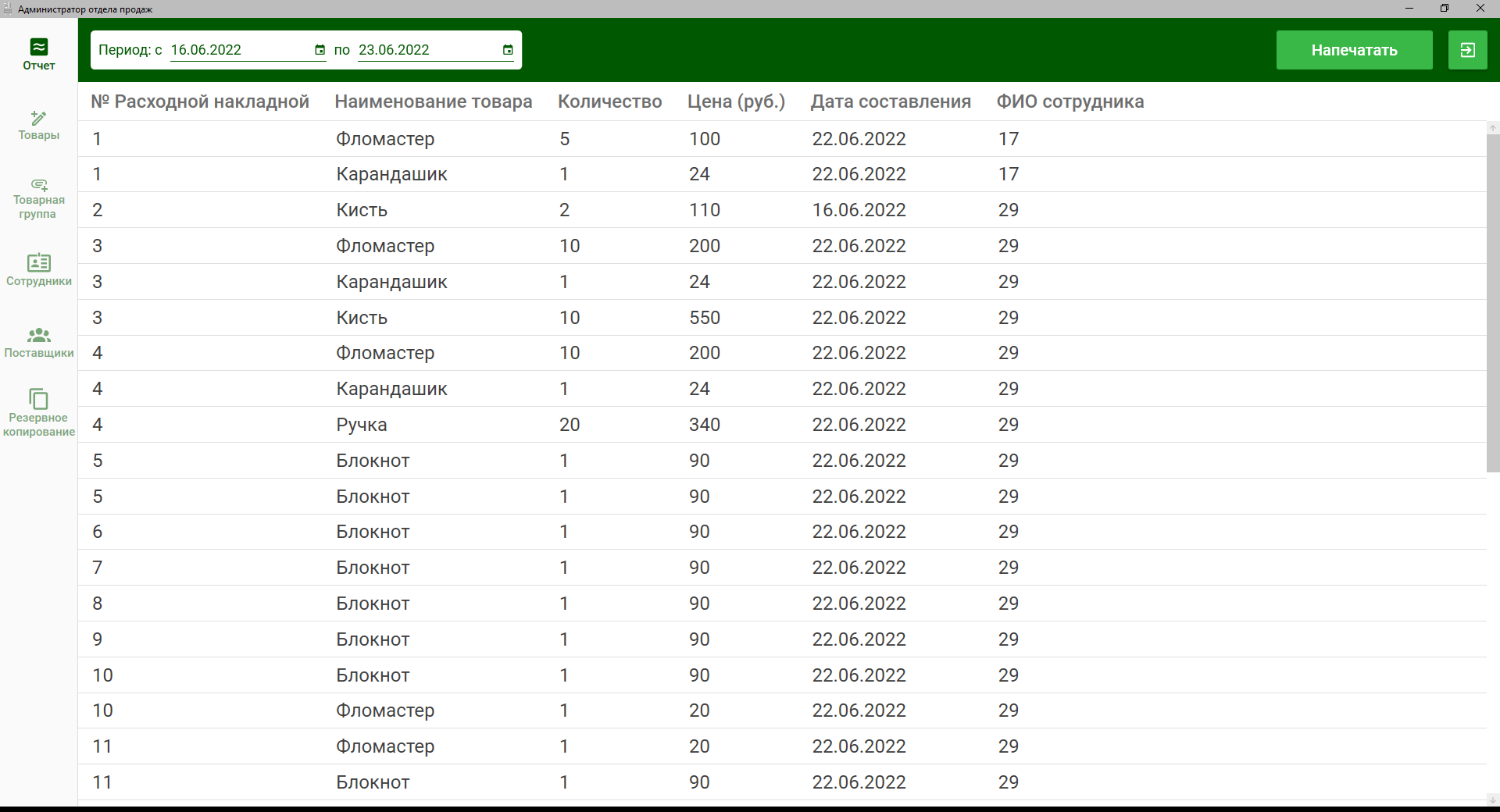


Рисунок 2.3.8 – Окно администратора отдела продаж

В верхнем левом углу данного окна мы можем выбрать две даты. Первая – минимальное число с которого будут выводиться, а второе – максимальное. Тем самым, у нас получается перриод времени. При нажатии на кнопку «Напечатать», формируется отчет по проданным товарам за промежуток времени.

Следующее окно – «Сотрудники»

На рисунке 2.3.9 показано окно «Сотрудники»

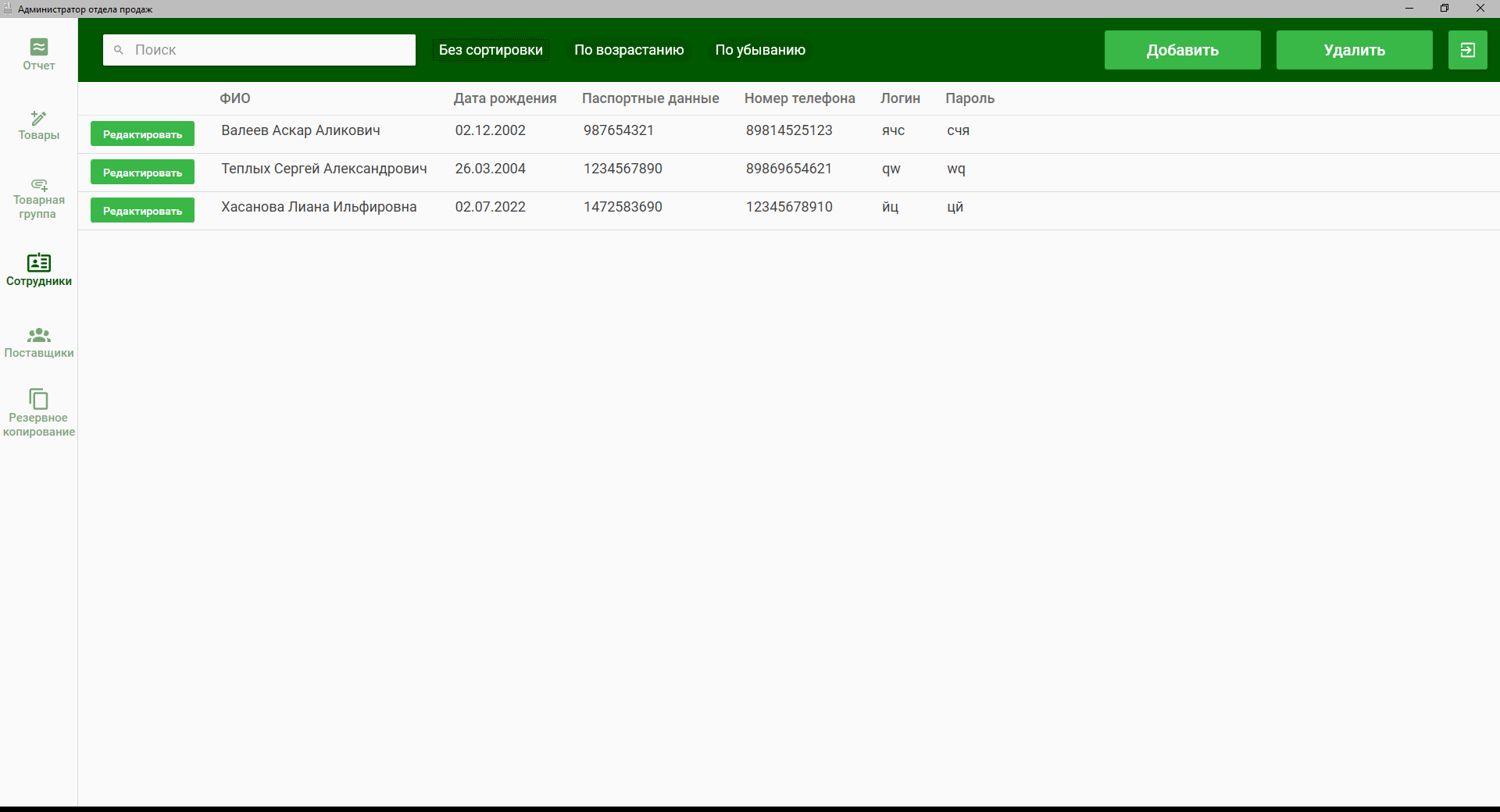


Рисунок 2.3.9 – Окно «Сотрудники»

В этом окне хранится список всех работников компании. Администратор отела продаж может добавлять новых сотрудников, удалять старых, и корректировать их данные. Также можно сортировать список сотрудников и осуществлять по нему поиск.

Работа администратора отдела продаж с товарами, товарными группами и поставщиками полностью аналогична менеджеру по продажам.

Администратор отдела продаж может создавать резервные копии данных.

На рисунке 2.3.10 показано окно «Резервное копирование»

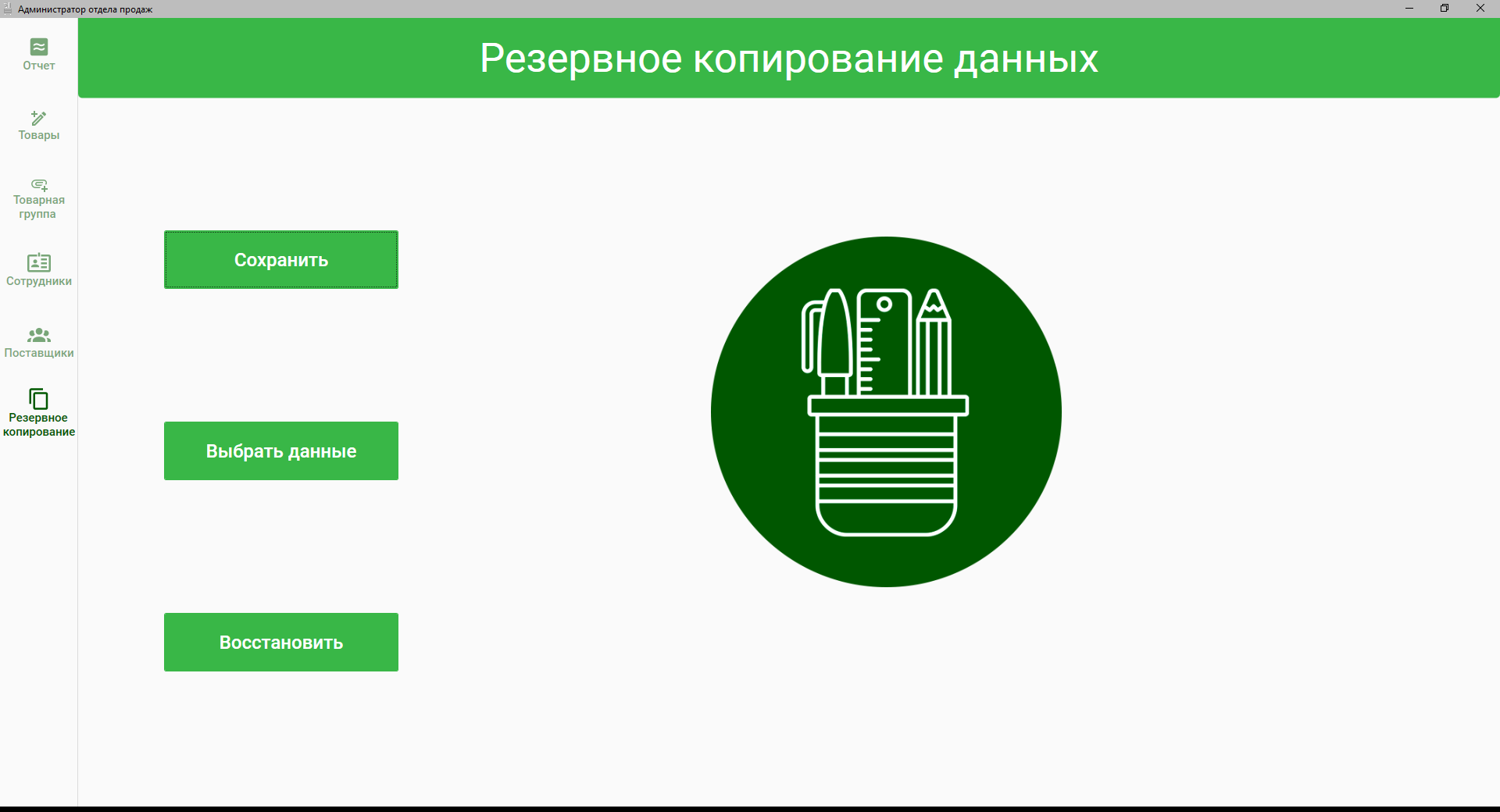


Рисунок 2.3.10 – Окно «Резервное копирование»

На рисунке 2.3.11 показан пример сообщения об ошибке.

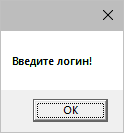


Рисунок 2.3.9 – Сообщение об ошибке

В таблице 2.3.1 показаны сообщения пользователю.

Таблица 2.3.1 – Сообщения пользователю

|  |  |  |
| --- | --- | --- |
| Сообщение | Причина | Что делать |
| Добавьте товары в корзину для составления расходной накладной | При формировании расходной накладной корзина была пуста | Добавить товары в корзину |
| Данные введены неверно! | Заполнены не все поля | Заполнить все поля |
| Выберите данные | Не выбраны данные из таблицы | Выбрать данные |
| Выберите товар для удаления | Не выбран товар для удаления | Выбрать товар |

2.4 Меры по обеспечению защиты информации

Администратор отдела продаж имеет право на создание отчетов за определенный период времени, изменение всей базы данных, добавление сотрудников, поставщиков и делать резервное копирование и восстановление.

Для защиты от непредвиденных ошибок и сбоев программы, была разработана функция резервного копирования и восстановления.

ЗАКЛЮЧЕНИЕ

В ходе написания курсового проекта была разработана структура приложения «Автоматизация учета канцелярских товаров на складе оптовой базы». При этом были изучены новые информационные технологии, были приобретены навыки в области построения и обработки данных.

Результатом работы стало создание программы Канцелярик.exe.

При выполнении курсового проекта были решены следующие задачи:

* проведен анализ предметной области;
* разработан алгоритм решения поставленной задачи;
* разработано программное обеспечение для формирования расходных накладных и отчетов по проданным товарам;
* проведено тестирование.

Цель данного курсового проекта упростить работу сотрудников склада по формированию документов и учета товаров путем создания автоматизированной информационной системы достигнута.

Программный продукт имеет дружественный пользователю интерфейс, для работы с программной не требуется дополнительное обучение.

Программа Канцелярик.exe прошла тестирование на данных контрольного примера.

СПИСОК СОКРАЩЕНИЙ

БД – база данных

ИС – информационная система

ПО – программное обеспечение

АИС – автоматизированная информационная система

ПО ПК – программное обеспечение персонального компьютера

Руб – рублей

ОЗУ – оперативное запоминающее устройство

СУБД – система управления базами данных

SQL – Structured Query Language (структурированный язык запросов)

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Федеральный закон "Об образовании в Российской Федерации" от 29.12.2012 N 273-ФЗ;
2. ГОСТ 2.105-95 ЕСКД. Общие требования к текстовым документам;
3. ГОСТ 2.316-68 ЕСКД. Правила нанесения на чертежах надписей технических требований и таблиц;
4. ГОСТ 19.701-90 ЕСКД. Схемы алгоритмов, программ, данных и систем;
5. ГОСТ 7.12-93 ЕСКД. Библиографическая запись. Сокращение слов на русском языке. Общие требования и правила;
6. ГОСТ 7.82-01 ЕСКД. Библиографическая запись. Библиографическое описание электронных ресурсов;
7. ГОСТ 7.1-03 ЕСКД. Библиографическая запись. Библиографическое описание;
8. ГОСТ 2.104-68 ЕСКД. Основные надписи;
9. ГОСТ 7.32-2001 Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления;
10. ГОСТ 19.781-90. Единая система программной документации. Программное обеспечение системы обработки информации. Термины и определения. [Текст]. – Введ. 1992–01–01 -М.: Изд-во стандартов, 1992.- 27 с;
11. ГОСТ 24.701-86. Единая система стандартов автоматизированных систем управления. Надежность автоматизированных систем управления. Основные положения. [Текст]. – Введ. 1987–01–07 -М.: Изд-во стандартов, 1990.- 37 с;
12. ГОСТ 34.003-90. Информационная технология. Комплекс стандартов на автоматизированные системы. Термины и определения. [Текст]. – Введ. 1992–01–01 -М.: Изд-во стандартов, 1990.- 158 с;
13. Астапчук, В. А. Корпоративные информационные системы: требования при проектировании: учебное пособие для вузов [Электронный ресурс]/ В. А. Астапчук, П. В. Терещенко. — 2-е изд., испр. и доп. — Москва: Издательство Юрайт, 2022. — 113 с.— Режим доступа: <https://urait.ru/bcode/492141>, по паролю;
14. Илюшечкин, В. М. Основы использования и проектирования баз данных: учебник для среднего профессионального образования [Электронный ресурс]/ В. М. Илюшечкин. — испр. и доп. — Москва: Издательство Юрайт, 2022. — 213 с.— Режим доступа: <https://urait.ru/bcode/491755> по паролю;
15. Кругликов, В. Н. Интерактивные образовательные технологии: учебник и практикум для вузов [Электронный ресурс]/ В. Н. Кругликов, М. В. Оленникова. — 3-е изд., испр. и доп. — Москва: Издательство Юрайт, 2022. — 355 с. — Режим доступа: <https://urait.ru/bcode/488475>, по паролю;
16. Нестеров, С. А. Базы данных: учебник и практикум для вузов [Электронный ресурс]/ С. А. Нестеров. — Москва: Издательство Юрайт, 2022. — 230 с. — Режим доступа: <https://urait.ru/bcode/489693>, по паролю;
17. Парфенов, Ю. П. Постреляционные хранилища данных: учебное пособие для вузов [Электронный ресурс]/ Ю. П. Парфенов; под научной редакцией Н. В. Папуловской. — Москва: 121 с. — Режим доступа: <https://urait.ru/bcode/492609>, по паролю;
18. Проектирование информационных систем: учебник и практикум для вузов [Электронный ресурс]/ под общей редакцией Д. В. Чистова. — Москва: Издательство Юрайт, 2022. — 258 с. — Режим доступа: <https://urait.ru/bcode/489307>, по паролю.
19. Стасышин, В. М. Базы данных: технологии доступа: учебное пособие для вузов [Электронный ресурс]/ В. М. Стасышин, Т. Л. Стасышина. — 2-е изд., испр. и доп. — Москва: Издательство Юрайт, 2022. — 164 с. — Режим доступа: https://urait.ru/bcode/492177, по паролю;
20. Толстобров, А. П. Управление данными: учебное пособие для вузов [Электронный ресурс]/ А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2022. — 272 с. — Режим доступа: <https://urait.ru/bcode/496748>, по паролю.