Центросоюз Российской Федерации

Нижегородский областной союз потребительских обществ

«Нижегородский экономико-технологический колледж»

**КУРСОВАЯ РАБОТА**

по ПМ.05 Проектирование и разработка информационных систем

Тема: «Разработка информационной системы

торговой компании»

Выполнил:

студент группы 42П

специальности 09.02.07

Информационные системы и программирование

Митьков Е.А

Руководитель:

Кисарова М.В., преподаватель НЭТК

Дата сдачи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

г. Арзамас, 2024

СОДЕРЖАНИЕ

ВВЕДЕНИЕ 3

1. АНАЛИЗ И ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ «ТОРГОВОЙ КОМПАНИИ» 4

* 1. Описание предметной области 4
  2. Определение функциональных требований 6
  3. Моделирование структуры данных 7
  4. Выбор методов и средств разработки 8

2. РЕАЛИЗАЦИЯ ПРОЕКТА ИНФОРМАЦИОННОЙ СИСТЕМЫ «НАЗВАНИЕ СИСТЕМЫ» 10

1. Подключение к базе данных 10
2. Роли и авторизация 12
3. Пользовательский интерфейс оператора системы 13
4. Пользовательский интерфейс администратора системы 14
5. Тестирование 16

ЗАКЛЮЧЕНИЕ 17

ПРИЛОЖЕНИЯ

ВВЕДЕНИЕ

Современный рынок розничной торговли предъявляет высокие требования к эффективности управления бизнесом. Оптимизация внутренних процессов, учет товаров, взаимодействие с поставщиками и мониторинг финансовой деятельности становятся ключевыми факторами успеха компаний. Информационные системы играют решающую роль в автоматизации этих процессов, повышая конкурентоспособность предприятий.

Актуальность данной работы обусловлена необходимостью внедрения современной информационной системы, способной обеспечить комплексный подход к решению указанных задач. Целью исследования является проектирование и разработка информационной системы для торговой компании, направленной на повышение эффективности управленческих решений, снижение рисков и увеличение доходности бизнеса.

Основными задачами исследования являются:

- анализ предметной области и выявление функциональных требований;

- проектирование структуры данных и выбор методов и средств разработки;

- реализация проекта информационной системы;

- тестирование и оценка качества созданной системы.

Объектом исследования выступает торговая компания, предметом — процессы автоматизации и оптимизации ее деятельности посредством разработки информационной системы.

Методами исследования служат анализ, синтез, моделирование и эксперимент. Источниками исследования послужили учебные пособия, стандарты и специализированные публикации по тематике проектирования и разработки информационных систем.

Практическая значимость работы заключается в создании инструмента, позволяющего повысить уровень конкурентоспособности и эффективности торговой компании благодаря внедрению инновационных решений в сфере информационных технологий.

1. АНАЛИЗ И ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ «ТОРГОВОЙ КОМПАНИИ»

1.1. Описание предметной области

Предметная область связана с деятельностью торговой компании, включающей разнообразные процессы организации торговли, взаимодействия с поставщиками, управлением товарами и клиентами, а также контролем финансовых показателей. Основная цель информационной системы заключается в автоматизации и оптимизации процессов принятия решений, мониторинга ситуации на рынке и повышения эффективности функционирования компании.

Основные субъекты предметной области:

1. Торговые точки: Включают разные форматы торговли, такие как супермаркеты, магазины и киоски. Каждая точка характеризуется своими особенностями, такими как размеры помещений, условия аренды, наличие секций и отделов, особенности обслуживания клиентов и подходы к учету продаж.

2. Сотрудники: Ключевыми фигурами являются продавцы, чья деятельность направлена на реализацию товаров клиентам. Учет продаж осуществляется персонально, что важно для оценки индивидуальной производительности работников.

3. Клиенты: Для улучшения качества обслуживания и понимания покупательского поведения система собирает персональные данные о клиентах, особенно в больших торговых площадках вроде супермаркетов и специализированных магазинов.

4. Номенклатура товаров: Вся продукция классифицируется согласно определённым критериям, таким как категории, производители, цены и объемы закупок. Управление ассортиментом является важной частью стратегии развития компании.

5. Поставщики: Организация взаимодействует с множеством поставщиков, выбирая лучшие варианты для удовлетворения потребностей бизнеса. Регулярный мониторинг предложений поставщиков и обновление базы данных о товарах обеспечивают адаптацию к изменениям на рынке.

6. Задачи и запросы: Предусмотрены многочисленные механизмы обработки запросов, позволяющие анализировать эффективность деятельности предприятия, оценивать производительность персонала, рассчитывать прибыльность операций и контролировать качество предоставляемых услуг.

Главные блоки системы включают:

1. Модуль учёта товаров и продаж. Здесь реализуются процедуры регистрации поступления товаров, распределения их по магазинам, фиксации совершаемых сделок купли-продажи и формирования соответствующих документов.

2. Модуль аналитики и отчётности. Данный модуль предназначен для накопления статистики о работе сотрудников, оценке популярности отдельных видов товаров среди потребителей, расчете рентабельности отдельных подразделений компании и выявлению проблемных зон, требующих внимания руководства.

3. Модуль взаимодействия с поставщиками. Через этот модуль происходят операции выбора оптимального поставщика для каждой отдельной закупки, оформления заказов и отслеживания доставки товаров.

4. Модуль финансового планирования и бюджетирования. Модуль поддерживает расчёты затрат на оплату труда, коммунальных платежей, налогов и иных издержек, позволяя эффективно планировать денежные потоки и минимизировать риски дефицита бюджета.

5. Модуль клиентской базы и CRM. Реализует работу с информацией о клиентах, создание персональных профилей постоянных покупателей, проведение маркетинговых мероприятий и оценку удовлетворенности покупателей качеством обслуживания.

1.2. Определение функциональных требований

Управление Торговыми Точками

1. Регистрация и ведение данных о торговых точках различного типа (супермаркеты, магазины, киоски).

- Хранение информации о характеристиках торговых точек (размер, аренда, коммунальные расходы, количество прилавков).

2. Возможность изменять статус принадлежности торговых площадей (собственность / аренда).

Управление Поставщиками

1. Поддержка каталога поставщиков с возможностью добавления, удаления и обновления записей.

- Фиксация истории взаимодействия с каждым поставщиком (товары, условия поставок, контракты).

- Автоматическое пополнение справочника товаров новыми позициями при появлении нового товара.

Формирование Заказов

1. Система должна поддерживать механизм формирования заказов поставщикам на основе заявок торговых точек.

- Возможность сравнения цен и выбора оптимального поставщика.

- Учет возможности приобретения одних и тех же товаров от разных поставщиков по разным ценам.

Логистика Товаров

2. Распределение поступивших товаров по торговым точкам.

- Генерация распределения товаров с возможностью проверки текущего состояния запасов.

Учёт Продаж

1. Сбор и хранение информации о продажах каждого товара с детализацией:

- Название товара, количество проданного, цена реализации.

- Для супермаркетов и магазинов дополнительно фиксируется имя покупателя и его характеристика.

2. Обеспечение гибкости в изменении цен на товары внутри торговых точек.

Работа с Персоналом

3. Создание и поддержка базы данных сотрудников (продавцов), включающей персональные данные и показатели производительности.

- Предоставление инструментов для расчета зарплаты работников в зависимости от объемов продаж.

Аналитика и Отчётность

1. Поддержка множества видов отчетов, среди которых:

- Поиск и получение списка поставщиков определённого товара за нужный период.

- Получение списков покупателей определенных товаров с учётом объёма покупок.

- Информация о наличии товаров в торговых точках.

- Данные о выручке и зарплатах сотрудников.

- Статистика продаж отдельных товаров по отдельным торговым точкам.

- Расчёт показателей рентабельности и прибыли торговых объектов.

Интеграция и Совместимость

2. Поддержка интеграции с существующими системами бухгалтерского учёта и ERP-решениями компании.

1.3. Моделирование структуры данных

Целью моделирования структуры данных является создание подробной модели, отражающей все необходимые объекты и отношения, существующие в рамках информационной системы. Эта модель должна обеспечивать эффективное хранение и обработку данных, необходимых для функционирования системы.

Моделирование структуры данных включает следующие этапы:

1. Идентификация сущностей: Определение ключевых объектов, которые будут храниться в базе данных. Например, торговые точки, сотрудники, товары, поставщики и покупатели.

2. Определение связей: Установление взаимосвязей между сущностями. Например, связь между поставщиками и товарами, сотрудниками и торговыми точками.

3. Атрибуты сущностей: Описание свойств каждой сущности. Например, название магазина, адрес, телефон сотрудника, цена товара.

4. Нормализация данных: Приведение данных к нормальной форме для устранения избыточности и повышения эффективности хранения и обработки.

5. Создание диаграммы ER-модели: Графическое представление структуры данных, включающее сущности, атрибуты и связи.

По результатам моделирования создается концептуальная модель данных, которая служит основой для дальнейшей разработки физической схемы базы данных. Эта модель помогает разработчикам понять, как данные будут организованы и связаны друг с другом, обеспечивая прозрачность и структурированность проекта.

1.4. Выбор методов и средств разработки

Выбор методов и средств разработки направлен на достижение следующих целей:

1. Обеспечение соответствия требованиям бизнеса.

2. Повышение качества и надежности программного продукта.

3. Сокращение сроков и стоимости разработки.

4. Поддержка дальнейшего развития и масштабируемость системы.

При выборе методов и средств учитываются следующие критерии:

1. Соответствие функциональным требованиям: выбранные методы и средства должны поддерживать реализацию всех необходимых функций системы.

2. Производительность и масштабируемость: инструменты должны обеспечивать высокую производительность и возможность расширения системы в будущем.

3. Совместимость с существующими системами: важно учитывать интеграцию с уже используемым оборудованием и ПО.

4. Стоимость владения: выбор инструментов должен учитывать затраты на приобретение, внедрение и поддержку.

5. Поддержка и развитие: наличие документации, сообщества разработчиков и доступность обновлений важны для долгосрочной поддержки системы.

Среди методов разработки выделяются:

- Agile-методологии: гибкий подход, позволяющий быстро реагировать на изменения требований и повышать качество продукта.

- Waterfall: последовательный подход, подходящий для проектов с чётко определёнными требованиями.

- Scrum: методология Agile, обеспечивающая регулярную обратную связь и итерационное улучшение продукта.

Средствами разработки являются инструменты и технологии, используемые для реализации проекта. К ним относятся:

- Языки программирования: C#

- СУБД: MySQL

- IDE: Visual Studio

- Система контроля версий: Git

Итоговый выбор методов и средств зависит от специфики проекта, команды разработчиков и бюджета. Важно провести тщательное исследование рынка и выбрать оптимальные решения, соответствующие поставленным задачам.

2. РЕАЛИЗАЦИЯ ПРОЕКТА ИНФОРМАЦИОННОЙ СИСТЕМЫ «НАЗВАНИЕ СИСТЕМЫ»

2.1. Подключение к базе данных

Чтобы подключить приложение на C# к базе данных SQL Server, выполните следующие шаги:

1. Установка пакета NuGet: Добавьте пакет System.Data.SqlClient в ваш проект через менеджер пакетов NuGet.

2. Создание строки подключения: Определите строку подключения к вашей базе данных. Она обычно выглядит следующим образом:

string connectionString = @"Data Source=(local)\\SQLEXPRESS;Initial Catalog=MyDatabase;Integrated Security=True";

Где:

- Data Source: сервер баз данных.

- Initial Catalog: имя базы данных.

- Integrated Security: используется встроенная аутентификация Windows.

3. Открытие соединения: Создайте объект SqlConnection и откройте соединение с базой данных.

using (SqlConnection conn = new SqlConnection(connectionString))

{

conn.Open();

}

4. Выполнение запросов: Используйте класс SqlCommand для выполнения SQL-запросов.

using (SqlCommand cmd = new SqlCommand("SELECT \* FROM Products", conn))

{

using (SqlDataReader reader = cmd.ExecuteReader())

{

while (reader.Read())

{

Console.WriteLine(reader["ProductName"]); // вывод названия продукта

}

}

}

5. Обработка исключений: Всегда обрабатывайте возможные исключения, возникающие при работе с базой данных.

try

{

// код работы с базой данных

}

catch (SqlException ex)

{

Console.WriteLine(ex.Message);

}

finally

{

if (conn != null && conn.State == ConnectionState.Open)

{

conn.Close(); // закрываем соединение

}

}

Подключение к базе данных SQL в C# осуществляется путем установки необходимого пакета, создания строки подключения, открытия соединения и выполнения SQL-запросов. Правильная обработка ошибок и закрытие соединений обеспечивают надежность и стабильность приложения.

2.2. Роли и авторизация

Роль представляет собой группу прав и привилегий, назначаемых пользователям. В контексте информационной системы торговой компании роли могут включать:

- Администратор: Полный доступ ко всей системе, включая настройку пользователей и ролей.

- Менеджер: Доступ к управлению заказами, поставками и складскими запасами.

- Кассир: Ограниченный доступ, предназначенный для обработки транзакций и продаж.

- Поставщик: Возможность просмотра информации о поставляемых товарах и статусах заказов.

Авторизация — процесс проверки полномочий пользователя перед предоставлением доступа к ресурсам системы. Основными методами авторизации являются:

1. Windows Authentication: Использование встроенных учётных записей операционной системы.

2. Forms Authentication: Самостоятельная реализация механизма авторизации через формы входа.

3. Role-Based Access Control (RBAC): Предоставление доступа на основе назначенных ролей

Для реализации авторизации в приложении на C# можно воспользоваться стандартными средствами ASP.NET Identity. Вот основные шаги:

1. Настройка конфигурации: Включите систему авторизации в файле Startup.cs вашего веб-приложения.

services.AddIdentity<ApplicationUser, ApplicationRole>()

.AddEntityFrameworkStores<ApplicationDbContext>();

2. Регистрация пользователей: Реализуйте регистрацию новых пользователей с назначением соответствующих ролей.

var userManager = context.GetService<UserManager<ApplicationUser>>();

await userManager.CreateAsync(new ApplicationUser { UserName = "admin@example.com", Email = "admin@example.com" }, "password");

await userManager.AddToRoleAsync(user, "Administrator");

3. Проверка прав доступа: Используем атрибут [Authorize(Roles = "Admin")] для ограничения доступа к контроллерам или действиям.

[Authorize(Roles = "Admin")]

public IActionResult ManageUsers()

{

return View();

}

Правильное использование ролей и механизмов авторизации обеспечивает защиту данных и ограничение доступа к функционалу системы. Следование принципам RBAC и использование стандартных средств ASP.NET позволяют создать надёжную и удобную систему управления пользователями и правами доступа.

2.3. Пользовательский интерфейс оператора системы

При создании пользовательского интерфейса следует придерживаться следующих принципов:

1. Простота и ясность: Минималистичный дизайн, понятные обозначения элементов управления.

2. Логичность структуры: Четкая организация меню и вкладок, интуитивное расположение кнопок и полей ввода.

3. Доступность информации: Быстрый доступ к часто используемым функциям и данным.

4. Отзывчивость и скорость: Минимальное время загрузки страниц и выполнение операций.

Пользовательский интерфейс оператора может включать следующие элементы:

- Меню навигации: Панель с основными функциями (например, заказы, склад, отчёты).

- Форма поиска: Быстрое получение необходимой информации.

- Таблицы данных: Представление списков товаров, клиентов, заказов.

- Диаграммы и графики: Наглядное отображение статистики и динамики продаж.

- Формы редактирования: Изменение данных о товарах, клиентах, поставках.

5. Адекватная визуализация данных: Удобные таблицы, графики и диаграммы для представления информации.

Вот несколько примеров экранов, которые могут быть полезны оператору:

- Экран списка заказов: Отображение текущих заказов с возможностью фильтрации и сортировки.

- Карточка товара: Просмотр детальной информации о товаре, включая цену, остатки на складе и историю продаж.

- Статистика продаж: Анализ продаж за определённый период с графическим представлением результатов.

- Форма создания нового заказа: Простое заполнение данных о заказе и отправка поставщику.

Эффективный пользовательский интерфейс оператора способствует повышению производительности труда и улучшению качества обслуживания клиентов. Грамотное проектирование и применение современных технологий позволяют создать удобный и эффективный инструмент для работы операторов торговой компании.

2.4. Пользовательский интерфейс администратора системы

Интерфейс администратора должен обладать следующими характеристиками:

1. Полный контроль: Возможность изменять настройки системы, добавлять и удалять пользователей, настраивать права доступа.

2. Мониторинг: Отображение текущего состояния системы, статистика использования ресурсов, уведомления о проблемах.

3. Безопасность: Наличие строгих мер защиты, таких как двухфакторная аутентификация и журналы аудита действий администраторов.

4. Удобство использования: Чёткое разделение функций, удобные фильтры и сортировка данных.

Интерфейс администратора может включать следующие разделы:

- Панель мониторинга: Общая информация о состоянии системы, ключевые показатели (KPI).

- Управление пользователями: Создание, удаление и изменение профилей пользователей, назначение ролей и прав доступа.

- Конфигурация системы: Настройки базы данных, сервера приложений, интеграции с внешними сервисами.

- Отчёты и аналитика: Генерация отчётов по активности пользователей, состоянию запасов, истории изменений.

- Защита и аудит: Просмотр журналов событий, настройка уведомлений о подозрительной активности.

Особое внимание уделяется безопасности интерфейса администратора:

- Аутентификация: Двухфакторная аутентификация, проверка IP-адресов, лимит попыток входа.

- Шифрование: Шифрование паролей и чувствительных данных.

- Контроль доступа: Ограничение доступа к административному интерфейсу только доверенным лицам.

Интерфейс администратора играет важную роль в обеспечении стабильной и безопасной работы информационной системы. Его грамотное проектирование и реализация способствуют эффективному управлению ресурсами и минимизации рисков, связанных с несанкционированным доступом и ошибками администрирования.

2.5. Тестирование

Существуют разные виды тестирования, каждый из которых решает свою задачу:

1. Юнит-тестирование: Проверка отдельных компонентов системы (методов, классов).

2. Интеграционное тестирование: Оценка взаимодействия модулей и подсистем.

3. Функциональное тестирование: Проверка правильности реализации функциональных требований.

4. Нагрузочное тестирование: Изучение поведения системы при высоких нагрузках.

5. Тестирование безопасности: Проверка устойчивости системы к атакам и попыткам взлома.

Тестирование проходит в несколько этапов:

1. Планирование: Определение целей и стратегии тестирования, разработка тест-плана.

2. Подготовка: Разработка тестов, подготовка тестовых данных и среды тестирования.

3. Исполнение: Проведение тестов, регистрация результатов.

4. Анализ: Анализ выявленных дефектов, составление отчета о результатах тестирования.

5. Корректировка: Исправление найденных ошибок, повторное тестирование исправленного функционала.

Правильно организованное тестирование существенно повышает качество информационной системы, снижает риски сбоев и повышает удовлетворенность пользователей. Применение автоматизированных инструментов и соблюдение стандартов тестирования делают этот процесс эффективным и предсказуемым.

ЗАКЛЮЧЕНИЕ

В результате проведенного исследования была разработана информационная система для торговой компании, нацеленная на автоматизацию и оптимизацию основных бизнес-процессов. Разработанная система обеспечивает комплексное решение задач учета товаров, управления персоналом, взаимодействия с поставщиками, анализа данных и подготовки отчетности.

Полученные результаты подтверждают актуальность выбранной темы и демонстрируют возможность значительного повышения эффективности деятельности компании за счет внедрения современных информационных технологий. В ходе выполнения работы были успешно выполнены следующие задачи:

- проведен глубокий анализ предметной области и определены функциональные требования к системе;

- спроектирована структура данных и выбраны эффективные методы и средства разработки;

- создана полноценная информационная система, включающая модули учета товаров, анализа и отчетности, взаимодействия с поставщиками и финансовый блок;

- проведены тесты, подтвердившие работоспособность и функциональность системы.

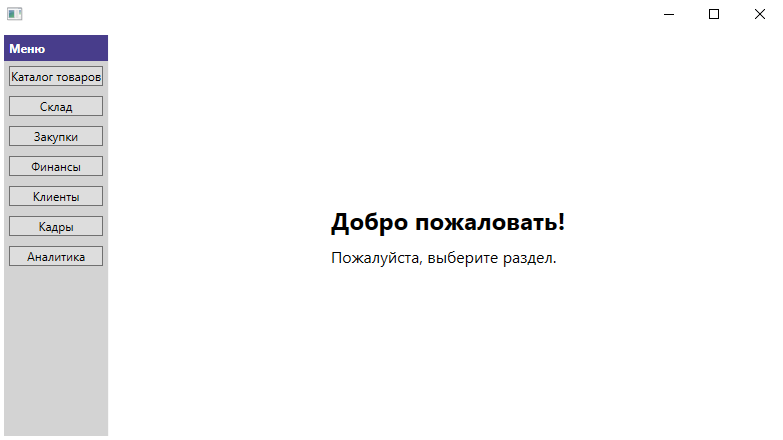
Таким образом, разработанная информационная система позволит торговой компании значительно повысить качество обслуживания клиентов, снизить операционные расходы и увеличить прибыль. Результаты работы имеют большую практическую ценность и могут служить основой для дальнейших исследований и разработок в области информационных технологий для бизнеса.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

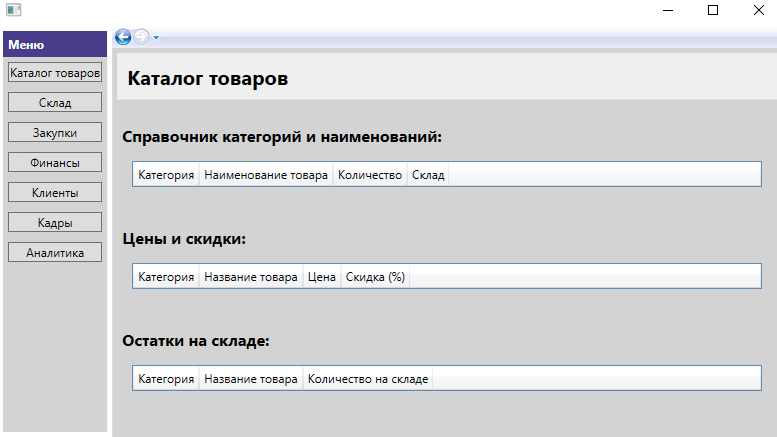
1. Абрамов, Г. В. Проектирование и разработка информационных систем : учебное пособие для СПО / Г. В. Абрамов, И. Е. Медведкова, Л. А. Коробова. — Саратов : Профобразование, 2020. — 169 c. — ISBN 978-5-4488-0730-5. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROFобразование : [сайт]. — URL: https://profspo.ru/books/88888. — Режим доступа: для авторизир. пользователей
2. Вичугова А.А. Инструментальные средства разработки компьютерных систем и комплексов [Электронный ресурс] : учебное пособие для СПО / А.А. Вичугова. — Электрон. текстовые данные. — Саратов: Профобразование, 2019. — 135 c. — 978-5-4488-0015-3. — Режим доступа: http://www.iprbookshop.ru/66387.html
3. Котляров, В. П. Основы тестирования программного обеспечения : учебное пособие для СПО / В. П. Котляров. — Саратов : Профобразование, 2019. — 335 c. — ISBN 978-5-4488-0364-2. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROFобразование : [сайт]. — URL: https://profspo.ru/books/86202 . — Режим доступа: для авторизир. пользователей
4. Профессиональное управление проектом. — : учебное пособие / К. Хелдман, пер. с англ. — 6-е издание. — Москва : Лаборатория знаний, 2019. — 731 с. — ISBN 978-5-9963-2503-0. Номер в ЭБС: 924003- https://www.book.ru/book/924003
5. Спицина, И. А. Разработка информационных систем. Пользовательский интерфейс : учебное пособие для СПО / И. А. Спицина, К. А. Аксёнов ; под редакцией Л. Г. Доросинского. — 2-е изд. — Саратов, Екатеринбург : Профобразование, Уральский федеральный университет, 2020. — 98 c. — ISBN 978-5-4488-0768-8, 978-5-7996-2872-7. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROFобразование : [сайт]. — URL: https://profspo.ru/books/92370. — Режим доступа: для авторизир. пользователей

**15** источников

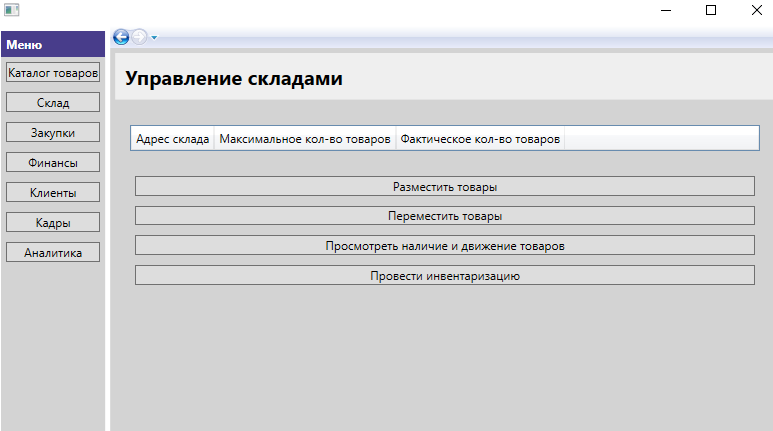
ПРИЛОЖЕНИЕ А

****

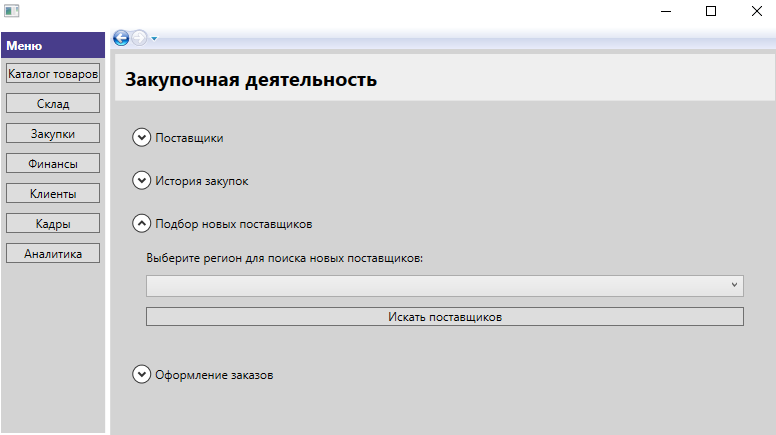
1. Главное окно



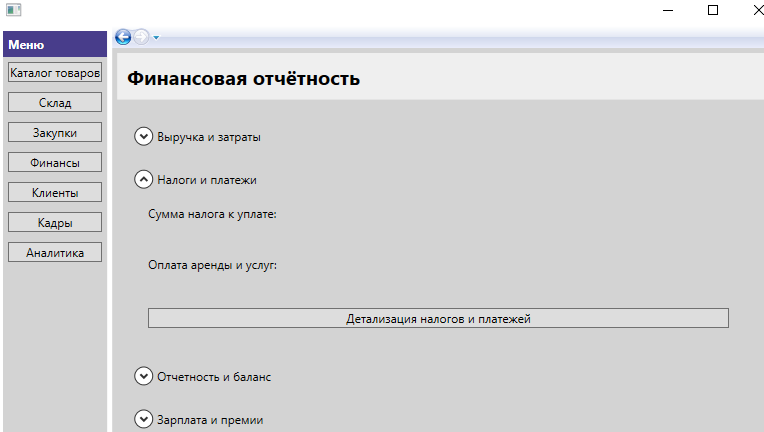
1. Каталог товаров



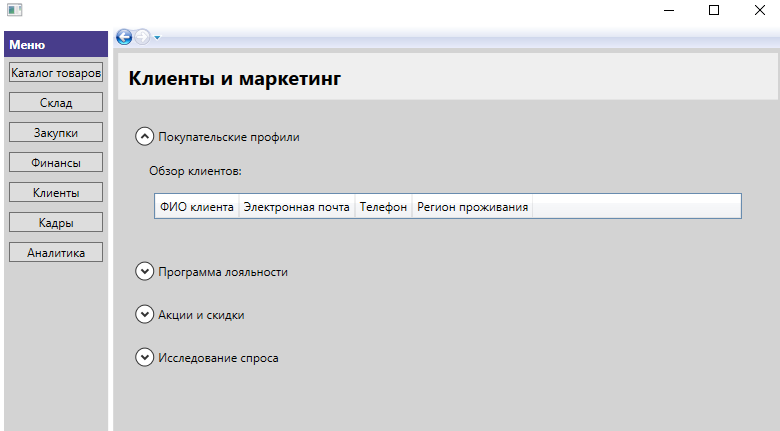
1. Раздел управление складами



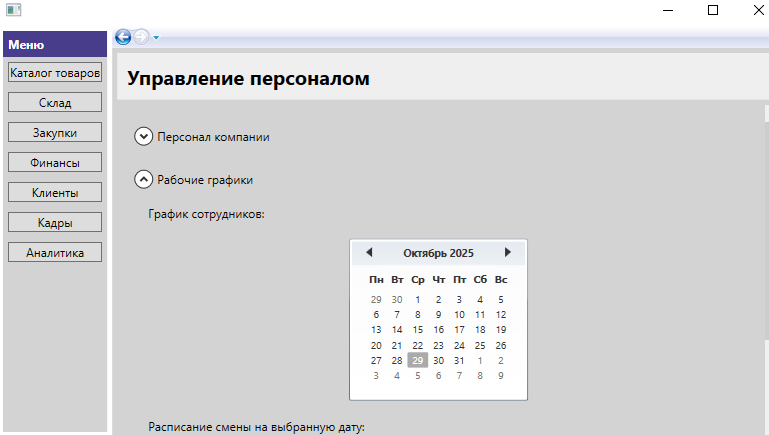
1. Раздел закупок



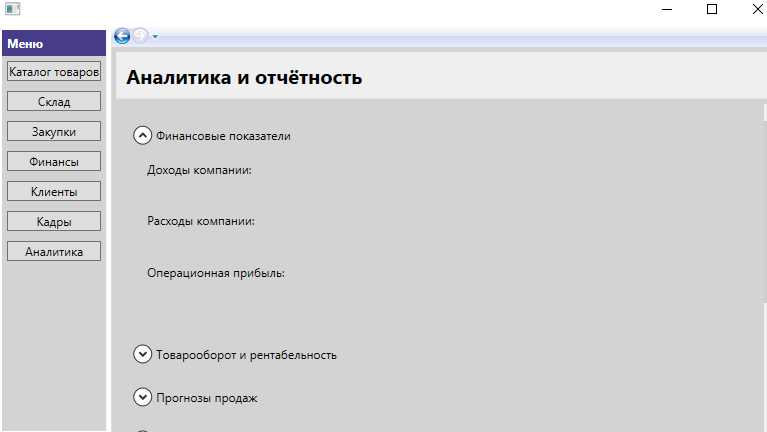
1. Раздел финансы



1. Раздел клиенты



1. Раздел кадры



1. Раздел аналитики