Министерство образования Тульской области

Государственное профессиональное образовательное учреждение

Тульской области

«Донской политехнический колледж»

Разработка Информационной системы учета продукции для магазина «Красное&белое»

Курсовая работа МДК 02.01

«Технология разработки программного обеспечения»

|  |  |
| --- | --- |
| Студента группы С-20-1 | Д.В Чернышов |
| Руководитель | С.М. Гвоздев |
| Проверил | \_\_\_\_\_\_\_\_\_\_\_\_\_\_  (дата) |
| Оценка | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Донской, 2022

Содержание

|  |  |  |
| --- | --- | --- |
|  | [Введение](#_Введение) ………………………………………………………………. | 3 |
| 1 | [Техническое задание](#_Техническое_задание) …………………………………………………. | 5 |
| 1.1 | [Назначение работы](#_1.1_Назначение_работы)……………………………………………………. | 5 |
| 1.2 | [Требования к функциональным характеристикам](#_1.2_Требования_к)………………….. | 5 |
| 1.3 | [Требования к надёжности и безопасности](#_1.3_Требования_к)…………………………... | 6 |
| 1.4 | [Требования к составу и параметрам технических средств](#_1.4_Требования_к)………… | 6 |
| 1.5 | [Требования к информационной и программной совместимости](#_1.5_Требования_к)….. | 6 |
| 1.6 | Порядок контроля и приёмки………………………………………… | 7 |
| 2 | [Разработка технического проекта](#_2_Разработка_технического)……………………………………. | 8 |
| 2.1 | [Анализ требований и спецификаций](#_2.1_Анализ_требований)………………………………… | 8 |
| 2.2 | [Этап эскизного проектирования программного](#_2.2_Этап_эскизного) обеспечения……... | 10 |
| 2.3 | Проектирование внутренней структуры…………………………….. | 13 |
| 3 | Реализация программного обеспечения…………………….……….. | 15 |
| 3.1 | Обоснование выбора средств разработки…………………………… | 15 |
| 3.2 | Разработка программного обеспечения……………………………... | 16 |
| 3.3 | Технико-экономические показатели………………………………… | 20 |
|  | Заключение……………………………………………………………. | 23 |
|  | Список используемых источников…………………………………... | 24 |
|  | Приложение А………………………………………………………… | 26 |
|  | Приложение Б…………………………………………………………. | 30 |
|  | Приложение В…………………………………………………………. |  |

# ВВЕДЕНИЕ

Сегодня информацию рассматривают как один из основных ресурсов развития общества, а информационные системы и технологии как средство повышения производительности и эффективности работы людей. Создание современных электронных вычислительных машин позволило автоматизировать обработку данных во многих сферах человеческой деятельности, в том числе торговле. Без современных систем обработки данных трудно представить сегодня передовые производственные технологии, управление экономикой на всех ее уровнях, научные исследования, образование, издательское дело, функционирование средств массовой информации, проведение крупных спортивных состязаний. Значительно расширило сферу применения систем обработки данных появление персональных компьютеров. Одним из наиболее распространенных классов систем обработки данных являются информационные системы.

Внедрение ИС - очень сложный и дорогостоящий проект, осуществление которого сопряжено с целым рядом разноплановых, трудно детерминируемых рисков, которые приводят к тому, что проект либо вообще терпит фиаско, либо не достигает поставленных целей. Кроме того, анализ эффективности проектов автоматизации методологически сложен и вызывает многие трудности с её эксплуатацией

Объектом исследования является информационная система учета продукции в магазине «Красное & Белое».

Предметом исследования является процесс разработки и внедрения автоматизированной информационной системы в магазин «Красное & Белое».

Целью курсовой работы является создание программного продукта, необходимого для учета доставляемого сырья (продукции) в магазин «Красное & Белое».

Для достижения поставленной цели необходимо решить следующие задачи:

* проанализировать предметную область;
* составить техническое задание;
* выбрать инструментальные средства разработки;
* выполнить программирование ПП;
* рассчитать технико-экономические показатели.

Практической значимостью является то, что данный программный продукт сокращает затраты времени на учет продукции. Так же является значительным облегчением для работников магазина

# 1 Техническое задание

# 1.1 Назначение работы

Программный продукт «Красное & Белое» служит для подсчета и учета продукции на складе в магазине.

Преимуществом программного продукта «Красное & Белое» является:

1. Экономия времени – сокращение времени для обработки данных о товаре, легкий поиск и получение информации о продукции.
2. Удобство в использовании
3. Многозадачность
4. Хорошая защита
5. Простота в использовании
6. контроль вводимой информации.
7. автосохранение информации.
8. блокировка действий пользователя.

# 1.2 Требования к функциональным характеристикам

Функциональная характеристика - это набор рабочих, эксплуатационных параметров любой техники: прибора, устройства, блока, детали, дающая количественную оценку из свойств.

Данный программный продукт служит для получения, хранения и передачи информации о поступившей продукции.

Программный продукт «Красное & Белое» должен предоставлять администратору возможность:

* добавлять, редактировать, удалять информацию;
* управлять структурой приложения
* вести учет и контроль над совершением различных операций;
* управлять размещением в приложении информации о поступивших товарах;
* управлять достоверностью размещенной информации.

# 1.3 Требования к надёжности и безопасности

Основными требованиями надежности и безопасности разрабатываемого программного продукта следует считать:

* защита информации от копирования и скачивания.
* Двухфакторная аутентификация

# 1.4 Требования к составу и параметрам технических средств

Основными требованиями к составу и параметрам технических документов являются:

1. Двухъядерный процессор 2 ГГц, класса Core 2 Duo 2.4 ГГц или Althon X2 2.7 ГГц
2. Оперативная память: не менее 1 ГБ
3. Свободное место на жестком диске: не менее 100 мегабайт

# 1.5 Требования к информационной и программной совместимости

Для эксплуатации программного продукта необходимо наличие следующих компонентов:

* операционная система семейства Microsoft®Windows® (не ниже 2013).
* доступ к сети-Интернет
* кроссплатформенность
* компьютер – 1280 x 1024.

# Порядок контроля и приёмки

Для проверки выполнения заданных функций программного продукта «Красное & Белое» устанавливаются следующие виды испытаний:

* тестовые испытания;
* опытная эксплуатация;
* приемочные испытания.

Срок приема – сдачи ПО: 09.10.22 – 11.04.23 (6 месяцев).

Условия приема – сдачи:

- работа программного обеспечения «Красное & Белое»

корректна.

- отладка и тестирования пройдены успешно.

- протокол испытаний.

Протокол испытаний – это документ, содержащий результаты испытаний и другую информацию, относящуюся к испытанию (таблица 1.1).

Таблица 1.1 – Протокол испытаний

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №п/п | Дата | | Испытания | Результат |
| 1 | 09.10.2022 | | Моделирование объекта | Созданы инфологическая, даталогическая, физическая модели, а также ER – диаграмма. |
| 2 | 22.10.2022 | | Составление технического задания | ТЗ составлено и согласованно |
| 3 | 20.11.2022 | | Разработка эскизного проекта | Создана DFD-диаграмма и SADT-модель |
| 4 | 21.12.2022 | | Программирование | Написан код |
| 5 | 23.12.2022 | | Отладка и тестирование | Исправлены ошибки |
| 6 | | 24.12.2022 | Внедрение ПП | Использование ПП |
| 7 | | 25.12.2022 | Сопроводительная документации | Составлена документация |

По завершении проектирования технического задания следует приступить к разработке технического проекта.

2 Разработка технического проекта.

2.1. Анализ требований и спецификаций.

Анализ требований – часть процесса разработки программного обеспечения, включающая в себя сбор требований к программному обеспечению (ПО), их систематизацию, выявление взаимосвязей, а также документирование. [1]

Спецификация требований программного обеспечения — структурированный набор требований к программному обеспечению и его внешним интерфейсам. [2] Спецификация требований программного обеспечения описывает функциональные и нефункциональные требования.

IDEF0 - нотация графического моделирования, используемая для создания функциональной модели, отображающей структуру и функции системы, а также потоки информации и материальных объектов, связывающих эти функции.[3]

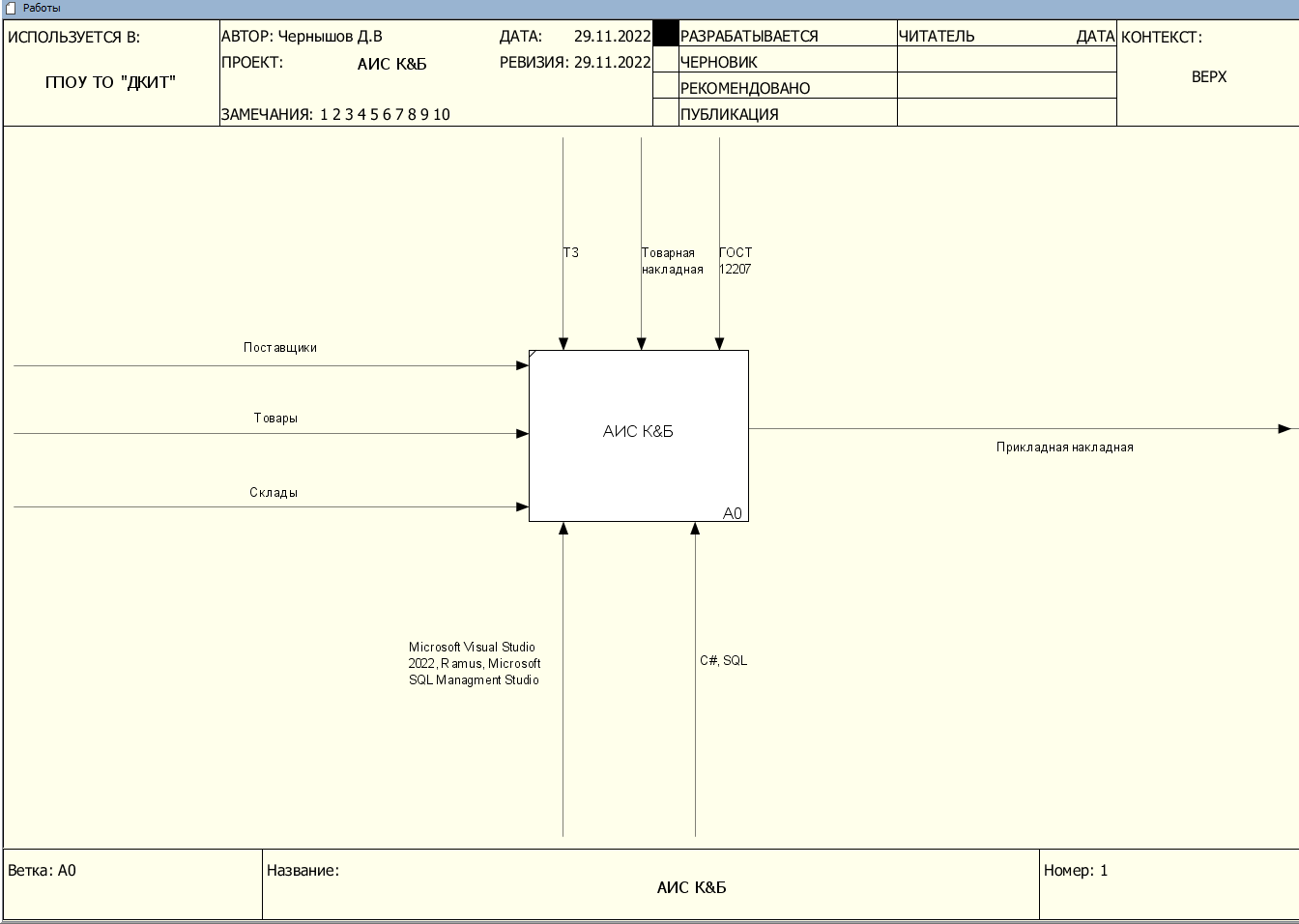
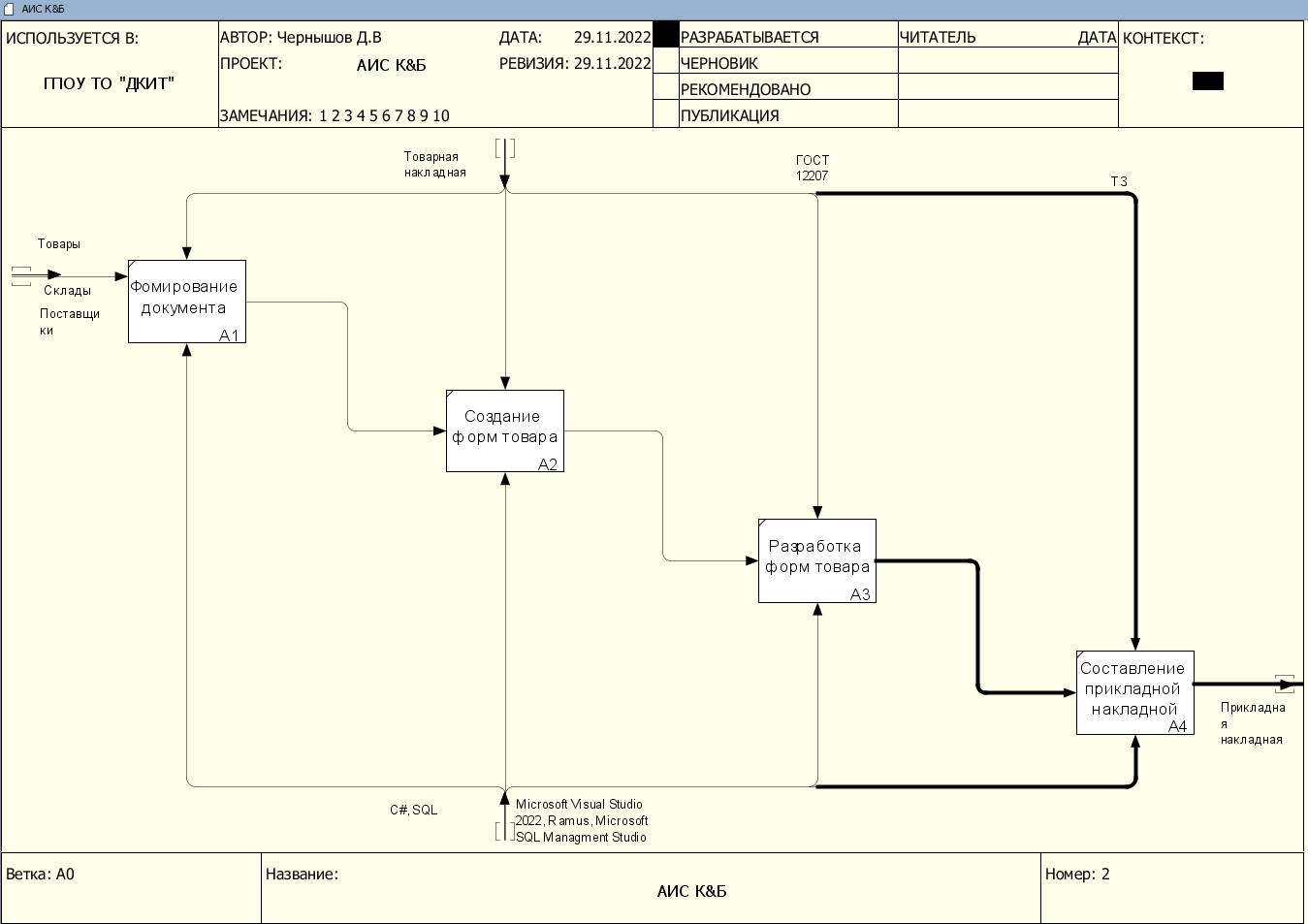


Рисунок 2.1 - Модель IDEF0

IDEF1 — это методология моделирования информационных потоков внутри системы, позволяющая отображать и анализировать их структуру и взаимосвязи []. Одна из основных ценностей и причин стремительного развития информационных технологий — это высочайший темп наращивания человечеством информации. Именно поэтому сразу после IDEF0, где изложена суть функций системы, идёт IDEF1: для последующего анализа и реализации любой системы необходимо чётко и точно определить все данные, которые в ней будут использоваться, и каждый шаг, в котором та или информация будет участвовать для вычисления новой. [4]

 Рисунок 2.2 - Модель IDEF1

ER - диаграмма отношений сущностей (ERD) — это визуальное представление базы данных, которое показывает, как связаны элементы внутри []. Диаграмма ER состоит из двух типов объектов — сущностей и отношений. Сущность в этом контексте — это компонент данных из набора данных, отображаемый в виде фигуры на холсте. Отношения между сущностями представлены в виде строк. [5]

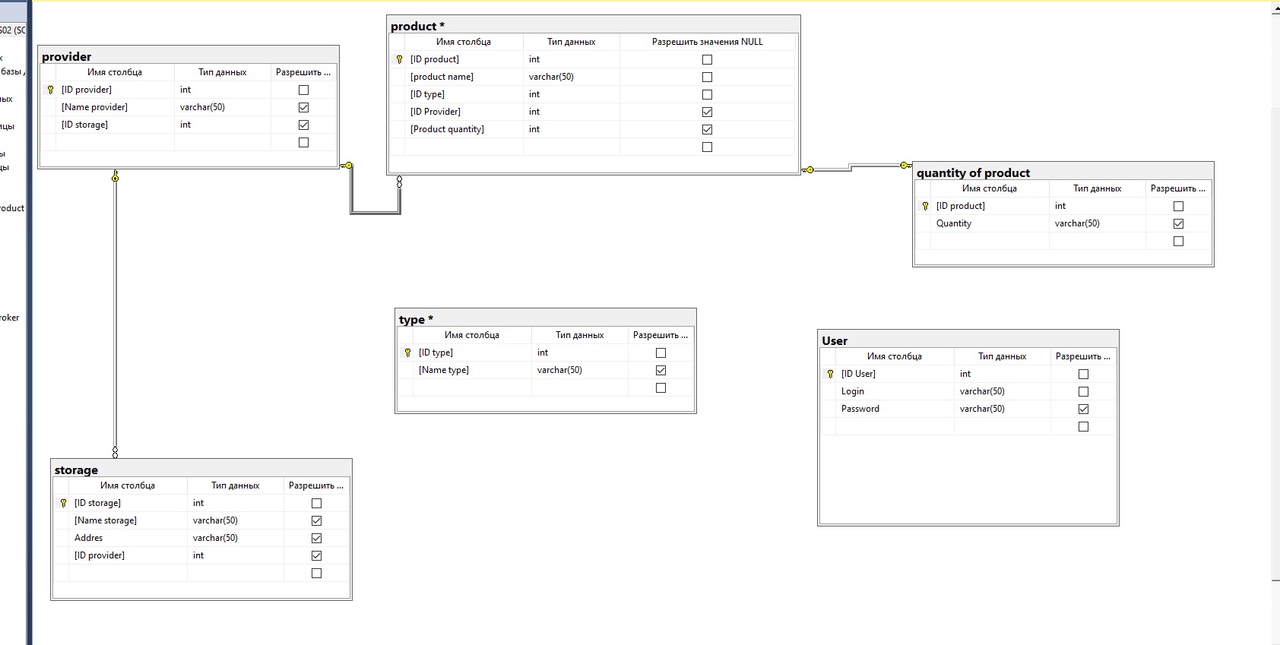


Рисунок 2.3 - ER - диаграмма

Проанализировав требования и спецификации разработки технического проекта необходимо перейти к этапу эскизного проектирования программного обеспечения.

2.2 Этап эскизного проектирования

Эскизный проект – Эскизный проект представляет собой совокупность конструкторских документов, отражающих принципиальные конструктивные решения, дающие общее представление об устройстве и принципе работы изделия, а также его основные параметры и габаритные размеры. Таким образом, на этапе эскизного проекта предусматриваются две процедуры: выбор оптимальных параметров и компоновка. [5]

Прототипирование — это один из начальных этапов разработки, в ходе которого создается предварительный дизайн сайта, лэндинга, приложения или другого проекта (его структура со схематичным изображением основных элементов). [6]

menu

button

Content

(text,img)

Рисунок 2.4 – Экран запуска ПО

Button3

Button2

Button1

Pass

Log

Text

Рисунок 2.5 Меню запуска

Дизайн-макет - это схематичное изображение финальной идеи с указанием всех деталей. В нем указываются концепция, шрифты, тексты, изображения, расположение всех элементов и общая картина продукта. [7]



Рисунок 2.6 - Дизайн макета ПО

2.3 Проектирование внутренней структуры ПО

Во время создания программного обеспечения необходимо верно разработать его структуру, потому что она повлиять на эксплуатацию программного продукта. Для этого важно построить внутреннюю структуру таким образом, чтобы она не ввела пользователя в лабиринт, и наоборот являлась очень легкой в эксплуатации.

В программном продукте «Красное&Белое» применяется линейная структура, которая выглядит в виде цепочки страниц с возможностью перехода из одной в другую.

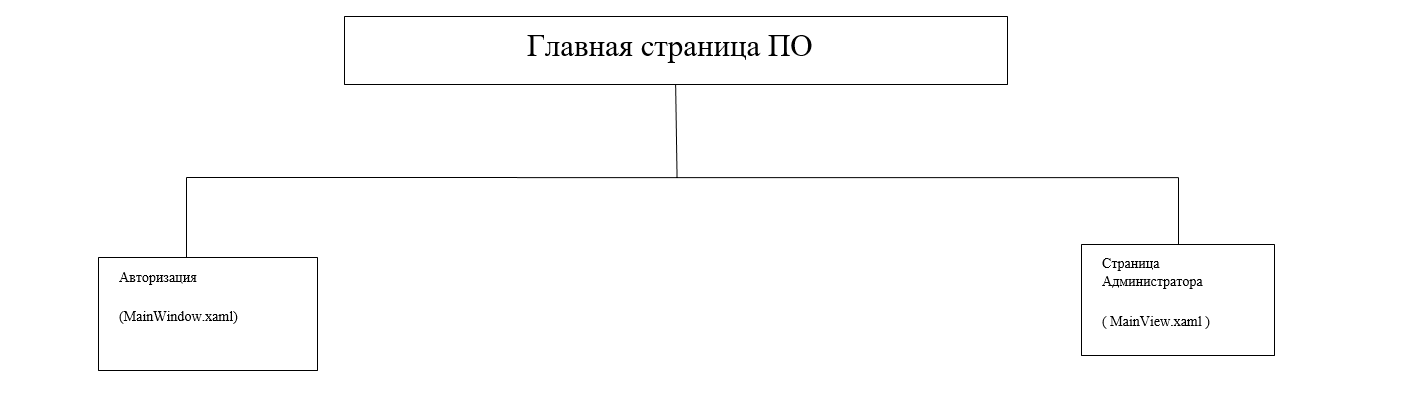


Рисунок 2.8 - Внутренняя структура ПО

Понятная логическая схема программного продукта повысит его эффективность. Такое ПО легче и быстрее внедрять и использовать, проще анализировать их данные, не допуская появления непредвиденных ошибок.

Далее необходимо перейти к реализации программного продукта.

3 Реализация программного обеспечения

3.1 Обоснование выбора средств разработки

Для реализации программного обеспечения следует применить следующие средства разработки: Microsoft Visual Studio и Microsoft SQL Server. Перейдем к определению для того чтобы осознавать с чем и как мы будем работать.

Visual Studio представляет собой полностью интегрированную среду разработки. Она спроектирована таким образом, чтобы делать процесс написания кода, его отладки и компиляции в сборку для поставки конечным потребителям как можно более простым. На практике это означает, что Visual Studio является очень сложным приложением с многодокументным интерфейсом, в котором можно делать практически все, что касается разработки кода.

Далее перейдем к описанию их преимуществ среди аналогов.

Преимуществами Visual Studio являются:

* редактор исходного кода Visual Studio с поддержкой IntelliSense позволяет очень удобно отредактировать исходный код и при этом не потерять нить изменений;
* мощнейший отладчик кода работать как с машинными, так и с исходными кодами;
* Visual Studio позволяет быстро и легко создавать современные облачные приложения;
* Visual Studio имеет полную поддержку .NET и ее унифицированной платформы для веб-приложений;
* поддержка нескольких языков программирования;
* кроссплатформенная поддержка.

Microsoft SQL Server является одной из наиболее популярных систем управления базами данных. Данная СУБД подходит для самых различных проектов: от небольших приложений до больших высоконагруженных проектов. Специфика работы сервера базы данных SQL Server заключается в транзакционной обработке данных. Это означает, что по каждому запросу от СУБД обрабатывается и сохраняется небольшое количество информации. Применение SQL Server позволяет автоматизировать решение различных бизнес-задач, поддерживать проведение аналитики данных в режиме онлайн, отслеживать направление ресурсов СУБД, управлять транзакциями.

SQL Server характеризуется такими особенностями как:

* высокоскоростной доступ к данным, обеспечиваемый надежной клиент-серверной архитектурой СУБД;
* надежность и безопасность – благодаря возможности шифрования данных и резервное копирование;
* простота работы и администрирования, обусловленные понятной структурой языка программирования SQL;
* масштабирование системы
* автоматизация рутинных задач;
* удобны поиск;
* поддержка работы с другими решениями Microsoft;

Также в программе предусмотрена синхронизация, службы преобразования информации и полноценный web-ассистент. Дополнительно в него интегрирован сервис интерактивного анализа.

3.2 Разработка программного обеспечения

Разработка программного продукта начинается с создания проекта в Visual Studio. Далее следует разметка самой главной страницы, что в конечном итоге должно нас привести к данному результату:

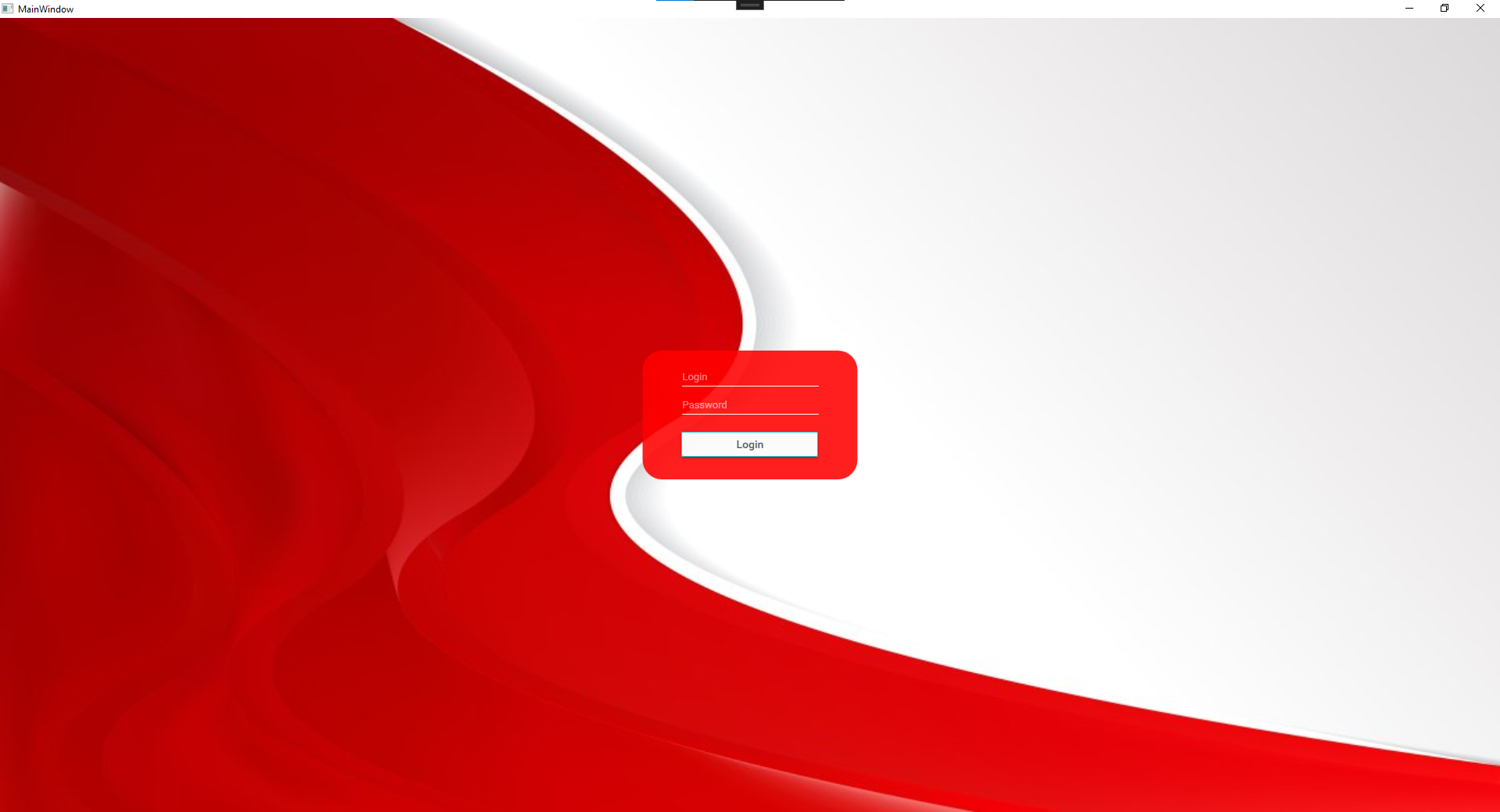


Рисунок 3.1 - Итог разметки MainWindowAuthorizationPage

Общим итогом разметки MainLoginPage является окно авторизации администратора.

Следующим шагом будет разметка основной страницы администратора

MainView.

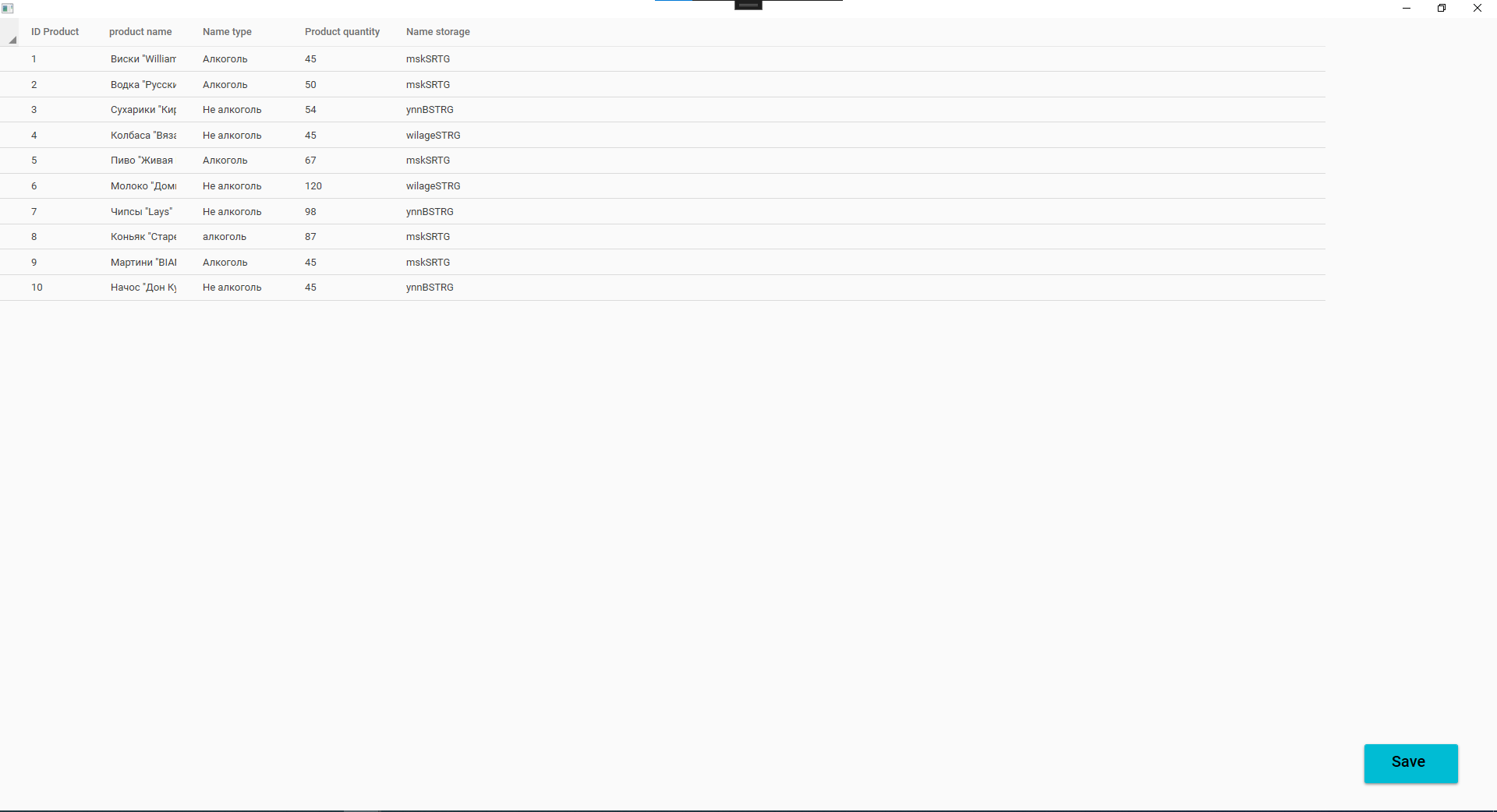


Рисунок 3.3 - Итог разметки MainView

После разметки MainView нам стала доступна возможность просмотра, изменения и сохранение данных о товарах.

Разработав программный продукт необходимо перейти к технико-экономическим показателям для расчета срока окупаемости и доходность.

3.3 Технико-экономические показатели

В состав основных технико-экономических показателей входят:

1. Затраты на программное обеспечение и аппаратное обеспечение;
2. Затраты на услуги персонала;
3. Расчет годовой прибыли.

Расчет затрат на процессе разработки программного продукта для автоматизации палеонтологического музея представлен в таблице 3.1.

Таблица 3.1 - Затраты на ПО

|  |  |  |
| --- | --- | --- |
| Номер | Наименование | Цена |
| 1 | SQL Server | 19 670 ₽ |
| 2 | Microsoft Visual Studio | 0 ₽ |
| 3 | Figma | 0 ₽ |

Далее необходимо перейти к расчету затрат на аппаратное обеспечение.

Таблица 3.2 Затраты на аппаратное оборудование

|  |  |  |
| --- | --- | --- |
| Номер | Наименование | Цена |
| 1 | Компьютер | 14 998 ₽ |
| 2 | Компьютерная мышь | 200 ₽ |
| 3 | Коврик для мыши | 300 ₽ |
| 4 | Клавиатура | 200 ₽ |
| 5 | Принтер со сканером | 4350 ₽ |
| 6 | Монитор | 5780 ₽ |

Подведем расчет затраты на услуги и персонал, внедряющий ПП.

Таблица 3.3 Затраты на персонал и услуги

|  |  |  |
| --- | --- | --- |
| Номер | Наименование | Цена |
| 1 | Услуги  Разработка ПП | 45 498 |
| 2 | Персонал  Заработная плата | 30 000 ₽ |

Проведя все подсчеты, связанные с затратами на разработку высчитана сумма в размере 75 498 рублей.

Кроме затрат, главным составляющим процесса разработки ПП является расчет прибыли за год, и также сроки окупаемости программного продукта.

Расчет годовой прибыли следует выполнить по кварталам, с расчетом дохода, расхода и прибыли.

Для расчета дохода требуется рассмотреть количество проданного ПО за каждый квартал и стоимость разработанного ПП за каждый квартал и стоимость разработанного ПО за единицу, в данном случаем 25 000 рублей по формуле: Доход = Количество, проданного ПО \* Стоимость за единицу. (3.1)

К расходам стоит отнести заработную плату разработчику в размере 20 00 рублей в месяц, а в квартал 70 000 рублей.

Таблица 3.4 Расчет годовой прибыли

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Расчет годовой прибыли | | | | | |
|  | Квартал 1 | Квартал 2 | Квартал 3 | Квартал 4 | Год |
| Кол-во проданного ПО | 7 шт | 5 шт | 3 шт | 8 шт | 23 шт |
| Доход | 175 000₽ | 125 000₽ | 75 000₽ | 200 000₽ | 575 000₽ |
| Расход | 70 000₽ | 70 000₽ | 70 000₽ | 70 000₽ | 250 000₽ |
| Прибыль | 105 000₽ | 55 000₽ | 5 000₽ | 130 000₽ | 305 000₽ |

Таким образом, выполнен расчет годовой прибыли, основываясь на подсчете всех доходов и расходов.

Срок окупаемости – ключевой фактор в оценке реализации и внедрения программного продукта. Заказчику необходимо определить, сколько времени понадобится, чтобы окупить затраты. Для определения срока окупаемости используется формула (3.2):

PP = K0 / ПЧгс (3.2)

PP – срок окупаемости, выраженный в годах;

K0 – сумма вложенных средств;

ПЧгс – чистая годовая прибыль.

PP = 75 498/ 335 000 = 0,24 (3.3)

Таким образом, срок окупаемости составляет 2 с половиной месяца, что говорит о рентабельности создания ПП.

ЗАКЛЮЧЕНИЕ

В ходе выполнения курсовой работы был разработан программный продукт, а именно разработана информационная система учета продукции в магазине «Красное&Белое», целью которого являлась разработка программного обеспечения, осуществляющего услуги редактирования, добавления, удаление и предоставления информации о товаре. При разработке данного программного продукта были выполнены следующие задачи:

- составлено техническое задание;

- созданы эскизы и макеты ПП для дальнейшей реализации;

- выбраны технологии и средства разработки программного продукта;

- выполнено создание и разработка ПП;

- выполнена отладка и тестирование.

В ходе разработки программного продукта были использованы следующие программные средства: среда разработки Visual Studio, система управления реляционными базами данных Microsoft SQL Server, графический онлайн-редактор Figma.

Таким образом цель курсовой работы достигнута и решены поставленные задачи.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Тема 3. Информационные системы и технологии [Электронный ресурс] URL:[https://studfile.net/preview/4171546/page](https://studfile.net/preview/4171546/page:4/)(дата обращения: 23.12.2022)

2. Курсовая работа: Управление рисками внедрения информационных систем предприятия [Электронный ресурс] URL: <https://ronl.org/kursovyye-raboty/informatika/322024/>(дата обращения: 23.12.2022)

3. Анализ требований [Электронный ресурс] URL: <https://habr.com/ru/post/340956/>(дата обращения: 23.12.2022)

4. Спецификация требований программного обеспечения [Электронный ресурс] URL:

<https://ru.wikipedia.org/wiki/>(дата обращения: 23.12.2022)

5. 5.5. Нотация idef0 [Электронный ресурс] URL: <https://studfile.net/preview/10081394/page:7/>(дата обращения: 23.12.2022)

6. Моделирование бизнес-процессов [Электронный ресурс] URL: <https://studfile.net/preview/5441370/>(дата обращения: 23.12.2022)

7. Диаграммы сущность-связь (erd) [Электронный ресурс] URL: <https://studfile.net/preview/3828360/page:7/>(дата обращения: 23.12.2022)

8. Эскизный проект [Электронный ресурс] URL: <https://studfile.net/preview/9406747/page:3/>(дата обращения: 23.12.2022)

9. UNISENDER Словарь Маркетолога Прототипирование [Электронный ресурс] URL: <https://www.unisender.com/ru/glossary/chto-takoe-prototipirovanie-i-zachem-ono-nuzhno/>(дата обращения: 23.12.2022)

10. UNISENDER Словарь Маркетолога Дизайн-макет [Электронный ресурс] URL: <https://www.unisender.com/ru/glossary/chto-takoe-prototipirovanie-i-zachem-ono-nuzhno/>(дата обращения: 23.12.2022)

11. Сложные свойства и конвертеры типов Metanit [Электронный ресурс] [URL: https://metanit.com/sharp/wpf/2.3.php](URL:%20https://metanit.com/sharp/wpf/2.3.php) (дата обращения: 23.12.2022)

12. Методы Metanit [Электронный ресурс] URL: <https://metanit.com/sharp/tutorial/2.8.php>

13.Лавровская, О.Б. Технические средства информатизации: практикум: учебное пособие для ссузов по напр. подготовки 230000 "Информатика и вычислительная техника" и спец. "Обеспечение информационной безопасности автоматизированных систем". - Москва: Академия, 2019. - 207 с.

14. Кудрина, Е. В. Основы алгоритмизации и программирования на языке c#: учеб. пособие для СПО / Е. В. Кудрина, М. В. Огнева. — М.: Издательство Юрайт, 2019. — 322 с.

15.Технологии разработки программного обеспечения, 2016. [Электронный ресурс]. URL: https://vscode.ru/articles/tehnologiya-razrabotki-po.html (дата обращения: 23.12.2022)

14. Разработка и эксплуатация автоматизированных информационных систем : учеб. пособие для ссузов по спец. "Информатика и вычислительная техника". - 6-е изд. ; стер. - Москва : Академия, 2018. - 301 с.

1. Технологии разработки программного обеспечения, 2016. [Электронный ресурс]. URL: https://vscode.ru/articles/tehnologiya-razrabotki-po.html (дата обращения: 15.12.2022)
2. Проектирование информационной системы, 2017. [Электронный ресурс]. URL:https://prog.bobrodobro.ru/58593(дата обращения: 20.12.2022)
3. Фаулер М. Рефакторинг. Улучшение существующего кода. – Пер. с англ. – СПб: Символ-Плюс, 2003. – 432 с., ил.
4. Коберн А. Современные методы описания функциональных требований к системам. М: Издательство «Лори», 2017. 263 с.: ил.

ПРИЛОЖЕНИЕ А

Результат работы программного продукта

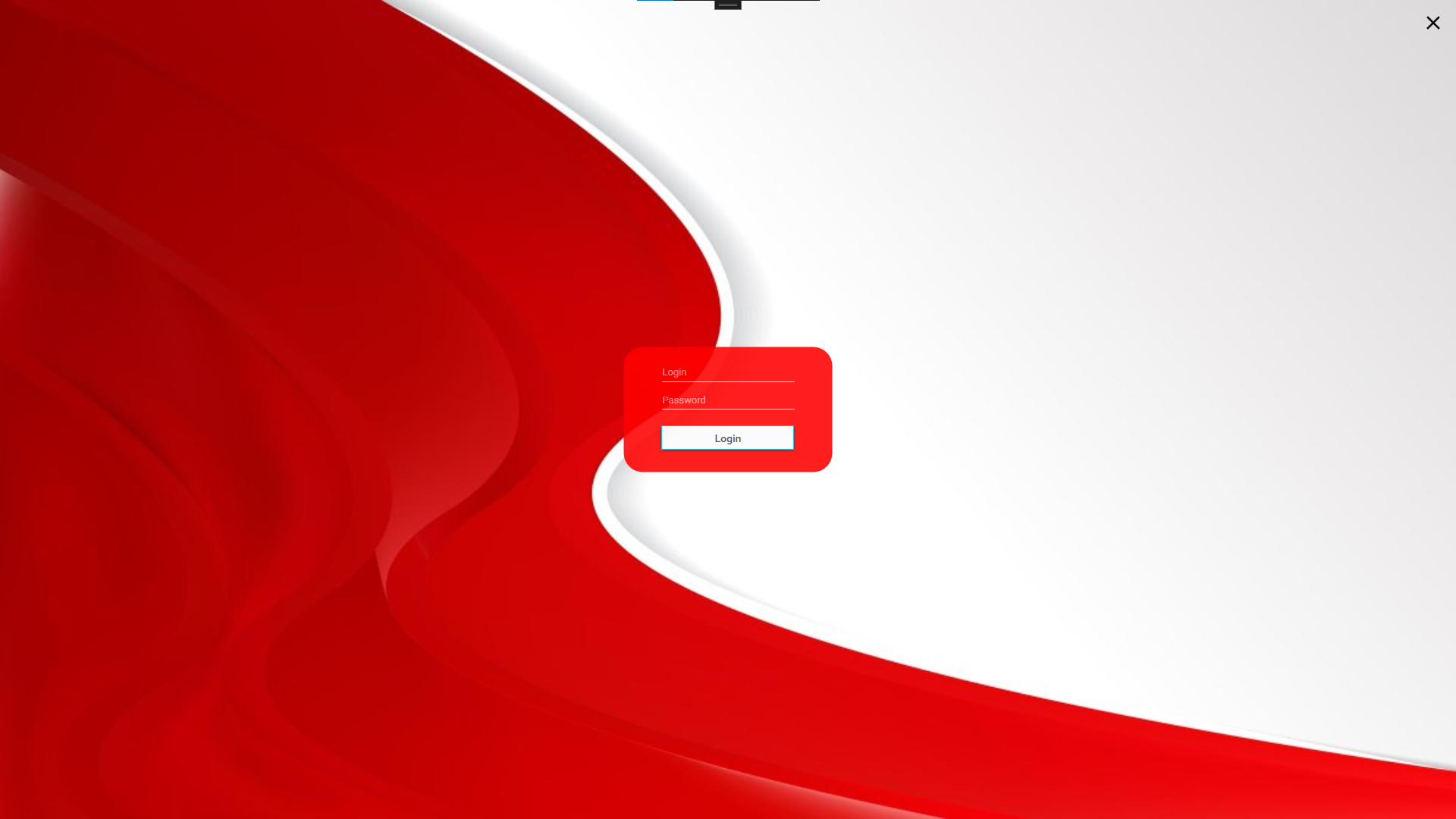


Рисунок 3.3 - Итог 1 страницы

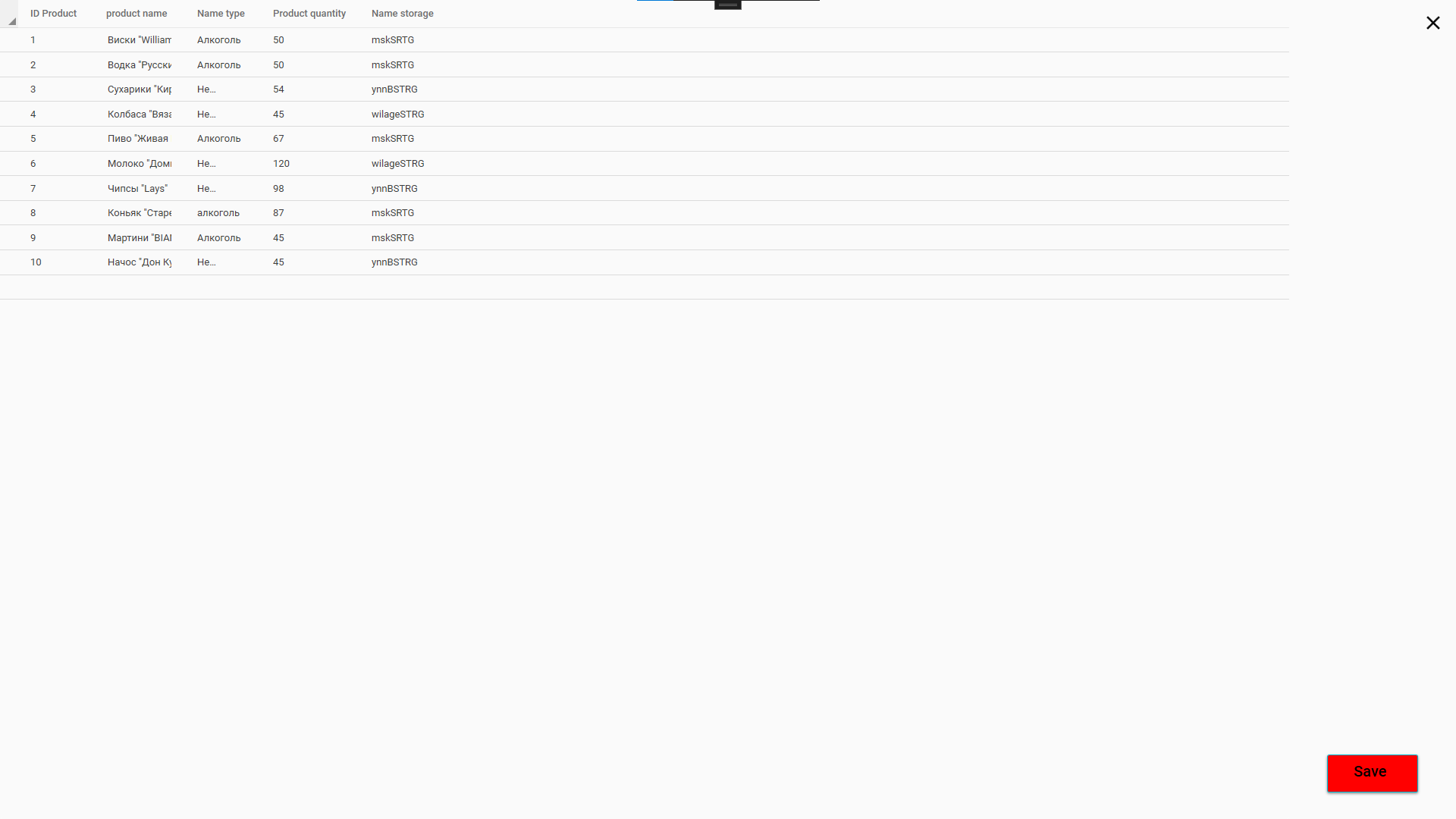


Рисунок 3.4 - Итог 2 страницы

ПРИЛОЖЕНИЕ Б

Листинг программного обеспечения

<Window

x:Class="KrasnoeBeloe.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:KrasnoeBeloe"

xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

Title="MainWindow"

Width="800"

Height="450"

WindowStyle="None"

WindowState="Maximized"

Background="{DynamicResource MaterialDesignPaper}"

FontFamily="{DynamicResource MaterialDesignFont}"

TextElement.FontSize="13"

TextElement.FontWeight="Regular"

TextElement.Foreground="{DynamicResource MaterialDesignBody}"

TextOptions.TextFormattingMode="Ideal"

TextOptions.TextRenderingMode="Auto"

mc:Ignorable="d">

<Border>

<Border.Background>

<ImageBrush ImageSource="/Image/kartinka.png" />

</Border.Background>

<Grid>

<Button Width="50" Height="50" x:Name="BtnClose" Margin="5 5"

Background="Transparent" Click="BtnClose\_Click" Cursor="Hand" HorizontalAlignment="Right" VerticalAlignment="Top" BorderBrush="Transparent">

<materialDesign:PackIcon Kind="Close" Width="30" Height="30" HorizontalAlignment="Center"/>

</Button>

<Border

Width="275"

Height="165"

Background="Red"

CornerRadius="25"

Opacity="0.88">

<StackPanel Margin="10,10" Orientation="Vertical">

<TextBox

x:Name="TxbLogin"

Width="175"

VerticalAlignment="Center"

materialDesign:HintAssist.Hint="Login"

BorderBrush="White"

Foreground="White"

Style="{StaticResource MaterialDesignFloatingHintTextBox}"/>

<TextBox

x:Name="TxbPassword"

Width="175"

VerticalAlignment="Center"

materialDesign:HintAssist.Hint="Password"

BorderBrush="White"

Foreground="White"

Style="{StaticResource MaterialDesignFloatingHintTextBox}"/>

<Button

Width="175"

Margin="10,22,10,10"

Click="Button\_Click"

Content="Login"

IsEnabled="{Binding DataContext.ControlsEnabled, RelativeSource={RelativeSource FindAncestor, AncestorType=Window}}"

Style="{StaticResource MaterialDesignPaperButton}" />

</StackPanel>

</Border>

</Grid>

</Border>

</Window>

<Window x:Class="KrasnoeBeloe.MainView"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes"

xmlns:local="clr-namespace:KrasnoeBeloe"

mc:Ignorable="d"

WindowStyle="None"

WindowState="Maximized"

TextElement.Foreground="{DynamicResource MaterialDesignBody}"

TextElement.FontWeight="Regular"

TextElement.FontSize="13"

TextOptions.TextFormattingMode="Ideal"

TextOptions.TextRenderingMode="Auto"

Background="{DynamicResource MaterialDesignPaper}"

FontFamily="{DynamicResource MaterialDesignFont}"

Height="675"

Width="1300">

<Border>

<Grid>

<Grid.ColumnDefinitions>

<ColumnDefinition/>

<ColumnDefinition Width="220"/>

</Grid.ColumnDefinitions>

<Button Width="50" Height="50" x:Name="BtnClose" Margin="5 5"

Background="Transparent" Click="BtnClose\_Click\_1" Cursor="Hand" HorizontalAlignment="Right" VerticalAlignment="Top" BorderBrush="Transparent" Grid.Column="1">

<materialDesign:PackIcon Kind="Close" Width="30" Height="30" HorizontalAlignment="Center"/>

</Button>

<Border Grid.Column="1" >

<StackPanel VerticalAlignment="Bottom" Margin="35">

<Button x:Name="BtnClk" Height="50" Width="120" Click="BtnClk\_Click" Background="Red">

<TextBlock Text="Save" Width="50" Height="30" FontSize="20"/>

</Button>

</StackPanel>

</Border>

<DataGrid

x:Name="DataEmploerInfo"

AutoGenerateColumns="False"

CanUserAddRows="True"

HeadersVisibility="All">

<DataGrid.Columns>

<DataGridTextColumn Binding="{Binding ID\_product }"

Width="50" MinWidth="100"

EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnEditingStyle}"

ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}"

Header="ID Product"

IsReadOnly="True"/>

<DataGridTextColumn Width="50" MinWidth="120"

Header="product name"

Binding="{Binding product\_name, Mode=TwoWay}"/>

<DataGridTextColumn Width="30" MinWidth="100"

EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}"

ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}"

Header="Name type"

Binding="{Binding Name\_type}"/>

<DataGridTextColumn Width="30" MinWidth="130"

EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}"

ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}"

Header="Product quantity"

Binding="{Binding Product\_quantity}"/>

<DataGridTextColumn Width="30" MinWidth="130"

EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}"

ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}"

Header="Name storage"

Binding="{Binding Name\_storage}"/>

</DataGrid.Columns>

</DataGrid>

</Grid>

</Border>

</Window>