Министерство образования Московской области

Государственное образовательное учреждение высшего образования Московской области

«Государственный гуманитарно-технологический университет»

Ликино-Дулевский политехнический колледж – филиал ГГТУ

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

Дипломный проект

Разработка приложения по автоматизации процесса складского учета для клиента организации ООО «ЦА «Максималист»

Выполнил:

Стародубцев Максим Сергеевич \_\_\_\_\_\_\_\_\_\_\_

студент группы ИСП.20А\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

по специальности

09.02.07 Информационные системы и программирование

очной формы обучения

Руководитель:

Селиверстова Ольга Михайловна \_\_\_\_\_\_\_\_\_

Оценка:

\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ликино-Дулево

2024 год

**Содержания**

[Введение 3](#_Toc167721705)

[1. Разработка системного проекта 6](#_Toc167721706)

[1.1. Назначение разработки 6](#_Toc167721707)

[1.2. Требования к функциональным характеристикам 6](#_Toc167721708)

[1.3. Требования к надёжности и безопасности 6](#_Toc167721709)

[1.4. Требования к составу и параметрам технических свойств 6](#_Toc167721710)

[1.5. Требования к информационной и программной совместимости 6](#_Toc167721711)

[2. Разработка технического проекта 6](#_Toc167721712)

[2.1. Обоснование выбора Case – средств, Case – технологии 6](#_Toc167721713)

[2.2. Проектирование модели данных 6](#_Toc167721714)

[2.3. Детальное проектирование интерфейса 6](#_Toc167721715)

[2.4. Функциональная схема 6](#_Toc167721716)

[3. Реализация 6](#_Toc167721717)

[3.1. Обоснование выбора средств разработки 6](#_Toc167721718)

[3.2. Руководство системного программиста 6](#_Toc167721719)

[3.3. Руководство программиста 6](#_Toc167721720)

[3.4. Руководство пользователя 6](#_Toc167721721)

[4. Тестирование и отладка 6](#_Toc167721722)

[4.1. Виды тестирования 6](#_Toc167721723)

[4.2. Отладка приложения 6](#_Toc167721724)

[5. Расчет базовой себестоимости разрабатываемого продукта 7](#_Toc167721725)

[5.1. Исходные данные 7](#_Toc167721726)

[5.2. Расчёты затрат на выполнение программы 7](#_Toc167721727)

[5.3. Расчет отчислений на социальное страхование и обеспечение 8](#_Toc167721728)

[5.4. Расчет базовой себестоимости компьютерного продукта 8](#_Toc167721729)

[Заключение 11](#_Toc167721730)

[Список литературы 11](#_Toc167721731)

[Приложения 11](#_Toc167721732)

# **Введение**

В настоящее время на торговых предприятиях особенно важны вопросы обеспечения процесса торговли необходимыми материальными ресурсами и эффективного использования этих ресурсов, совершенствования управления складским хозяйством и запасами продукции, организации сбытовой деятельности и управления транспортными потоками. Изменчивость спроса потребителей требует от торговых предприятий постоянного анализа рынка, чтобы поддерживать ассортимент продукции на уровне, достаточном для удовлетворения спроса. На основе проведенных исследований на 45 предприятиях, в том числе 65% торговых, выяснилось, что для развития управления складами важными факторами являются повышение требований клиентов к обслуживанию, улучшение качества складского сервиса (время, качество, надежность), увеличение расходов на дистрибуцию, эффективное управление запасами и внедрение информационных технологий.

Для обеспечения конкурентоспособности торговое предприятие должно организовать свою деятельность таким образом, чтобы минимизировать риски, ущерб и издержки, связанные с процессами хранения и сбыта продукции, и максимизировать прибыль от ее продажи. Для этого в первую очередь нужно повысить эффективность управления складским хозяйством.

**Актуальность и значимость рассматриваемой проблемы** –

Выбранная тема становится актуальной в связи с тем, что складское хозяйство имеет значительный потенциал для улучшений, которые, полностью реализованы, окажут воздействие на другие аспекты деятельности торгового предприятия.

**Цель исследования** – Целью выпускной квалификационной работы заключается в оценке управления складом на предприятии и разработке плана действий для улучшения складских операций на предприятии.

**Задачи:**

1. Изучение литературы, относящейся к предметной области;
2. Формирование требований к разрабатываемому программному продукту;
3. Разработка программного продукта;

4. Разработка полной и понятной документации руководства пользователя и программиста;

5. Расчет базовой стоимости программного продукта.

**Методы**:

1. Microsoft Visual Studio 2022– для запуска и разработки приложения
2. Figma – для проектирования интерфейса
3. BPwin – для проектирования диаграммы.
4. Microsoft Excel – для вывода отчетности из программы.
5. Microsoft SQL Server Management Studio 18 – интегрированная среда для управления любой инфраструктурой SQL, от SQL Server до баз данных;

**Результатом работы** является приложение, по продажи товара со склада клиенту

**Предлагаемая работа** состоит из введения, пяти разделов, заключения и приложения. Во введении обосновывается актуальность темы, формулируются цель, задачи и практическая значимость работы.

Первый раздел посвящен описанию назначение разработки, требованию к функциональным характеристикам, требованию к надёжности и безопасности, требованию к составу и параметрам технических свойств, а также требованию к информационной и программной совместимости.

Второй раздел является разработкой технического проекта в Case-средствах. Обоснование выбора Case – средств, Case – технологии, проектирование модели данных, детальное проектирование интерфейса, функциональная схема.

Третий раздел описывает обоснование выбора средств разработки, руководство системного программиста, руководство программиста, руководство пользователя

Четвертый раздел посвящен виды тестирования, отладка приложения.

Пятый раздел описывает расчет базовой стоимости продукта. исходные данные, расчёты затрат на выполнение программы, расчет отчислений на социальное страхование и обеспечение, расчет базовой себестоимости компьютерного продукта.

В заключении описываются основные результаты выпускной квалификационной работы.

Приложение содержит …..*листинги основных программных модулей, используемых в разработанной системе, а также диаграммы и схемы…(НАПИСАТЬ СВОЕ)*.

Общий объем работы 59 страницы. Список литературы содержит 15 источников.

# **Разработка системного проекта**

## **Назначение разработки**

Автоматизированная информационная система «SalesOfMaterials» предназначена для продажи товара со склада для оптовой торговли. Пользователями программы выступают сотрудник, кладовщик и администратор. Продажа товаров к заказчикам осуществляется на основании договоров купли-продажи, в которых оговариваются условия поставки. Акте приема-передачи указываются координаты, где продукция будет передана покупателю, наименование товара, количество, цена и сумму. Товарная накладная предназначена для оформления операций по отпуску и приёму товаров со склада. Данные первичных документов по сбыту товаров, содержащем название документа по разгрузке, его дату и номер, краткую характеристику документа, дату регистрации документа, сведения о проданных товаров.

Оформление и учет реализации товаров зависят от способа расчета за приобретаемые товары между покупателем и продавцом. Товары реализуются за наличный и безналичный расчет. Сотрудник ведет журнал продажи товаров, где указывается: номер по порядку, дата отпуска, наименование товаров, количество единиц и сумма отпуска, фамилия, инициалы и подпись лица, выдавшего товар. Данные первичных документов фиксируются в карточках учета, которые выполняют роль регистров складского учета. Кладовщик распределяет товар по заказам. Администратор раздаёт права пользователям и проверяет документы на распределения.

## **Требования к функциональным характеристикам**

Приложение представляет возможности по продажи товара со склада. Пользователи приложения смогут создавать аккаунты и авторизироваться в существующие. Пользователи имеют возможность смотреть, редактировать и удалять данные в приложении (материалы, состав материала, склады, контрагенты, поступления, состав поставки, движения поступлений, хранение материалов, пользователи). Также пользователь сможет найти с помощью поиска, сортировки и фильтрации нужные данные. Оформление приходной накладной с помощью Excel и Создание диаграммы по продажи за (месяц, квартал, год)

## **Требования к надёжности и безопасности**

Программа должна быть в достаточной степени надёжна от сбоев. На крайний случай предусмотрено сохранение данных БД в приложении «Microsoft SQL» или восстановление данных в случае завершения работы. Также в приложение присутствует авторизация для защиты личных данных

## **Требования к составу и параметрам технических свойств**

Таблица №1 «Требования к составу и параметрам тех. средств»

|  |  |
| --- | --- |
| Процессор | Intel® Core™ i5-6400 CPU @ 2.70GHz |
| Оперативная Память | 2 x 4 ГБ, DDR4, DIMM, 2133 МГц |
| Разрешение экрана | 1920x1080 |
| Размер экрана | 20 дюйма |
| Устройства ввода | Мышь, клавиатура |
| Дисковое пространство | 1Гб |
| Подключаемые устройства | Принтер |
| Операционная система | Windows 8.1 10 11 |

## **Требования к информационной и программной совместимости**

Для корректной работы программы необходимо:

ОС Windows 10/11 – операционная система.

Visual Studio 2022 – для запуска и работы приложения.

Microsoft SQL Server Management Studio 18 – интегрированная среда для управления любой инфраструктурой SQL, от SQL Server до баз данных;

Microsoft Excel – для вывода отчетности из программы.

# **Разработка технического проекта**

## **Обоснование выбора Case – средств, Case – технологии**

Выбор инструмента CASE значительно определяется используемой методикой проектирования информационных систем (ИС). Ключевые методики включают структурный (или функциональный) подход, объектно-ориентированный подход, а также методологию ARIS, которая рассматривается отдельно.

Структурный подход к созданию ИС основан на разложении системы на функции, подлежащие автоматизации. Это означает, что система делится на функциональные подсистемы, которые далее разбиваются на более мелкие подфункции и задачи. В настоящее время популярностью пользуются следующие инструменты:

CA ERwin Process Modeler (бывший BPwin)

CA ERwin Data Modeler (бывший ERwin)

Vantage Team Builder

Oracle Designer

BPwin является эффективным инструментом для моделирования, который применяется в анализе, документации и оптимизации сложных бизнес-процессов. С его помощью можно выявлять и устранять конфликты, достигая гармонизации процессов. Вот основные функции BPwin:

* **Моделирование функций (IDEF0)**: проведение системного анализа бизнес-процессов, включая изучение регулярно выполняемых функций, используемых ресурсов и получаемых результатов.
* **Моделирование потоков данных (DFD)**: визуализация потоков данных, которые циркулируют между различными операциями.
* **Моделирование потоков работ (IDEF3)**: анализ этапов бизнес-процесса и точек принятия решений, оказывающих влияние на его протекание.

Выполнение работы с применением СASE-технологий: IDEF(), IDEF (3), DFD.

## **Проектирование модели данных**

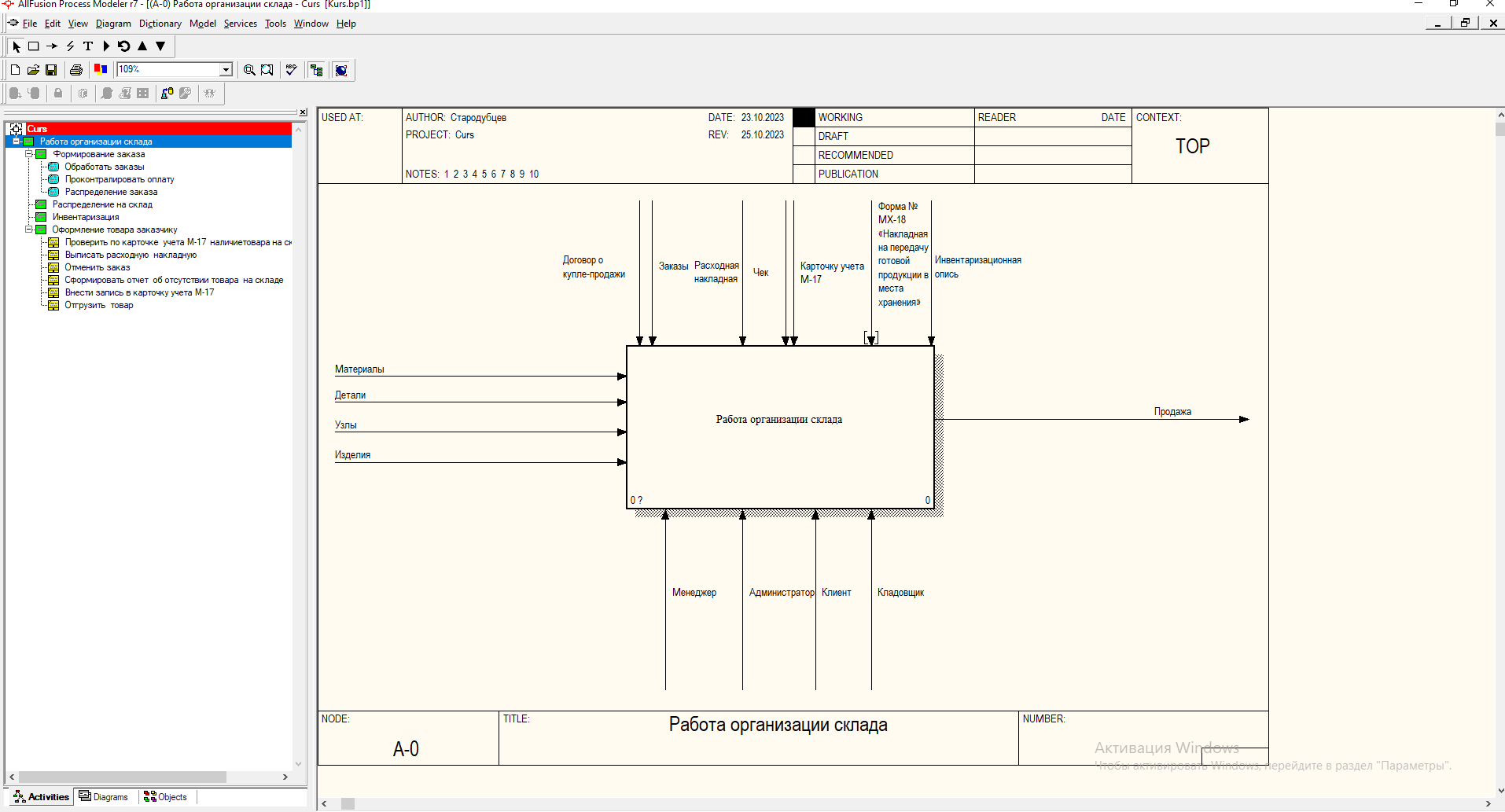


Рис.1 «Контекстная диаграмма»

Выполнение процесса декомпозиции контекстной диаграммы.

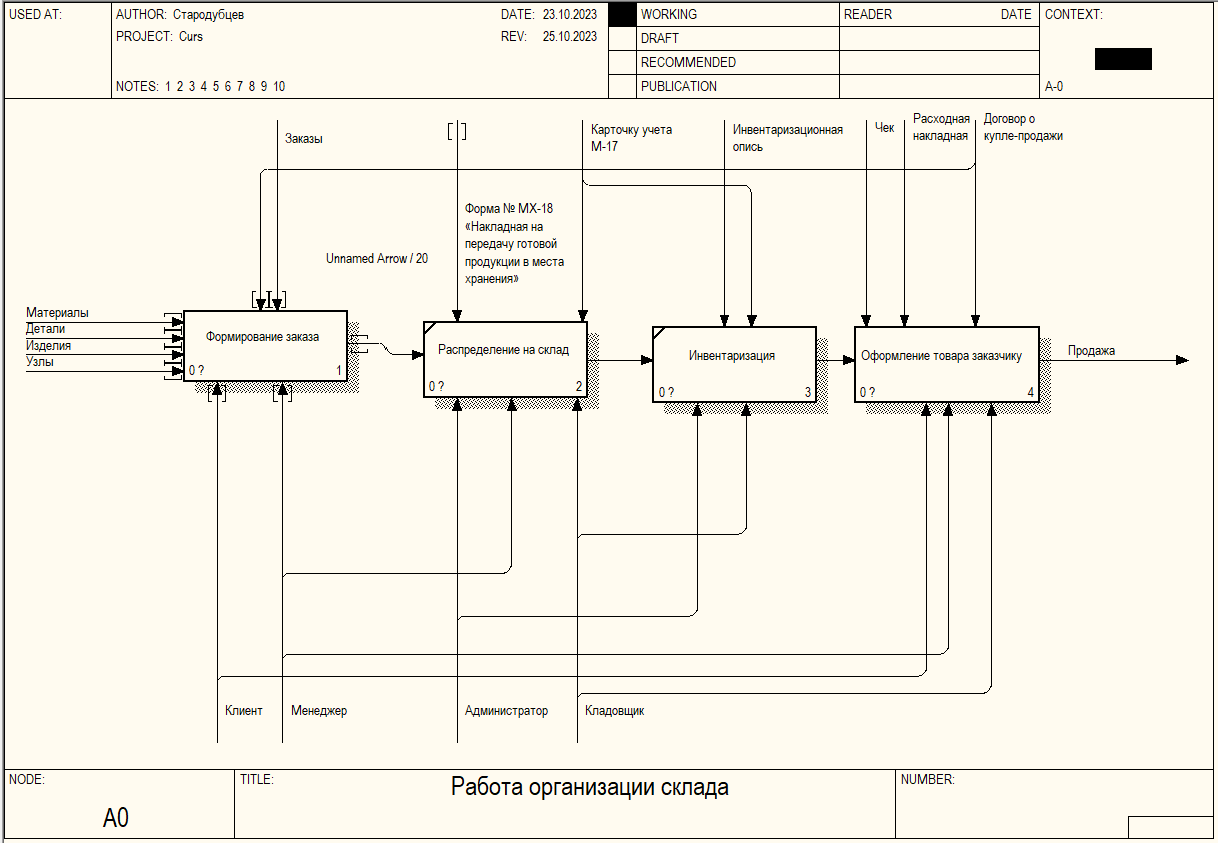


Рис.2 «Процесс декомпозиции»

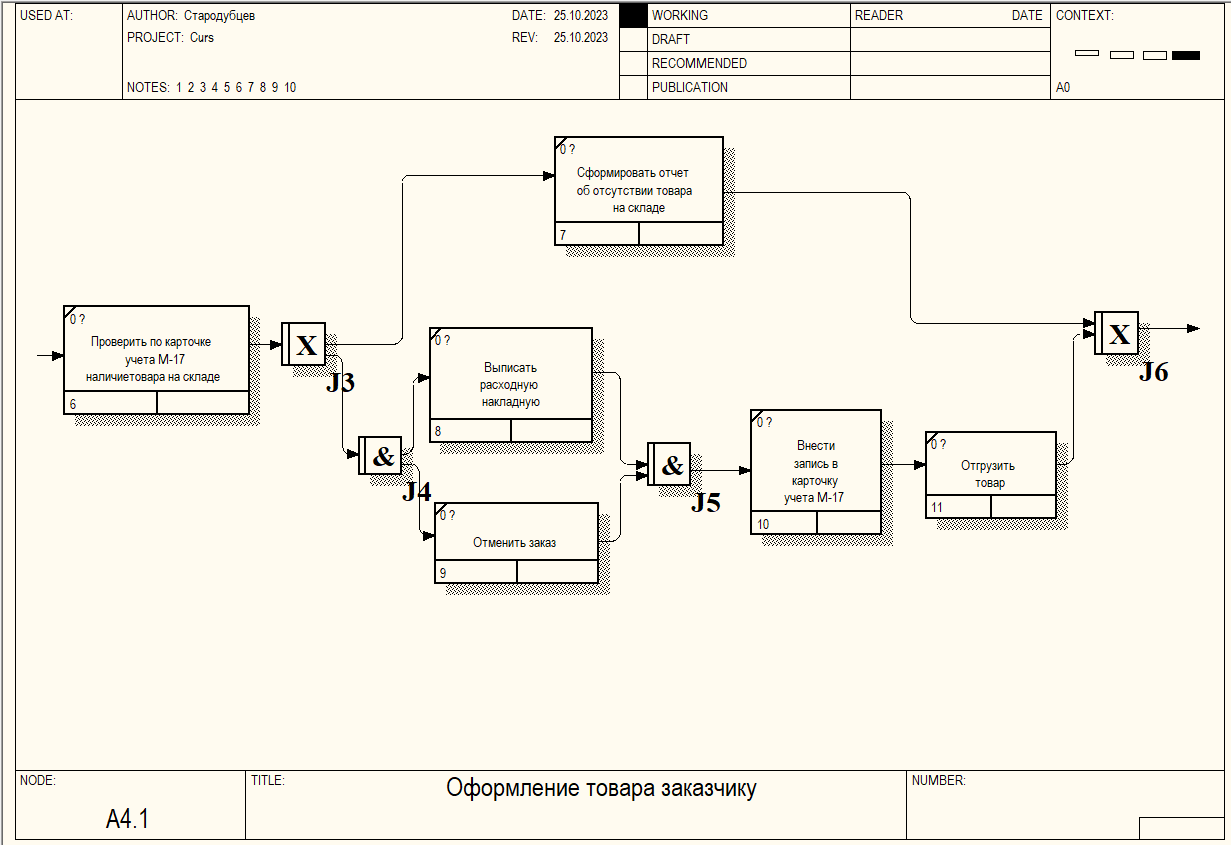


Рис.3 «Логическое представление процесса «Оформление товара заказчику»

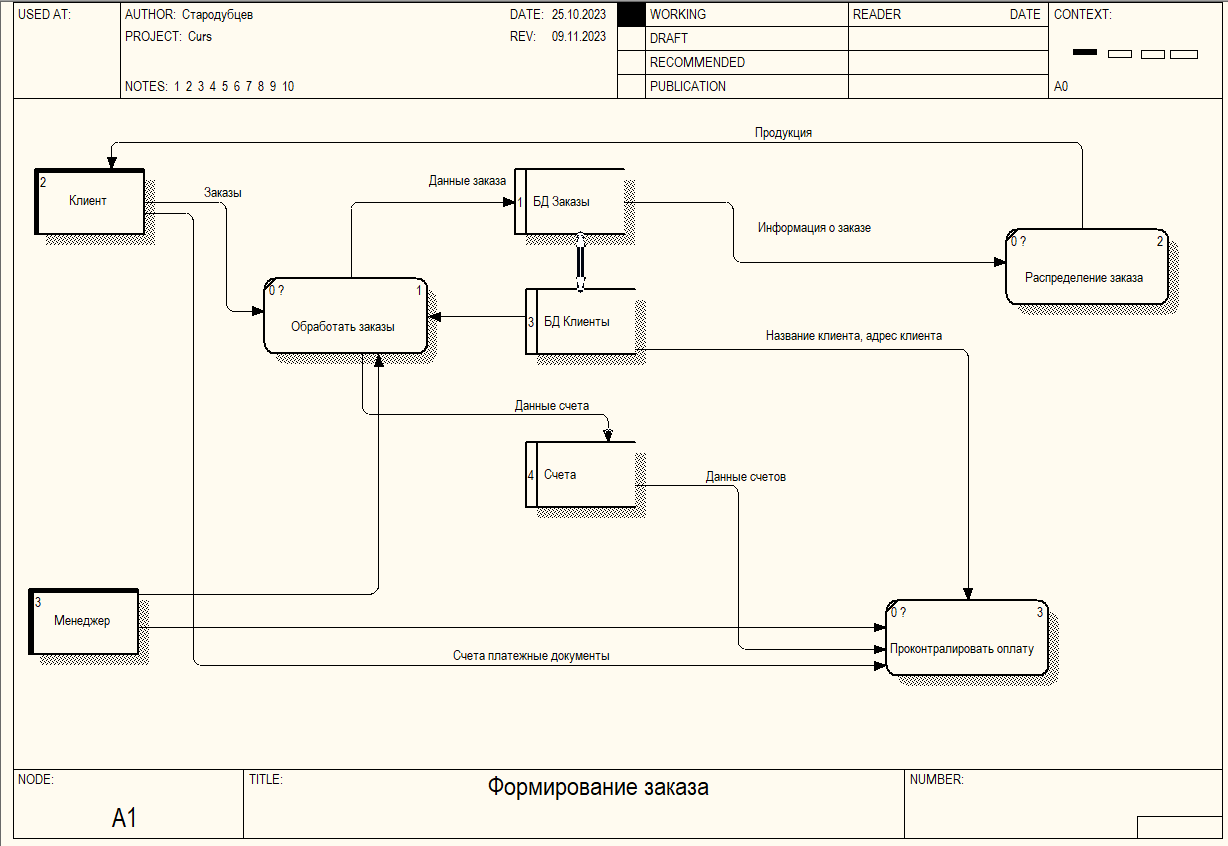


Рис.4 «Хранилище данных»

## **Детальное проектирование интерфейса**

WPF (Windows Presentation Foundation) — это инструмент для создания пользовательского интерфейса в приложениях на операционной системе Windows. Он использует язык XAML и основан на векторной системе визуализации, которая не зависит от разрешения дисплея и учитывает возможности современной графики.

Ключевые особенности WPF:

* векторная система визуализации для высокой производительности и поддержки разных разрешений;
* язык XAML для описания интерфейса, позволяющий разработчикам и дизайнерам сотрудничать;
* встроенные стили элементов и поддержка двухмерной и трёхмерной графики;
* привязка данных для связывания различных данных с элементами управления;
* макеты и шаблоны для создания адаптивных пользовательских интерфейсов;
* поддержка мультимедиа, текста и документов.

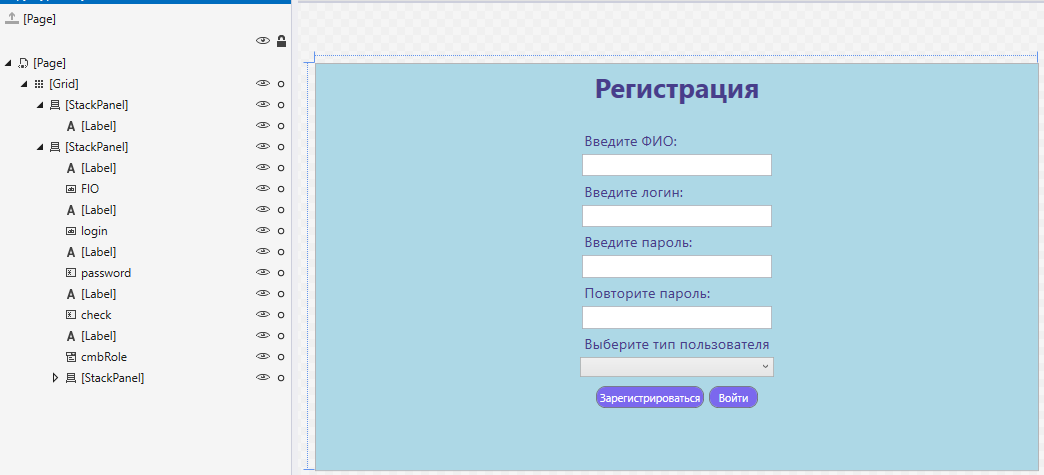


Рис.5 «Регистрация»

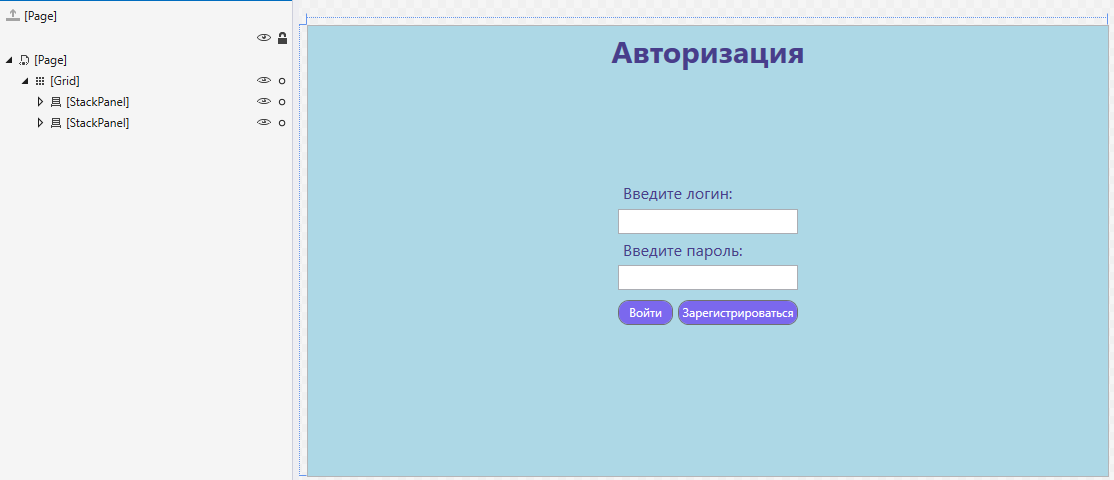


Рис.6 «Авторизация»

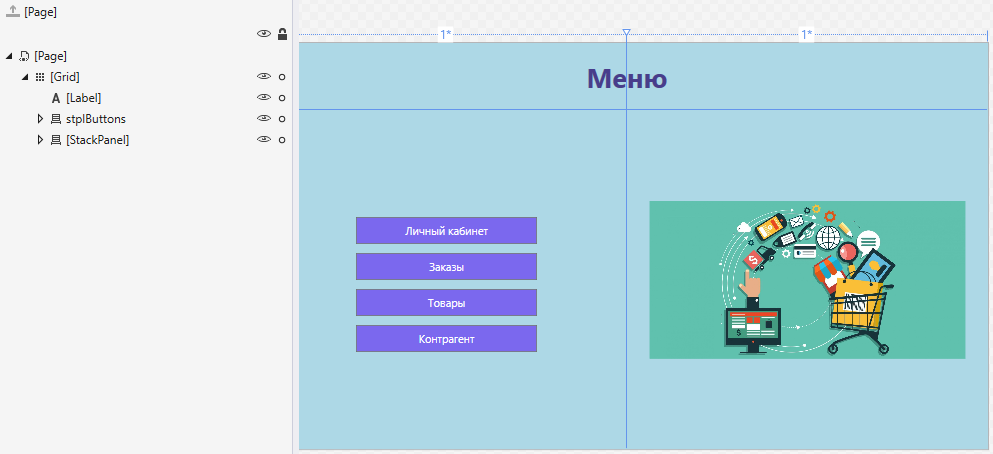


Рис.7 «Меню»

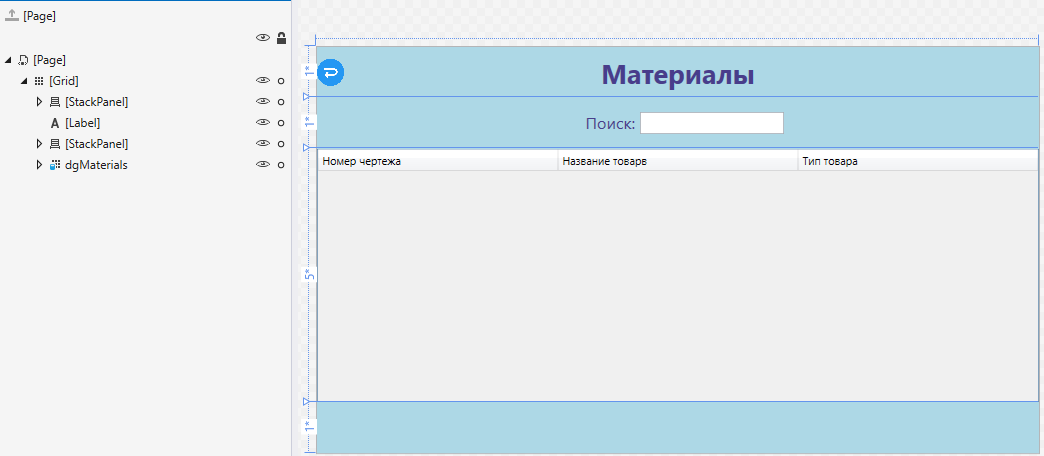


Рис.8 «Материалы»

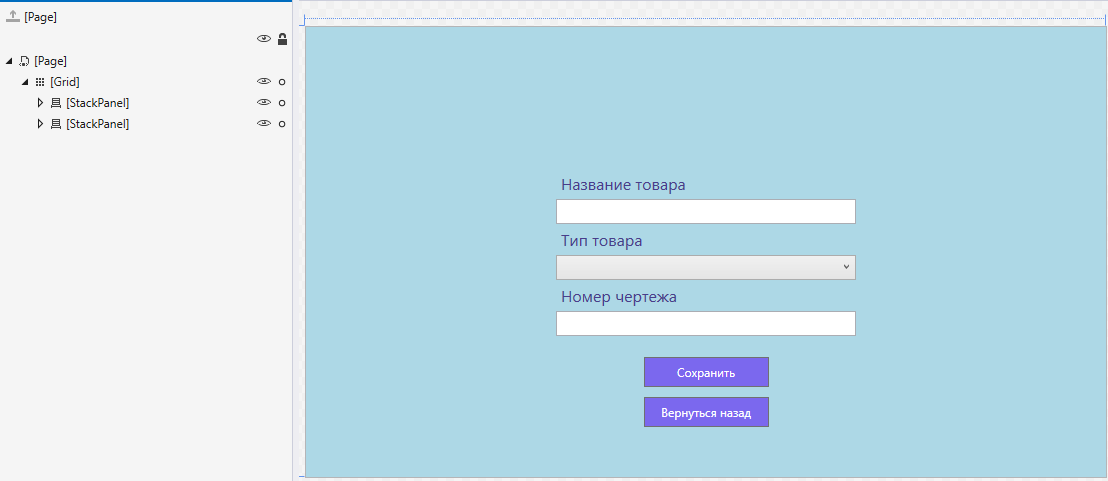


Рис.9 «Добавление материалов»

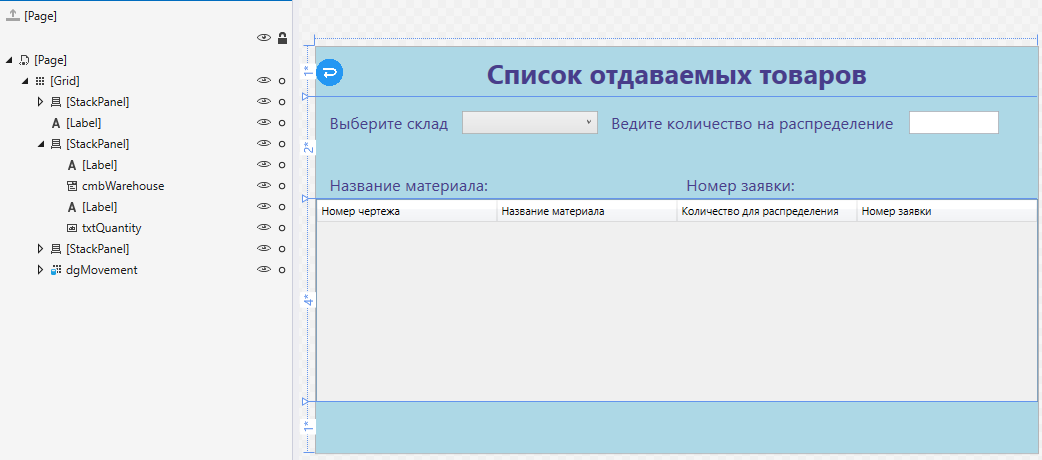


Рис.10 «Список отдаваемых товаров»

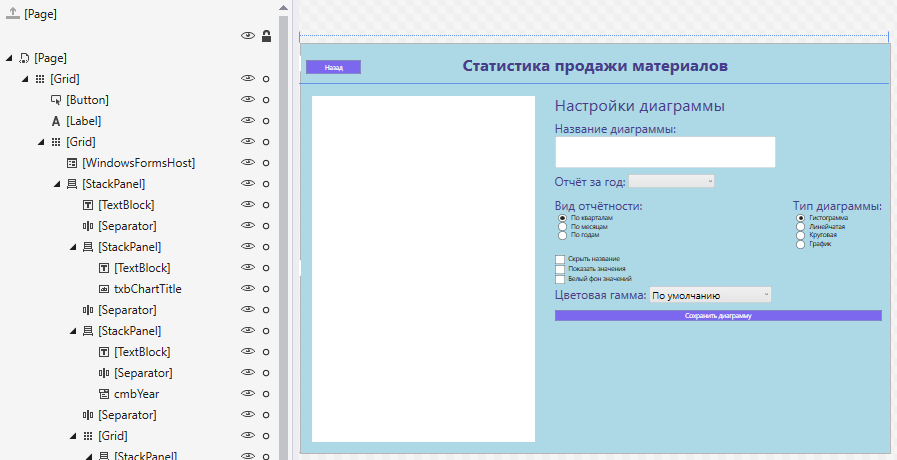


Рис.11 «Статистика продажи материалов

## **ER-диаграмма**

## 

Рис.12 «Диаграмма «Сущность-связь

# **Реализация**

## **Обоснование выбора средств разработки**

Для разработки выбран один из самых мощных инструментов - Visual Studio 2022. Разработчики регулярно сталкиваются с различными вызовами в обработке информации, такими как необходимость в скоростных решениях, основанных на управлении данными, потребность в повышении производительности и мобильности команды разработчиков, а также требования к сокращению бюджета IT-отдела при одновременном расширении инфраструктуры для соответствия растущим требованиям. Microsoft SQL Server 2022 предназначен для решения этих задач. MS SQL Server 2018 - это комплексное решение для управления и анализа данных, которое способствует:

* созданию, развёртыванию и управлению промышленными приложениями, отличающимися повышенной безопасностью, масштабируемостью и надёжностью;
* усилению производительности IT-сектора за счёт уменьшения сложности создания, развёртывания и управления приложениями;
* обмену данными между различными платформами, приложениями и устройствами, облегчая интеграцию внутренних и внешних систем;
* контролю затрат без ущерба для качества, доступности, масштабируемости и безопасности.

Учитывая все эти преимущества, MS SQL Server 2018 выбран в качестве сервера баз данных.

## **Руководство системного программиста**

Основные задачи системного программиста включают разработку, тестирование, отладку и поддержку низкоуровневого программного обеспечения, такого как операционные системы, драйверы устройств и системные утилиты. Вот некоторые из ключевых обязанностей:

* Разработка системного ПО: создание программного обеспечения, которое управляет аппаратными ресурсами и предоставляет общие сервисы для приложений.
* Тестирование и отладка: проверка системного ПО на наличие ошибок и уязвимостей, а также их исправление.
* Поддержка аппаратного обеспечения: разработка и обновление драйверов для нового аппаратного обеспечения и обеспечение совместимости с операционными системами.
* Оптимизация производительности: улучшение эффективности и скорости работы системы путём оптимизации кода и алгоритмов.
* Обеспечение безопасности: реализация мер безопасности для защиты системы от внешних и внутренних угроз.
* Документирование: создание технической документации для системного ПО, которая помогает другим разработчикам и пользователям понять его работу.
* Техническая поддержка: предоставление помощи и консультаций по вопросам, связанным с системным ПО.

Эти задачи требуют глубоких знаний в области компьютерных наук, программирования и аппаратного обеспечения, а также способности решать сложные технические проблемы.

Для проверки работы функции авторизации выполняются следующие шаги (таблица №5).

Таблица №2 «Проверка авторизации»

|  |  |  |
| --- | --- | --- |
| **№** | **Операции по проверке** | **Результат** |
| 1 | **Подготовка к проверке**  Запустите программу «SalesOfMaterials», затем введите логин и пароль на странице авторизации. | Успешный запуск программы |
|  |  |  |
| 2 | **Проверка отображения информации пользователю ПО**  После успешного входа в систему пользователи перенаправляются на страницу «Меню». | Отображение страницы меню |

Для проверки работы функции удаления выполняются следующие шаги (таблица №6).

Таблица №3 «Проверка удаления»

|  |  |  |
| --- | --- | --- |
| **№** | **Операции по проверке** | **Результат** |
| 1 | **Подготовка к проверке**  Кликаем правой кнопкой мыши по одной или нескольким записям в таблице данных и выбираем опцию «Удалить». | Успешный вывод сообщения об удалении данных |
| 2 | **Проверка удаления данных пользователем**  После нажатия на кнопку и появления всплывающего сообщения с подтверждением удаления данных, нажмите кнопку «ОК», чтобы подтвердить своё действие. | Удаление данных из таблиц после закрытия окна |

Для проверки работы функции фильтрации выполняются следующие шаги (таблица №7).

Таблица №4 «Проверка фильтрации»

|  |  |  |
| --- | --- | --- |
| **№** | **Операции по проверке** | **Результат** |
| 1 | **Подготовка к проверке**  Кликаем правой кнопкой мыши по записям в таблице данных и выбираем опцию «Фильтр». | Успешный вывод |
| 2 | **Проверка фильтрации данных пользователем**  После нажатия на кнопку «Фильтр» выбираем нужную опцию и просматриваем таблицу | Вывод данных в таблицу по нужному фильтру |

## **Руководство программиста**

Таблица №5 «Словарь данных»

|  |  |  |  |
| --- | --- | --- | --- |
| **ключ** | **поле** | **обязательное** | **примечание** |
| **Заказы** | | | |
| Первичный | IdExpenseIvoices | Да | Идентификатор |
| Внешний | IdСounterparties | Да | Внешний ключ к таблице контрагентов |
| Внешний | IdEmployee | Да | Внешний ключ к таблице пользователей |
|  | Date\_Transfer | Да | Дата формирования заказа |
|  | DistributedInvoice | Да | Ответ |
| **Состав заказа** | | | |
| Первичный | IdExpenseComposition | Да | Идентификатор |
| Внешний | IdExpenseIvoices | Да | Внешний ключ к таблице заказы |
| Внешний | IdMaterial | Да | Внешний ключ к таблице материалы |
|  | Quantity | Да | Количество |
|  | Price | Да | Цена |
| **Пользователи** | | | |
| Первичный | IdEmployee | Да | Идентификатор |
|  | FIO | Да | ФИО |
|  | Login\_Employee | Да | Логин |
|  | Password\_Employee | Да | Пароль |
| Внешний | IdTypesEmployee | Да | Внешний ключ к таблице типам пользователей |
| **Тип пользователей** | | | |
| Первичный | IdTypesEmployee | Да | Идентификатор |
|  | Nazv\_type | Да | Название типа |
| **Контрагент** | | | |
| Первичный | IdСounterparties | Да | Идентификатор |
|  | Nazv\_Сounterparties | Да | Название контрагента |
|  | INN | Да | ИНН |
|  | Adress\_Сounterparties | Да | Адрес контрагента |
| **Материалы** | | | |
| Первичный | idMaterial | Да | Идентификатор |
|  | Name | Да | Название материала |
|  | DrawingNumber | Да | Чертежный номер |
| Внешний | idType | Да | Внешний ключ к таблице типам материала |
| **Иерархия** | | | |
| Первичный | idHierarchy | Да | Идентификатор |
| Внешний | idParent | Да | Внешний ключ к таблице материалы |
| Внешний | idChild | Да | Внешний ключ к таблице материалы |
|  | Quantity | Да | Количество |
| **Тип материала** | | | |
| Первичный | idType | Да | Идентификатор |
|  | Name | Да | Название типа материала |
| **Склад** | | | |
| Первичный | idWarehouse | Да | Идентификатор |
|  | Name | Да | Название склада |
|  | Address | Да | Адрес |
| **Хранение на складе** | | | |
| Первичный | idStorage | Да | Идентификатор |
| Внешний | idWarehouse | Да | Внешний ключ к таблице склад |
| Внешний | idMaterial | Да | Внешний ключ к таблице материалы |
|  | Quantity | Да | Количество |
| **Сбыт товара** | | | |
| Первичный | idMovement | Да | Идентификатор |
| Внешний | idWarehouse | Да | Внешний ключ к таблице склад |
| Внешний | idComposition | Да | Внешний ключ к таблице состав заказа |
| Внешний | idUser | Да | Внешний ключ к таблице пользователи |
|  | PartOfQuantity | Да | Отданное количество |
|  | ArrivalOrExpenditure | Да | Сбыт |

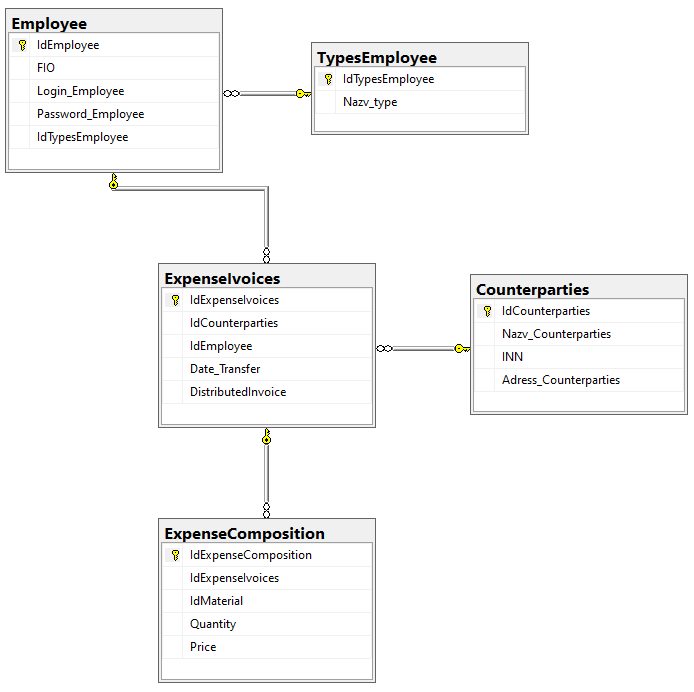


Рис.13 «Диаграмма продажи»

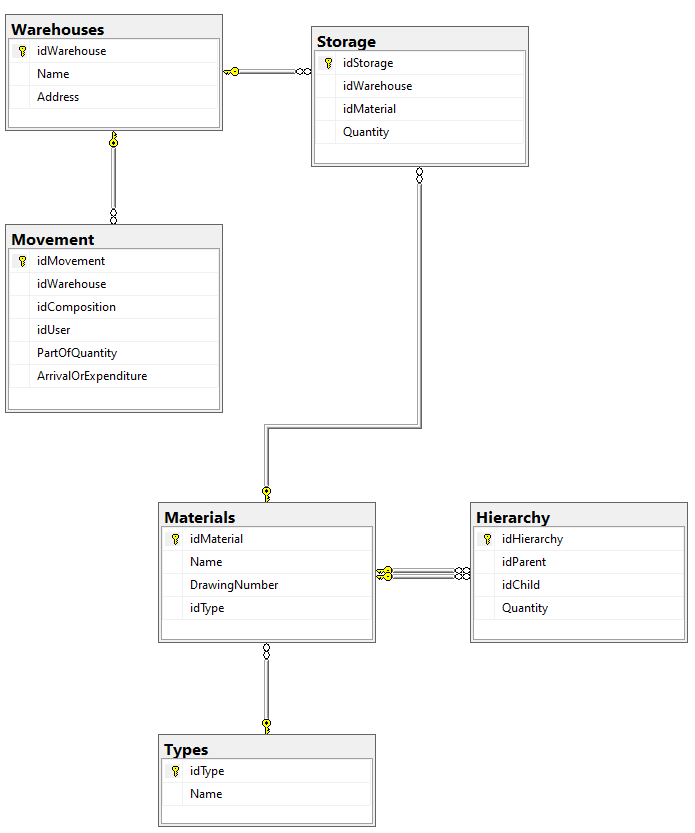


Рис.14 «Диаграмма склада»

## **Руководство пользователя**

При открытии программы ввести логин и пароль, привязанные к аккаунту. Исходя из роли в учётной записи (Администратора, Сотрудника, Кладовщика), осуществится переход на определённую страницу. Администратор появляются все кнопки. Сотрудник видит все кнопки кроме: «Распределения», «Список отдаваемых материалов», «Список отданных материалов», «Список пользователей». Кладовщик видит все кнопки кроме: «Добавить новый заказ», «Список пользователей». В случае, если данные для авторизации введены неверно, появляется уведомление. Если аккаунта ещё нет, то необходима нажать на кнопку «Зарегистрироваться».

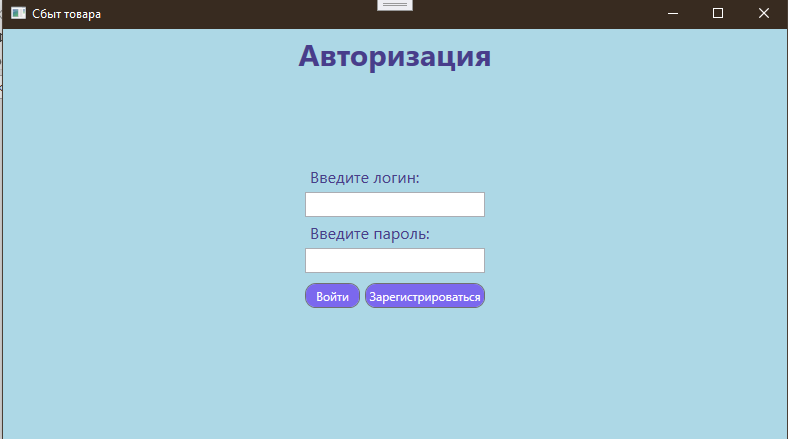


Рис.15 «Страница авторизации»

На странице регистрацию нужно заполнить поля «ФИО», «Логин», «Пароль», «Повторить пароль», «Тип пользователя». После того как все данные в полях заполнены нужно нажать на кнопку «Зарегистрироваться».

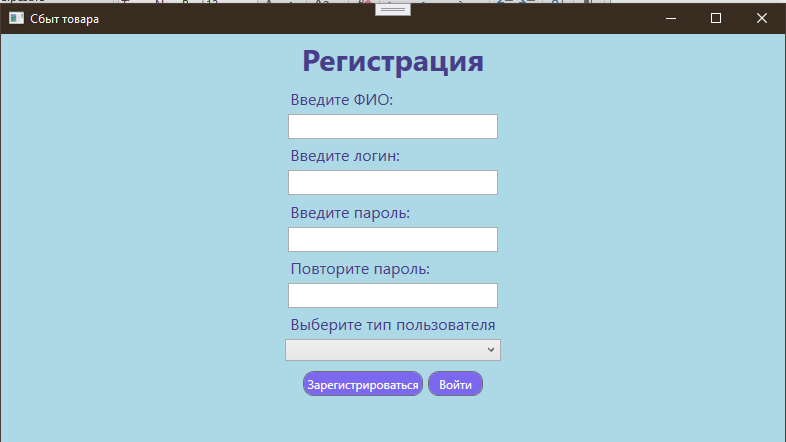


Рис.16 «Страница регистрации»

После авторизации (исполняемый «.exe» файл) появится страница «Меню» с возможностью перехода на страницы «Личный кабинет», «Заказы», «Список товаров на складе», «Товары», «Список отдаваемых материалов», «Список отданных материалов» и «Список пользователей».

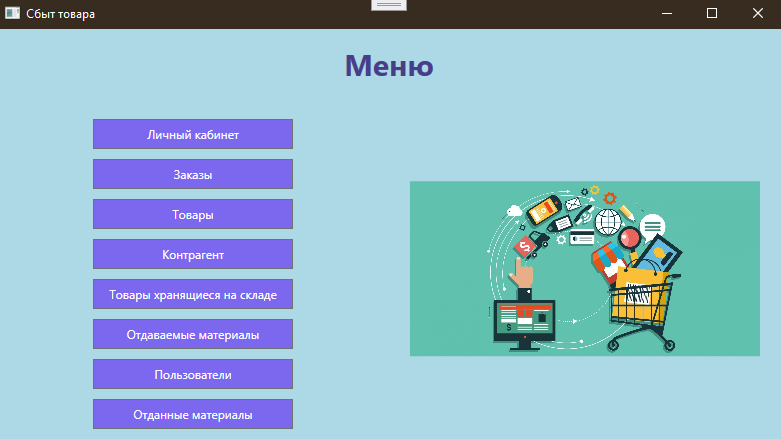


Рис.17 «Главная страница»

При нажатии в навигационном меню на кнопку «Товары» откроется страница с соответствующим списком товаров.

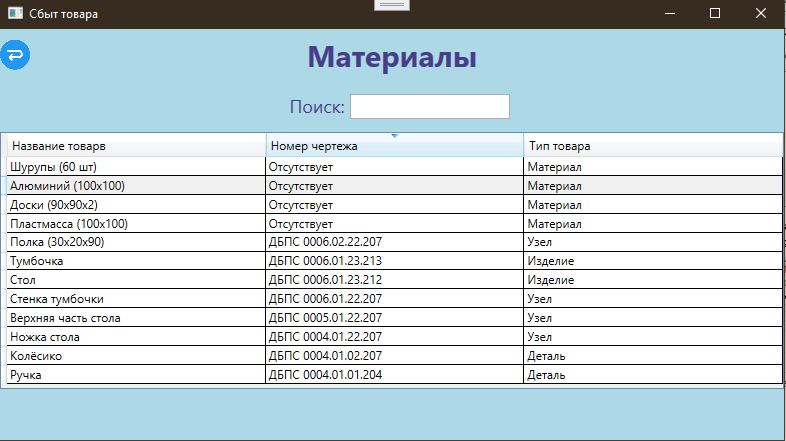


Рис.18 «Страница материалов»

При добавлении информации пользователю необходимо нажать ПКМ по таблице и выбрать из выпадающего списка «Добавить». Затем появится страница для добавления с пустыми значениями, которые потребуется заполнить. Обязательные поля: Название товара, Тип товара, Номер чертежа.

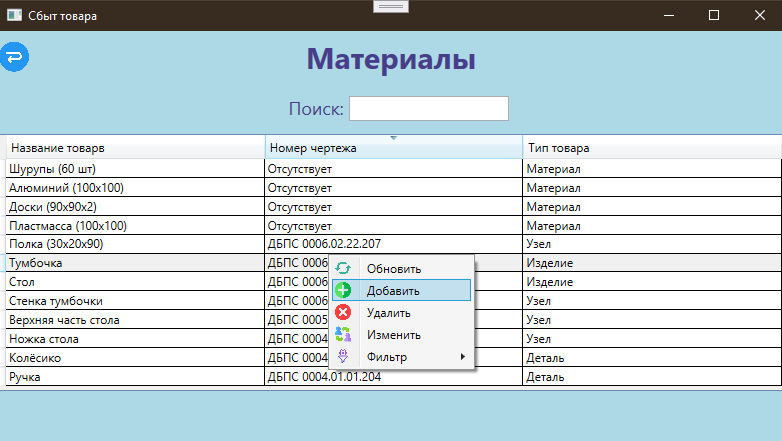


Рис.19 «Страница материалов с вызовом добавления»

Заполним страницу добавления данными.

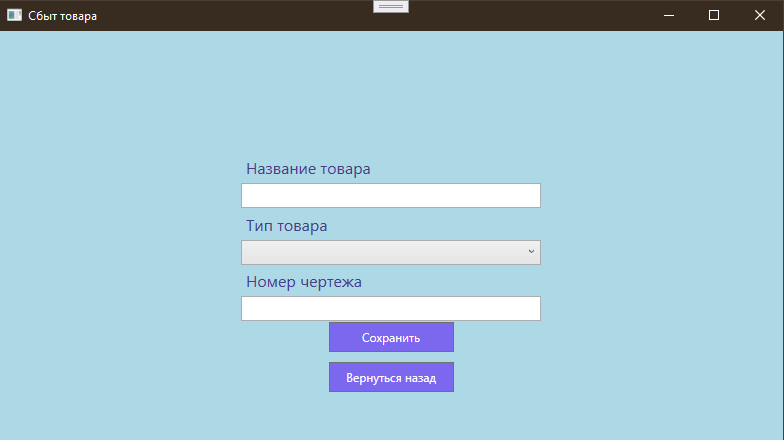


Рис.20 «Страница добавления материалов»

После нажатия кнопки «Сохранить» пользователь возвращается на страницу материалов.

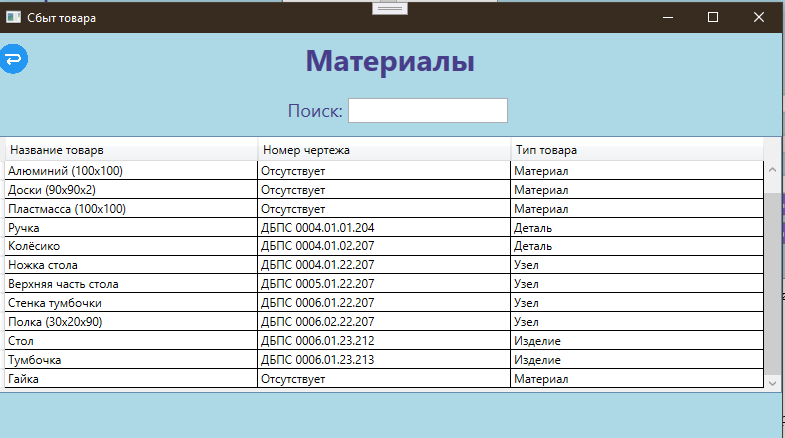


Рис.21 «Страница материалов после добавления»

При изменении информации пользователю необходимо нажать ПКМ по таблице и выбрать из выпадающего списка «Изменить». Затем появится страница для изменения с заполненными значениями, которые потребуется изменить. Обязательные поля: название контрагента, ФИО сотрудника, Дата отправки.

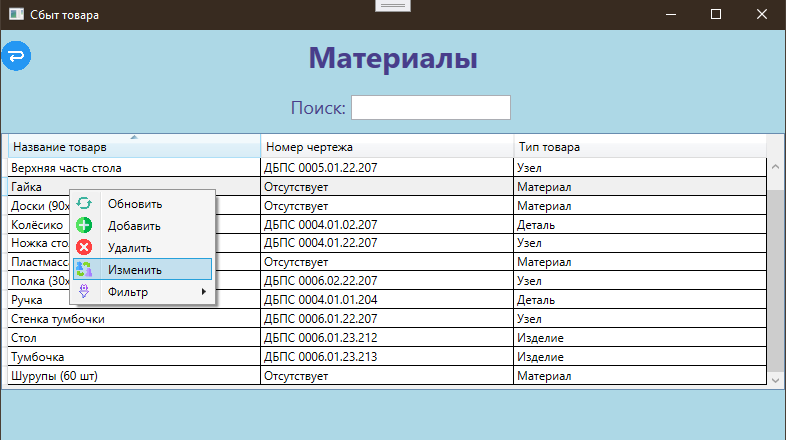


Рис.22 «Страница материалов с вызовом изменения»

Изменим данные на странице редактирование.

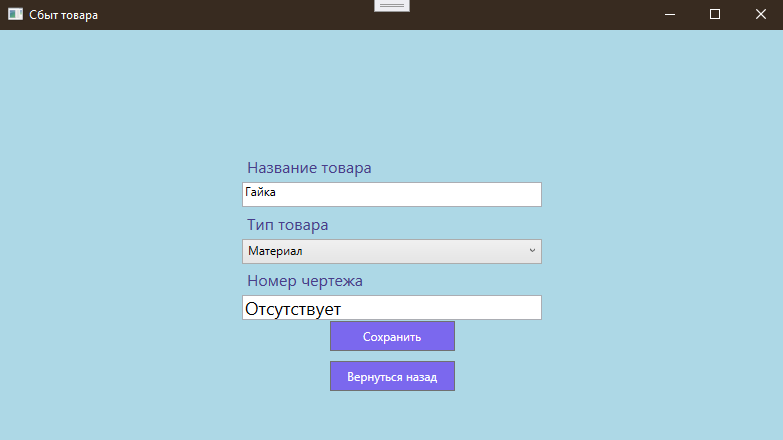


Рис.23 «Страница редактирование материалов»

После нажатия кнопки «Сохранить» пользователь возвращается на страницу материалов.

При удалении информации пользователю необходимо нажать ПКМ и из выпадающего списка выбрать «Удалить». Затем появится уведомление «Удалить заказ».

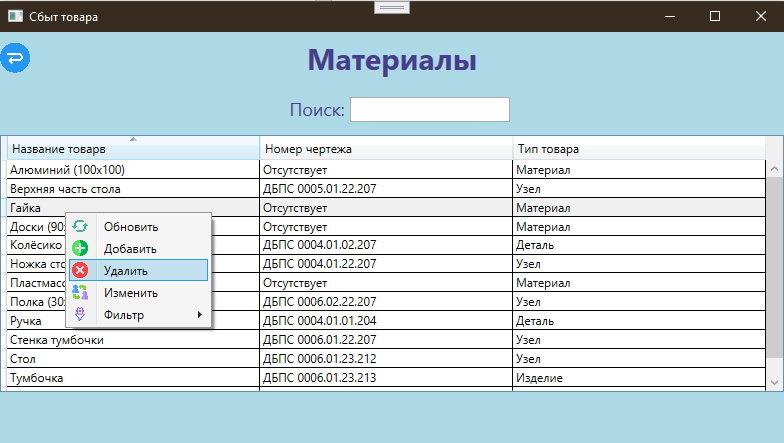


Рис.24 «Страница материалов с удалением записей»

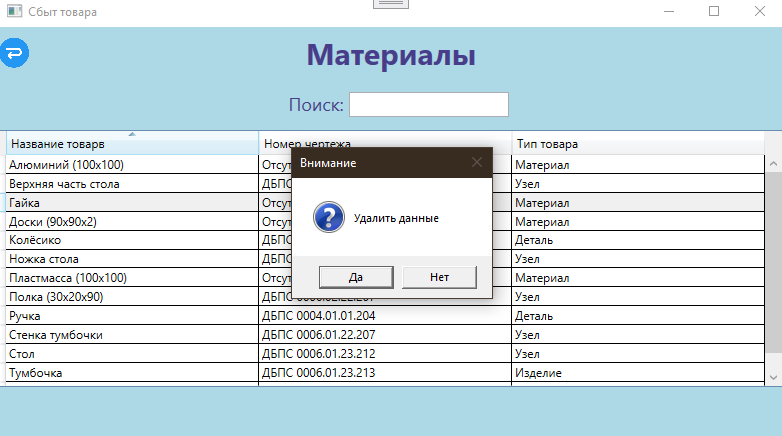


Рис.25 «Страница материалов с уведомление на удаление»

В приложении присутствует фильтрация для нахождения данных по атрибутам таблицы. С помощью ПКМ из списка выбираем из «Фильтрации» нужные нам данные. Если список с нужным фильтром больше не нужен, то нажимаем из этого же списка «Сбросить».

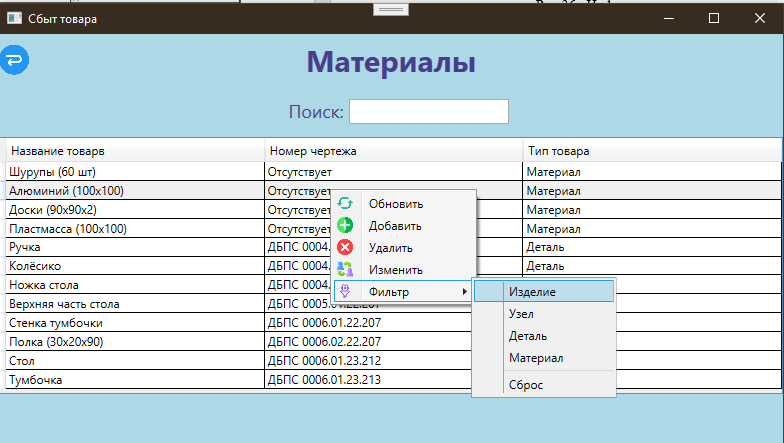


Рис.26 «Страница материалов с фильтром»

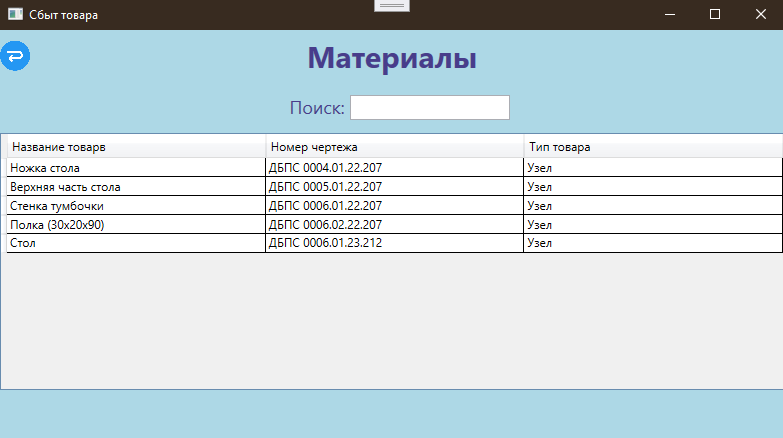


Рис.27 «Страница материалов после фильтрации»

Таблица №6 «Сообщения пользователю»

|  |  |
| --- | --- |
| Попытка добавления нового заказа в список с неуказанными данными | Подтверждение удаления нескольких заказов  . |
| Информация после удаления нескольких пунктов. | Информация после редактирования проекта. |
| Информация после распределения товара. | |

# **Тестирование и отладка**

## **Требования к приложению. Работа тестировщика с требованиями**

Требования к приложению следует изложить в соответствии с общими стандартами качества программного обеспечения:

* 1. Надежность и стабильность: Приложение должно быть стабильным и надежным, без постоянных сбоев и ошибок в работе.
  2. Удобство использования: Приложение должно быть интуитивно понятным и удобным для пользователя. Интерфейс должен быть простым и понятным, чтобы пользователи могли легко ориентироваться в приложении.
  3. Функциональность: Приложение должно обладать всеми необходимыми функциями и возможностями, описанными в спецификации. Все функции приложения должны работать корректно и эффективно.
  4. Производительность: Приложение должно обладать хорошей производительностью и быстрой скоростью работы. Время отклика должно быть минимальным, чтобы предотвратить задержки и зависания.
  5. Безопасность: Приложение должно обеспечивать защиту данных пользователей и обладать адекватными механизмами безопасности, чтобы защитить информацию от несанкционированного доступа.
  6. Поддержка: Приложение должно быть легко развертываемым и обновляемым. Разработчики должны предоставить поддержку и обновления для приложения, чтобы исправлять ошибки и улучшать его функциональность.

Работа тестировщика с требованиями начинается на этапе анализа и изучения требований к программе или системе. Важно уделить особое внимание следующим аспектам:  
1. Понимание требований: тестировщик должен тщательно изучить и понять все требования к разрабатываемой системе, чтобы иметь ясное представление о том, как должна работать система.

2. Проверка требований на корректность и полноту: тестировщик должен убедиться, что все требования являются корректными, не противоречат друг другу и полными.

3. Формирование тестовых сценариев: на основе изученных требований тестировщик составляет тестовые сценарии и тест-кейсы для проверки соответствия системы требованиям.

4. Планирование тестирования: тестировщик определяет стратегию и план тестирования, учитывая все выявленные требования к системе.

5. Выполнение тестов: согласно разработанному плану тестирования тестировщик проводит тесты, проверяя работу системы на соответствие требованиям.

6. Отчетность и отслеживание ошибок: при обнаружении несоответствий или ошибок в работе системы, тестировщик документирует их и передает ответственным лицам для исправления.

7. Регулярное обновление тестов: при изменении требований к системе необходимо вносить коррективы в тестовые сценарии и повторно проводить тестирование.  
Таким образом, работа тестировщика с требованиями - это важный этап в жизненном цикле разработки программного обеспечения, который позволяет обеспечить соответствие разработанной системы заданным требованиям и ее качество.

## **Отладка приложения**

Участок кода с добавление материала в таблицу, работающий некорректно:

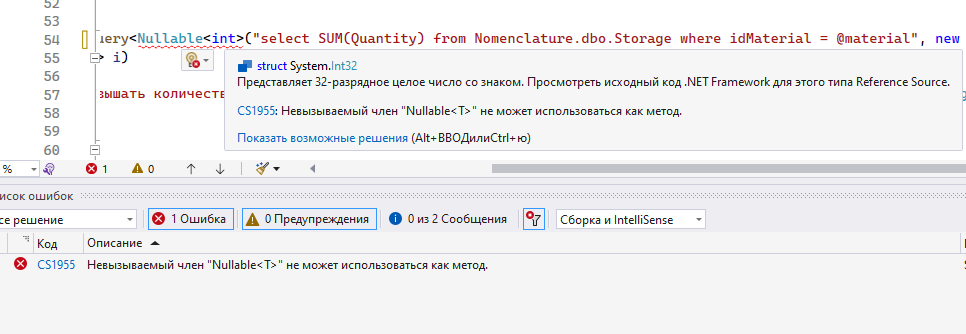


Рис.28 «Неявное преобразование типов»

Исправленный участок кода с добавление материала в таблицу, работающий корректно:



Рис.29 «Переделанный код для метода»

Участок кода с переходом на страницу добавления заказа, работающий некорректно:

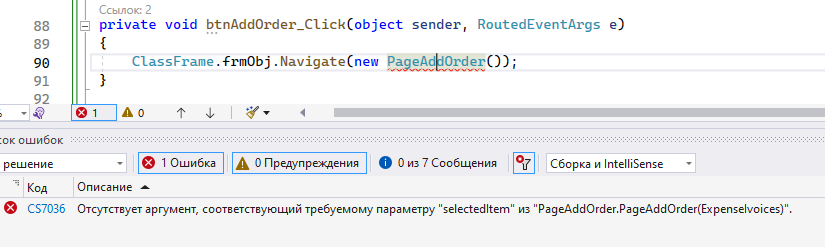


Рис.30 «Отсутствие аргумента, требуемой страницы»

Участок кода с переходом на страницу добавления заказа, работающий корректно:

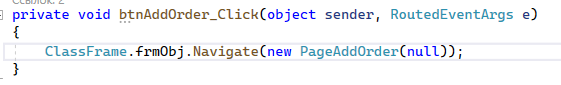


Рис.31 «Добавление пустого аргумента, требуемой странице»

**5. Расчет базовой себестоимости разрабатываемого продукта**

**5.1 Исходные данные**

Затраты времени и занятость специалистов соответствуют трудоемкости и сложности этапов создания приложения представлены в таблице № 7.

Таблица №7 «Исходные данные»

|  |  |
| --- | --- |
| **Этапы разработки программного продукта** | **Величина затрат рабочего времени, часы** |
| Анализ поставленной задачи | 5 |
| Разработка и проектирование БД | 15 |
| Проектирование интерфейса | 10 |
| Разработка интерфейса | 30 |
| Программирование | 60 |
| Оптимизация и рефакторинг | 30 |
| Тестирование | 35 |
| Отладка | 20 |
| Итого | 205 |

* 1. **Расчёты затрат на выполнение программы**

Создание компьютерной программы связанно со значительными затратами времени и соответственно большим удельным весом в себестоимости таких затратах, как оплата труда.

В работе по проектированию и разработке программы участвовал 1 специалист – разработчик с окладом – 20000 руб.

Расчет оплаты специалистов производиться исходя из дневной тарифной ставки.

Тарифная ставка — в трудовом праве фиксированный размер оплаты: труда работника за выполнение нормы труда определённой сложности за единицу времени.

Фонд оплаты труда- суммарные денежные средства, израсходованные в течение определённого периода времени на заработную плату, включая: стимулирующие оплаты и дополнительную оплату.

Расчёт чистых затрат на выполнение программы рассчитывается из затрат на заработную плату разработчику, умноженную на количество часов разработки и суммирование с прочими затратами на разработку, такие как электричество.

Затраты на заработную плату разработчику = 20000/(21\*8)\*205= 24404,76 руб.

* 1. **Расчет отчислений на социальное страхование и обеспечение**

Обычный размер ставки - для наемного работника - составляет 30 %. Пенсионный фонд Российской Федерации - 22 %

Фонд социального страхования - 2,9 %

Фонды обязательного медицинского страхования - 5,1 %

Итого: 30 %

Таблица №8 «Расчет отчислений на социальное страхование и обеспечение»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Должность работника** | **Пенсионный фонд РФ – ПФР (22%), руб.** | **Фонд социального страхования РФ –ФССРФ (2,9%), руб.** | **ФОМС (5,1%), руб.** | **Итого отчислений на социальное страхование и обеспечение (ОСО), руб.** |
| 1 | Разработчик | 5369,05 | 707,74 | 1244,64 | 7321,43 |

Формулы расчета для таблицы №6:

ПФР = ОТ \* 22% = 24404,76 руб. \* 22% = 5369,05 руб.

ФСС РФ = ОТ \* 2,9% = 24404,76 руб. \* 2,9% = 707,74 руб.

ФОМС = ОТ \* 5,1% = 24404,76 руб. \* 5,1% = 1244,64 руб.

ОСО = ПФР + ФССРФ + ФОМС = 5369,05руб. + 707,74 руб. + 1244,64 руб. = 7321,43руб.

* 1. **Расчет базовой себестоимости компьютерного продукта**

Себестоимость — это текущие затраты организации на производство и реализацию продукции, выраженные в денежной форме.

Себестоимость продукции является качественным показателем, так как она характеризует уровень использования всех ресурсов, находящихся в распоряжении организации.

Косвенные расходы — это затраты, связанные с производством и реализацией продукции (работ, услуг), которые нельзя напрямую отнести к изготовлению продукции (работ, услуг) и можно учесть в расходах в том периоде, когда они понесены.

К косвенным расходам можно отнести, например, затраты на услуги связи, аренду офиса и т.д.

Самое главное отличие прямых расходов от косвенных расходов в том, что сумма косвенных расходов в полном объеме относится к расходам текущего отчетного (налогового) периода, а прямых — к расходам текущего периода по мере реализации товаров, работ, то есть с учетом остатков незавершенного производства.

Исключением являются случаи, когда деятельность организации связана с оказанием услуг. Данные налогоплательщики вправе относить сумму прямых расходов отчетного (налогового) периода в полном объеме на уменьшение доходов от производства и реализации данного отчетного (налогового) периода без распределения на остатки незавершенного производства.

Относить ли расходы к прямым или косвенным расходам компания определяет отдельно для каждого производственного цикла.

Если те или иные ресурсы согласно технологическим регламентам не включены в производственный цикл, не являются его неотъемлемой частью, то затраты на них можно учитывать в составе косвенных расходов.

**Перечень косвенных расходов**

Косвенные расходы — это расходы, которые нельзя прямо отнести на конкретные виды продукции (работ, услуг).

К косвенным расходам относятся:

административно - управленческие расходы;

расходы на отопление и освещение помещений;

расходы на страхование;

расходы на содержание общехозяйственного персонала;

амортизационные отчисления и расходы на ремонт основных средств управленческого и общехозяйственного назначения;

арендная плата за помещения общехозяйственного назначения;

расходы по оплате информационных, аудиторских, консультационных и т. п. услуг;

расходы, связанные со сбытом продукции:

другие аналогичные по назначению управленческие расходы.

Таблица №9 «Расчёт базовой себестоимости»

|  |  |  |
| --- | --- | --- |
| **№** | **Статьи затрат** | **Сумма в рублях** |
| 1 | Фонд оплаты труда (ФОТ) | 31726,19 |
| 2 | Косвенные расходы (КР) 10% от ФОТ | 3172,62 |
| 3 | Расходы на продажу (РП) (ФОТ + КР) \* 10% | 3489,88 |
| 4 | Полная себестоимость (ПС) | 38388,69 |

Формулы расчета для таблицы №6:

ФОТ = ОТ + ОСО = 24404,76 руб. + 7321,43 руб. = 31726,19 руб.

В данной организации косвенные расходы составляют 10% от ФОТ.

КР = ФОТ \*10% = = 31726,19 руб. \* 10% = 3172,62 руб.

РП = (ФОТ + КР) \* 10% = (31726,19 руб. + 3172,62 руб.) \* 10% =3489,88 руб.

ПС = ФОТ + КР + РП = 31726,19 руб. + 3172,62 руб. + 3489,88 руб. = 38388,69 руб.

Итоговая стоимость разработки приложения для клиента организации ООО «ЦА Максималист» по продажи со склада материалов составит 38388,69 руб.

# **Заключение**

В процессе подготовки дипломной работы было проведено всестороннее изучение бизнес-процессов в ООО «ЦА «Максималист». Исследование показало, что внедрение система по сбыту товара позволит автоматизировать взаимодействие менеджеров с клиентами, что значительно повысит эффективность их работы. Современное программное обеспечение для учета товаров на складе облегчает выполнение различных операций. Предустановленные шаблоны документов, созданные программистами, позволяют сократить время, затрачиваемое сотрудниками склада на ручное заполнение бумаги при перемещении товаров.

В результате разработки программного продукта «SalesOfMaterials» была создана база данных на SQL Server, приложение на Visual Studio.

Приложение «SalesOfMaterials» предоставляет администратору, менеджеру и кладовщику ряд преимуществ, способствующих повышению эффективности их работы.

Централизованное управление заказами: программное обеспечение предоставляет одну платформу для формирования, обработки заявок, что дает возможность администратору эффективно координировать рабочие процессы, распределять задачи между исполнителями и контролировать выполнение работ в заданные сроки.

Улучшение взаимодействия с клиентами: программное обеспечение «SalesOfMaterials» позволяет создавать профили пользователей, управлять приоритетами задач. Каждый профиль пользователя содержит информацию о его роли и правах доступа к функционалу программы, что позволяет оптимизировать рабочий процесс и повысить эффективность работы команды.

Фильтрация и поиск данных: возможность применения фильтров по товару облегчает процесс поиска нужной информации пользователями, сокращая затраченное время на поиск товаров.

Анализ производительности и отчетность: предоставление статистических данных в виде диаграмм о продажах дает полезную информацию о развитии компании, а создание договора о продаже товаров в Excel помогает формализовать сделку.

Прозрачность и контроль: возможность распечатать договор и диаграмму позволяет структурировать и зафиксировать все процессы и взаимодействия, улучшая прозрачность и контроль со стороны управления.

В ходе выполнения работ было проведено тестирование и отладка приложения, в результате чего были устранены недочеты в отображении элементов интерфейса, логике функционирования приложения и синтаксические ошибки.

Оценка базовой стоимости создания программного продукта проводилась на основе объема работ и тарифа за час труда разработчика. Расчеты выплат, связанных с социальным страхованием и предоставлением гарантий работникам, также включены в анализ. Учитывая все непрямые затраты, определена общая стоимость разработки программного продукта для компании.

Приложение «SalesOfMaterials» представляет собой корпоративную систему, обеспечивающую эффективное управление рабочими процессами в области клиентского сервиса. Этот инструмент улучшает рабочую эффективность и способствует более широкому распространению сведений о компании.

# **Список литературы**

1. ГОСТ 34.602-89 Автоматизированные системы управления. Техническое задание на создание автоматизированной системы.
2. ГОСТ 19.503-79 Единая система программной документации. Технический проект.
3. ГОСТ 19.504-79 Единая система программной документации. Рабочая документация.
4. ГОСТ 19.505-79 Единая система программной документации. Текст программы.
5. ГОСТ Р ИСО/МЭК 12207-2010 Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программного обеспечения.
6. ГОСТ Р 51232-2019 Информационные технологии. Технологический процесс разработки и сопровождения программного обеспечения
7. Адам, Фримен ASP.NET 4.5 с примерами на C# 5.0 для профессионалов / Фримен Адам. - М.: Диалектика / Вильямс, 2021. - 2792 c.
8. Вагнер, Билл С# Эффективное программирование / Билл Вагнер. - М.: ЛОРИ, 2021. - 320 c.
9. Ватсон, Б. С# 4.0 на примерах (C# 4.0. How-To) / Б. Ватсон. - М.: БХВ-Петербург, 2021. - 608 c.
10. Албахари, Джозеф C# 3.0. Справочник / Джозеф Албахари, Бен Албахари. - М.: БХВ-Петербург, 2021. - 944 c.
11. Биллиг, В. А. Основы программирования на С# / В.А. Биллиг. - М.: Интернет-университет информационных технологий, Бином. Лаборатория знаний, 2021. - 488 c.
12. Зверева В.П. Сопровождение и обслуживание программного обеспечения компьютерных систем.: учебник для студ. учреждений сред. проф. образования /В.П. Зверева, А.В. Назаров. –2-е изд., испр. – Москва: Академия, 2020. – 256 с. – (Профессиональное образование.)
13. Овечкин Г.В. Компьютерное моделирование: учебник для учреждений СПО /Г.В. Овечкин, П.В. Овечкин - М.: Издательский центр «Академия»,2020.- 205 с. -(Профессиональное образование)
14. Перлова О.Н. Проектирование и разработка информационных систем: учебник для студентов СПО /О.Н. Перлова, О.П. Ляпина, А.В. Гусева.-3-е изд., испр. - Москва: Академия, 2020.- 252 с. – (Профессиональное образование).
15. Федорова Г.Н. Разработка, администрирование и защита баз данных: учебник для среднего профессионального образования / Федорова Г.Н — Москва: Издательство Академия, 2020. — 288 с. — (Профессиональное образование).

# **Приложения**

Код страницы «Список отдаваемых товаров»

***DistributionList.xaml.cs:***

public partial class DistributionList : Page

{

public ExpenseComposition \_currentItem = new ExpenseComposition();

public DistributionList()

{

InitializeComponent();

dgMovement.ItemsSource = ClassFrame.db.ExpenseComposition.ToList();

cmbWarehouse.ItemsSource = ClassFrame.db.Database.SqlQuery<Warehouses>("select \* from Nomenclature.dbo.Warehouses").ToList();

}

private void dgMovement\_MouseDoubleClick(object sender, MouseButtonEventArgs e)

{

\_currentItem = dgMovement.SelectedItem as ExpenseComposition;

if (\_currentItem != null)

{

foreach (Warehouses w in cmbWarehouse.ItemsSource)

{

int i = ClassFrame.db.Database.SqlQuery<int?>("select Quantity from Nomenclature.dbo.Storage where idWarehouse = @idW and idMaterial = @idM", new SqlParameter("@idW", w.idWarehouse), new SqlParameter("@idM", \_currentItem.IdMaterial)).FirstOrDefault() == null

? 0 : ClassFrame.db.Database.SqlQuery<int>("select Quantity from Nomenclature.dbo.Storage where idWarehouse = @idW and idMaterial = @idM", new SqlParameter("@idW", w.idWarehouse), new SqlParameter("@idM", \_currentItem.IdMaterial)).FirstOrDefault();

if (i > 0)

{

txtMaterials.Text = \_currentItem.Material.Name;

txtApplication.Text = Convert.ToString(\_currentItem.IdExpenseIvoices);

cmbWarehouse.SelectedItem = w;

MessageBox.Show($"Количество материала на складе: {i}", "Уведомление", MessageBoxButton.OK, MessageBoxImage.Information);

return;

}

}

MessageBox.Show("Такого материала нет на складах.", "Уведомление", MessageBoxButton.OK, MessageBoxImage.Information);

cmbWarehouse.SelectedIndex = -1;

}

}

private void btnBack\_Click(object sender, RoutedEventArgs e)

{

ClassFrame.frmObj.Navigate(new PageMenu());

}

private void MenuDistribute\_Click(object sender, RoutedEventArgs e)

{

if (string.IsNullOrEmpty(txtQuantity.Text))

{

MessageBox.Show("Поле количество на раcпределение пустое", "Ошибка ввода", MessageBoxButton.OK, MessageBoxImage.Error);

return;

}

if (Convert.ToInt32(txtQuantity.Text) > \_currentItem.DifferenceQuantity)

{

MessageBox.Show("Количество на раcпределение не может быть больше количества для распределения", "Ошибка ввода", MessageBoxButton.OK, MessageBoxImage.Error);

return;

}

Warehouses w = cmbWarehouse.SelectedItem as Warehouses;

try

{

ClassFrame.db.Database.ExecuteSqlCommand("insert into Nomenclature.dbo.Movement values (@warehouse,@composition,@user,@quantity,1)", new SqlParameter("@warehouse", w.idWarehouse), new SqlParameter("@composition", \_currentItem.IdExpenseComposition), new SqlParameter("@user", Users.GetUsers.ID), new SqlParameter("@quantity", Convert.ToInt32(txtQuantity.Text)));

ClassFrame.db.Database.ExecuteSqlCommand("update Nomenclature.dbo.Storage set Quantity = Storage.Quantity - @quantity where idMaterial = @material and idWarehouse = @warehouse", new SqlParameter("@quantity", Convert.ToInt32(txtQuantity.Text)), new SqlParameter("@material", \_currentItem.IdMaterial), new SqlParameter("@warehouse", w.idWarehouse));

ClassFrame.db.Database.ExecuteSqlCommand("delete Nomenclature.dbo.Storage where idMaterial = @material and Quantity = 0 and idWarehouse = @warehouse", new SqlParameter("@material", \_currentItem.IdMaterial), new SqlParameter("@warehouse", w.idWarehouse));

dgMovement.ItemsSource = ClassFrame.db.Database.ExecuteSqlCommand("d");

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

if (\_currentItem.DifferenceQuantity != 0)

{

cmbWarehouse.SelectedIndex++;

w = cmbWarehouse.SelectedItem as Warehouses;

int i = ClassFrame.db.Database.SqlQuery<int?>("select Quantity from Nomenclature.dbo.Storage where idWarehouse = @idW and idMaterial = @idM", new SqlParameter("@idW", w.idWarehouse), new SqlParameter("@idM", \_currentItem.IdMaterial)).FirstOrDefault() == null

? 0 : ClassFrame.db.Database.SqlQuery<int>("select Quantity from Nomenclature.dbo.Storage where idWarehouse = @idW and idMaterial = @idM", new SqlParameter("@idW", w.idWarehouse), new SqlParameter("@idM", \_currentItem.IdMaterial)).FirstOrDefault();

MessageBox.Show($"Количество материала на складе: {i}", "Уведомление", MessageBoxButton.OK, MessageBoxImage.Information);

}

else

{

MessageBox.Show("Успешно распределенно");

\_currentItem = null;

}

dgMovement.ItemsSource = ClassFrame.db.ExpenseComposition.ToList();

txtQuantity.Text = "";

}

private void MenuСheck\_Click(object sender, RoutedEventArgs e)

{

if (\_currentItem == null)

{

MessageBox.Show($"Позиция не выбрана.", "Уведомление", MessageBoxButton.OK);

return;

}

List<ExpenseComposition> list = ClassFrame.db.ExpenseComposition.Where(x => x.IdExpenseIvoices == \_currentItem.IdExpenseIvoices).ToList();

foreach (ExpenseComposition expense in list)

{

int? d = ClassFrame.db.Database.SqlQuery<int?>($"select sum(PartOfQuantity) from Nomenclature.dbo.Movement where idComposition = {expense.IdExpenseComposition} and ArrivalOrExpenditure = 1").Single();

if (d == null) d = 0;

if (d != expense.Quantity)

{

MessageBox.Show($"Товар \"{expense.Material.Name}\" не закрыт.", "Уведомление", MessageBoxButton.OK);

return;

}

}

try

{

ClassFrame.db.ExpenseIvoices.FirstOrDefault(x => x.IdExpenseIvoices == \_currentItem.IdExpenseIvoices).DistributedInvoice = true;

ClassFrame.db.SaveChanges();

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

MessageBox.Show("Заказ закрыт", "Уведомление", MessageBoxButton.OK, MessageBoxImage.Information);

ClassFrame.frmObj.Navigate(new PageOrder());

}

private void MenuUpdate\_Click(object sender, RoutedEventArgs e)

{

dgMovement.ItemsSource = ClassFrame.db.ExpenseComposition.ToList();

cmbWarehouse.ItemsSource = ClassFrame.db.Database.SqlQuery<Warehouses>("select \* from Nomenclature.dbo.Warehouses").ToList();

}

private void Back\_Click(object sender, RoutedEventArgs e)

{

ClassFrame.frmObj.Navigate(new PageMenu());

}

private void Back1\_Click(object sender, RoutedEventArgs e)

{

ClassFrame.frmObj.Navigate(new PageMenu());

}

}