Министерство образования Московской области

ГБПОУ МО «Колледж «Коломна»

09.02.07

К защите допускается

зам. директора по УР

\_\_\_\_\_\_\_Э.Б. Ромашкина

«\_\_\_\_»\_\_\_\_\_\_\_ 20\_\_\_г.

ДИПЛОМНЫЙ ПРОЕКТ

Тема: Разработка информационной системы для пекарни

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

ККОО.ДП2018.000

|  |  |  |
| --- | --- | --- |
| Студент | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2024 г. | Жабин Н.С. |
| Руководитель проекта | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2024 г. | Карташова Е. В. |
| Консультант по экономическому разделу | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2024 г. | Караваев А.В. |
| Нормоконтролер | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2024 г. | Грушникова Т.Н. |
| Рецензент | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2024 г. | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

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

2024

Министерство образования Московской области

Государственное бюджетное профессиональное образовательное учреждение Московской области «Колледж «Коломна»

|  |  |
| --- | --- |
|  | «Утверждаю»  Зам. директора по УР  \_\_\_\_\_\_\_\_\_\_ Э.Б. Ромашкина  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2024 г. |

ЗАДАНИЕ

на дипломный проект по специальности 09.02.07 «Информационные системы и программирование» студенту очного отделения, группы 42012

Жабину Никите Сергеевичу

Тема проекта: Разработка информационной системы для пекарни

1. Постановка задачи:

Разработать информационную систему для хранения данных о клиентах и товарах и заказа различной продукции в пекарне.

Информационная система должны решать следующие задачи:

* Авторизация пользователя;
* Хранение информации о клиентах и заказах;
* Возможность оперативного добавления и изменения данных;
* Возможность оформления новых заказов;
* Возможность сохранения заказа в файле pdf;

В реализованном приложении должны присутствовать следующие формы для просмотра необходимой информации пользователем:

* Информация о заказах продукции;
* Информация о продукции;
* Информация о клиентах;
* Информация о суммах заключенных договоров.

*Для возможности просмотра информации о заказах необходимы поля: номер, номер клиента, номер менеджера, стоимость, статус дата формирования, дата выполнения.*

*Для возможности просмотра информации о клиентах необходимы поля: номер, наименование, ИНН, адрес, телефон, email.*

*Для возможности просмотра информации о продукции необходимы поля: номер, название, номер типа товара, описание, вес, цена.*

2. Состав и объем дипломного проекта

2.1 Пояснительная записка, содержащая следующие обязательные разделы:

* Введение
* Основная часть
* Экономический раздел
* Требования по технике безопасности при работе с вычислительной техникой
* Выводы и заключение
* Список литературы
* Приложения

2.2 Презентация к докладу.

3. Требования к работе:

*Пояснительная записка должна содержать не менее 50 листов формата A4. Шрифт оформления пояснительной записки Times New Roman, 14 пт, 1,5 междустрочный интервал.*

*Презентация должна содержать не более 15 слайдов. На первом слайде указывается название учебного заведения, тема дипломного проекта, фамилия, имя и отчество студента и руководителя, название специальности, город и год. Фон презентации должен оформляться в светлых, нейтральных тонах с контрастным черным текстом. Размер шрифта – не менее 28. Заголовки нужно выделить, шрифт заголовков – не менее 36.*

4. Содержание экономического раздела:

*Экономический раздел включает следующие пункты:*

* *Понятие себестоимости;*
* *Затраты в составе себестоимости;*
* *Расчет себестоимости информационной системы.*

5. Содержание раздела по технике безопасности:

*Требования по технике безопасности при работе с вычислительной техникой*

Рекомендованная литература:

1. *Подбельский В.В. Базовый курс C#: учебник для среднего профессионального образования/ В.В. Подбельский.-М.: Издательство Юрайт,2022*
2. *Кудрина, Е.В. Основы алгоритмизации и программирования на языке С#: учеб. пособие для среднего профессионального образования / Е.В. Кудрина, М.В. Огнева. – М.: Издательство Юрайт, 2022.*
3. *Виноградова Н.А. Научно-исследовательская работа студента: Технология написания и оформления доклада, реферата, курсовой и выпускной квалификационной работы: учеб. пособие для студ. Учреждений сред. Проф. Образования / Н.А. Виноградова, Н.В. Микляева. – 13-е изд., стер. – М.: Издательский центр «Академия», 2023.*
4. *Федорова Г.Н. Разработка модулей программного обеспечения для компьютерных систем: учебник для студ. Учреждений сред. Проф. Образования / Г.Н. Федорова. – 3-е изд., исправленное. – М.: Издательский центр «Академия», 2023.*
5. *Липкович И.Э Производственная безопасность, Лабораторный практикум, Петренко Н.В., Егорова И.В., Пятикопов С.М., 2023*

Дата выдачи задания «05» апреля 2024 г.

Срок окончания дипломного проекта «13» июня 2024 г.

**Руководитель проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Руководитель структурного подразделения ­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Консультант по экономическому разделу \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Задание утверждено на заседании цикловой комиссии

Протокол № 7 от «11» марта 2024 г.

**Председатель цикловой комиссии \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2024 года.

Содержание

Введение

[1 Основная часть 8](#_Toc169766234)

[1.1 Постановка задачи](#_Toc169766235)

[1.2 Описание использованных технологий](#_Toc169766236)

[1.3 Проектирование базы данных 11](#_Toc169766237)

[1.4 Проектирование приложения 15](#_Toc169766238)

[1.5 Разработка приложения 18](#_Toc169766239)

[2 Экономический раздел 27](#_Toc169766240)

[2.1 Понятие себестоимости](#_Toc169766241)

[2.2 Затраты в составе себестоимости 28](#_Toc169766242)

[2.3 Расчет себестоимости информационной системы 31](#_Toc169766243)

[3 Техника бещопасности при работа с вычислительной техникой 39](#_Toc169766244)

[Выводы и заключение 42](#_Toc169766245)

[Список литературы 43](#_Toc169766246)

Приложение А – Код программы  [44](#_Toc169766247)

Приложение Б [– Флэш - накопитель с программой 52](#_Toc169766248)

В стремительно развивающемся информационном мире, где технологии неуклонно проникают во все сферы жизни, люди все чаще обращаются к цифровым решениям, отказываясь от традиционных бумажных носителей. Это связано с тем, что бумага является не самым надежным способом хранения информации. Она подвержена рискам потери, повреждения или даже кражи. В этом смысле цифровые системы представляют собой значительно более безопасную и удобную альтернативу.

В некоторых организациях все данные о клиентах и их заказах хранятся на бумажных носителях, что крайне неудобно для всех людей, которые работают с этим. Для того, чтобы существенно облегчить жизнь работникам и усовершенствовать работу с данными, было принято решение о создании информационных систем.

Так как мир уже не может существовать без информационных технологий, каждое предприятие должно иметь свою информационную систему.

Первое преимущество такой системы состоит в том, что не нужно среди кучи бумаг и папок искать нужную информацию – она вся находится и легко ищется в такой компьютерной системе. Такое хранение данных существенно сокращает время поиска, следовательно, продуктивность работы сотрудников становится значительно выше.

Второе преимущество заключается в человеческом факторе. Очень легко допустить ошибку, заполняя документы вручную. Информационная система же содержит в себе поля редактирования данных, которая в случае невнимательности даст возможность исправить данные.

Третье же преимущество состоит в сохранности и надежности данных. Любая информационная система строится на основе закрытой базы данных, доступ к которой могут получить лишь те люди, которые непременно связаны с сущностью системы.

Целью дипломного проекта является разработка информационной системы для заказа продукции в пекарне. Система должна обеспечивать хранение данных о клиентах и заказах. Также должна присутствовать возможность добавления и редактирования заказов и клиентов на разных страницах соответственно Предусмотреть поисковую строку, которая будет производить поиск данных по набранной строке.

Для достижения цели были выделены следующие задачи:

* Проектирование базы данных в соответствии с требованиями к функционалу системы;
* Разработка базы данных;
* Разработка информационной системы;
* Разработка функционала информационной системы в соответствии с требованиями к программному продукту.

Разработка такой информационной системы позволит пекарне улучшить качество обслуживания клиентов, повысить эффективность производства и управления, а также получить конкурентное преимущество на рынке.

1. ОСНОВНАЯ ЧАСТЬ

1. Постановка задачи

Требуется разработать информационную систему для пекарни. Система должна предусматривать функции по работе информацией о клиентах, заказах и продукции. Каждый клиент должен содержать следующие параметры: уникальный код, наименование, адрес, номер телефона, email и ИНН. Для каждого клиента формируется свой заказ. Каждый заказ должен иметь следующие характеристики: уникальный код, код клиента, код менеджера, создавшего заказ, стоимость, статус, дату формирования и дату выполнения заказа. Вместе с этим следует предусмотреть возможность формирования отчета по заказу в формате PDF.

Для информации о заказах необходимо сделать раздел с описанием товаров, где можно будет прочитать о всей продукции, которая доступна для заказа. Данный раздел должен содержать такие характеристики, как: уникальный код, тип продукции, наименование, описание, цена и вес.

Также должны быть предусмотрены возможности добавления и редактирования клиентов и заказов. Предусмотреть поисковую строку для поиска клиентов по наименованию, номеру телефона, почте.

1. Описание использованных технологий

Так как в приложении следует хранить большой объем информации, то нельзя не воспользоваться базами данных. Для удобной и быстрой работы была выбрана SQL Server Management Studio 20.

SQL Server Management Studio (SSMS) — это интегрированная среда для управления любой инфраструктурой SQL, от SQL Server до баз данных SQL Azure. SSMS предоставляет средства для настройки, наблюдения и администрирования экземпляров SQL Server и баз данных. Используйте SSMS для развертывания, мониторинга и обновления компонентов уровня данных, используемых приложениями, а также для создания запросов и скриптов.

SQL хранит информацию в реляционном виде, которая представляет собой набор таблиц из типизированных столбцов. В данной СУБД поддерживаются различные типы данных, включая основные, такие как Integer, Numeric, Char, Character varying, Date, Text и другие.

Реализация программной части информационной системы производилась в среде разработки Microsoft Visual Studio 2022.

Visual Studio — это мощное средство разработчика, которое можно использовать для единого завершения всего цикла разработки. Это комплексная интегрированная среда разработки, которую можно использовать для написания, редактирования, отладки и сборки кода, а затем для развертывания приложения. Помимо редактирования и отладки кода, Visual Studio включает компиляторы, средства завершения кода, систему управления версиями, расширения и многие другие функции для улучшения каждого этапа процесса разработки программного обеспечения.

Графическая часть приложения выполнена с использованием Windows Presentation Foundation (.NET Framework).

Windows Presentation Foundation (WPF) — аналог WinForms, система для построения клиентских приложений Windows с визуально привлекательными возможностями взаимодействия с пользователем, графическая (презентационная) подсистема в составе .NET Framework, использующая язык XAML.

В основе WPF лежит независимый от разрешения векторный модуль визуализации, использующий возможности современного графического оборудования. Возможности этого модуля расширяются с помощью комплексного набора функций разработки приложений, которые включают в себя язык XAML, элементы управления, привязку к данным, макет, двумерную и трехмерную графику, анимацию, стили, шаблоны, документы, мультимедиа, текст и типографические функции.

В качестве основного языка программирования был выбран язык C#.

C# – объектно-ориентированный язык программирования, разработанный так же компанией Microsoft и выпущенный в 1998 году как язык разработки приложений для платформы Microsoft .NET Framework. C# относится к семейству C-подобных языков и по синтаксису наиболее близок к C++ и Java. К наиболее отличительным особенностям C# можно отнести статическую типизацию, перегрузку операторов и поддержку полиморфизма.

Доступ к базе данных осуществляется с помощью средства .NET Core под названием Entity Framework Core 7.

Entity Framework — это современный модуль сопоставления отношений объектов, который позволяет создавать чистый, переносимый и высокоуровневый уровень доступа к данным с помощью .NET (C#) в различных базах данных, включая Базу данных SQL (локально и Azure), SQLite, MySQL, PostgreSQL и Azure Cosmos DB. Он поддерживает запросы LINQ, отслеживание изменений, обновления и миграции схемы.

Для графического интерфейса было принято решение использовать средства Material Design.

Material Design — это дизайн-система для создания интерфейсов программного обеспечения и приложений, разработанная компанией Google. Стиль расширяет идею «карточек», появившуюся в Google Now, более широким применением строгих макетов, вариантов анимации переходов, отступов и эффектов глубины (света и тени). По идее графических дизайнеров Google, у приложений не должно быть острых углов, карточки должны переключаться между собой плавно и практически незаметно.

Доступ для реализации сохранения pdf файла было использовано iTextSharp.

ITextSharp — это библиотека, используемая для создания документов PDF. Она предоставляет все примитивные функции, необходимые для создания документа PDF. Однако, поскольку все методы основаны на примитивных операциях, легко запутать внешний вид документа без соблюдения определенных стандартов.

1. Проектирование базы данных

В процессе проектирования базы данных были выявлены следующие сущности:

Пользователи характеризуется следующими параметрами:

* уникальный код;
* логин;
* пароль;
* имя;
* роль.

Продукция характеризуется следующими параметрами:

* уникальный код;
* наименование;
* код типа товара;
* описание;
* вес;
* цена.

Каждый товар содержит свой тип.

Типы товара характеризуются следующими параметрами:

* уникальный код;
* наименование типа.

Клиенты характеризуется следующими параметрами:

* уникальный код;
* наименование;
* ИНН;
* адрес;
* номер телефона;
* email.

Каждый клиент может содержать определенный заказ.

Заказы характеризуются следующими параметрами:

* уникальный код;
* код клиента;
* код пользователя;
* код статуса;
* стоимость;
* дата начала;
* дата конца.

Каждый заказ содержит свой статус.

Статусы заказа характеризуются следующими параметрами:

* уникальный код;
* наименование статуса.

Заказанная продукция характеризуется следующими параметрами:

* уникальный код;
* код заказа;
* код товара;
* количество.

В результате проектирования была получена следующая диаграмма «Сущность-связь» (Рисунок 1):

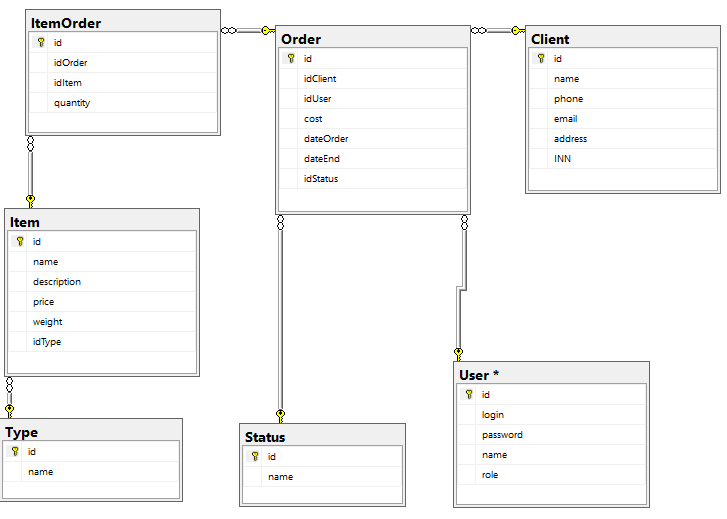


Рисунок 1 - ER диаграмма

Даталогическая модель базы данных показана в Таблицах 1, 2, 3, 4, 5, 6, 7:

Таблица 1 – Пользователи

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование поля | Тип | Ключи | Примечание |
| 1 | Уникальный код | Int | PK | Not Null |
| 2 | Логин | Nvarchar(50) |  | Not Null |
| 3 | Пароль | Nvarchar(50) |  | Not Null |
| 4 | Имя | Nvarchar(MAX) |  | Not Null |
| 5 | Роль | Nvarchar(50) |  | Not Null |

Таблица 2 – Тип товара

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование поля | Тип | Ключи | Примечание |
| 1 | Уникальный код | Int | PK | Not Null |
| 2 | Наименование типа | Nvarchar(MAX) |  | Not Null |

Таблица 3 – Продукция

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование поля | Тип | Ключи | Примечание |
| 1 | Уникальный код | Int | PK | Not Null |
| 2 | Код типа товара | Int | FK | Not Null |
| 3 | Название | Nvarchar(MAX) |  | Not Null |
| 4 | Описание | Nvarchar(MAX) |  | Not Null |
| 5 | Цена | Money |  | Not Null |
| 6 | Вес | Float |  | Not Null |

Таблица 4 – Статусы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование поля | Тип | Ключи | Примечание |
| 1 | Уникальный код | Int | PK | Not Null |
| 2 | Наименование статуса | Nvarchar(MAX) |  | Not Null |

Таблица 5 - Клиенты

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование поля | Тип | Ключи | Примечание |
| 1 | Уникальный код | Int | PK | Not Null |
| 2 | Наименование | Nvarchar(MAX) |  | Not Null |
| 3 | ИНН | Nvarchar(MAX) |  | Not Null |
| 4 | Адрес | Nvarchar(MAX) |  | Not Null |
| 5 | Номер телефона | Int |  | Not Null |
| 6 | Email | Nvarchar(MAX) |  | Not Null |

Таблица 6 – Заказы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование поля | Тип | Ключи | Примечание |
| 1 | Уникальный код | Int | PK | Not Null |
| 2 | Код клиента | Int | FK | Not Null |
| 3 | Код пользователя | Int | FK | Not Null |
| 4 | Код статуса | Int | FK | Not Null |
| 5 | Стоимость | Money |  |  |
| 6 | Дата конца | Date |  | Not Null |
| 7 | Дата начала | Date |  |  |

Таблица 7 – Заказанная продукция

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование поля | Тип | Ключи | Примечание |
| 1 | Уникальный код | Int | PK | Not Null |
| 2 | Код заказа | Int | FK | Not Null |
| 3 | Код товара | Int | FK | Not Null |
| 4 | Количество | Int |  | Not Null |

1. Проектирование приложения

Так как система предназначена для узкого круга лиц, то на текущий момент нет необходимости разграничивать доступ к данным приложения.

Диаграмма вариантов использования приложения: отражает отношения между пользователями и их возможности. (Рисунок 2)

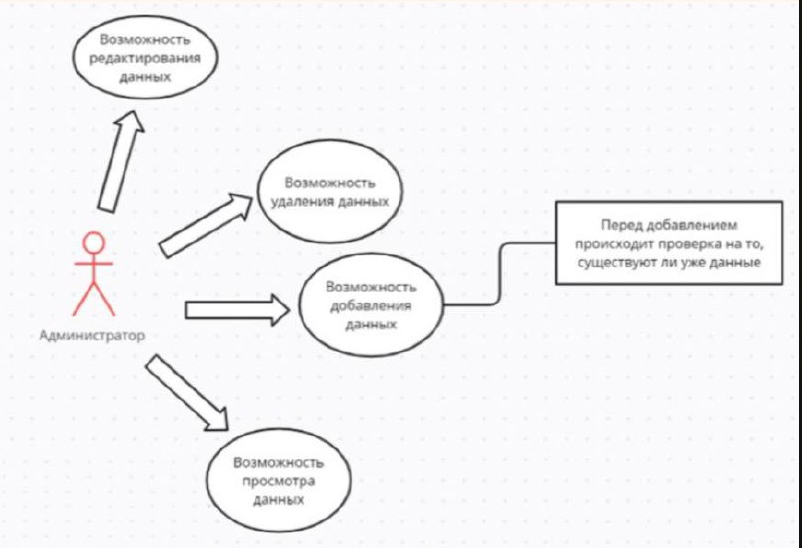


Рисунок 2 – Варианты использования

Диаграмма последовательности: описание жизненного цикла объекта и взаимодействие актеров информационной системы в рамках прецедента. (Рисунок 3)

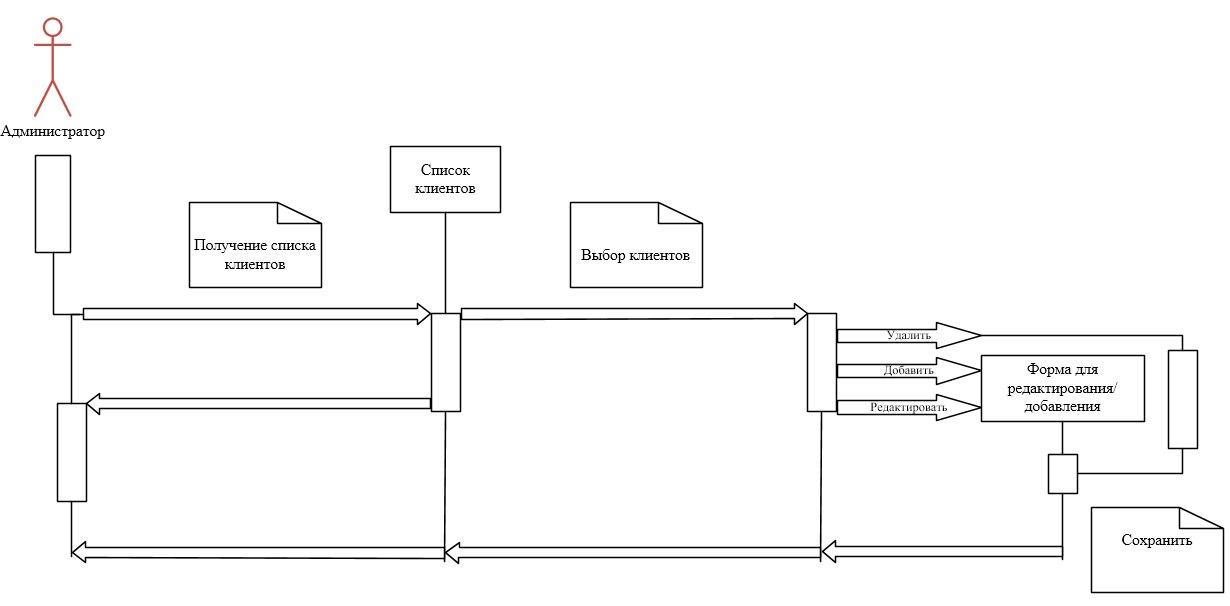


Рисунок 3 – Последовательность операций

Диаграмма деятельности: как поток управления переходит от одной деятельности к другой, при этом внимание фиксируется на результате деятельности. Результат может привести к изменению состояния системы или возвращению некоторого значения. (Рисунок 4)

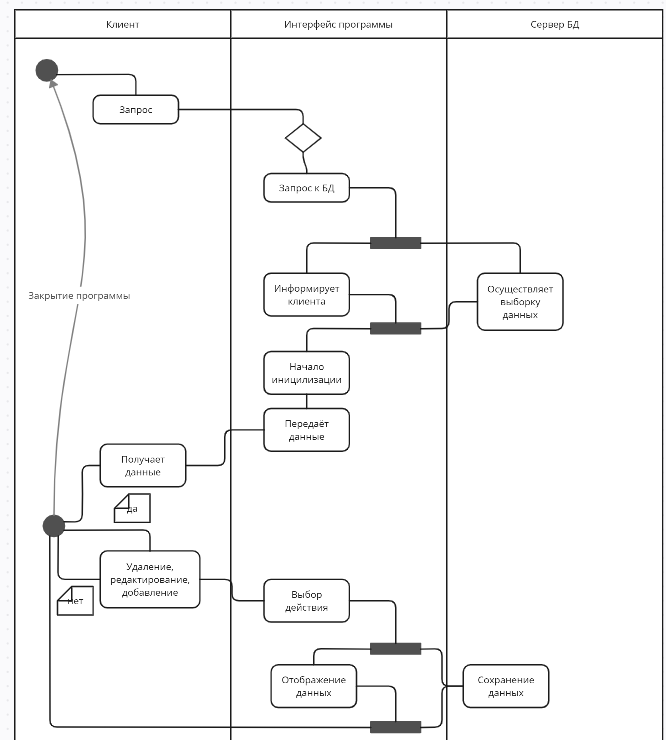


Рисунок 4 – Деятельность программы

1. Разработка приложения

Так как приложение связывается с СУБД, то при подключении базы данных в проекте автоматически создается модель базы данных с таблицами непосредственного взаимодействия. Модель базы данных представлена на Рисунке 5:

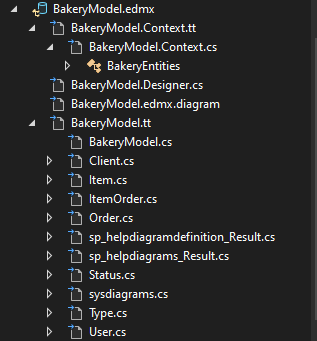


Рисунок 5 – Модель базы данных

Для работы с данными использовалась технология Entity Framework. Классы базы данных представлены в Таблице 8:

Таблица 8 – Таблица Entity Framework Core

|  |  |
| --- | --- |
| Название | Значение |
| Client.cs | Класс клиентов |
| Item.cs | Класс продукции |
| ItemOrder.cs | Класс заказанной продукции |
| Order.cs | Класс заказов |
| Users.cs | Класс пользователей |
| Type.cs | Класс типа продукции |
| Status.cs | Класс статуса заказа |

Приложение должно реализовывать обширный функционал, такой как:

* добавление данных;
* редактирование данных;
* поисковая строка;
* фильтрация;
* сохранение заказов pdf.

Обработка данных для списка выше представлена в Приложении А.

В приложении предусмотрено 10 форм, которые описаны в Таблице 9:

Таблица 9 – Формы приложения

|  |  |  |
| --- | --- | --- |
| Название | Основное назначение | Функционал |
| AuthWindow | Окно авторизации пользователя в систему, где требуется ввести логин и пароль | Возможность защитить данные от злоумышленников, необходимо заполнить все поля, чтобы войти в систему |
| MainWindow | Главное окно и возможностью перемещаться между страницами с помощью меню | Возможность перемещения и просмотра страниц с различными данными. |
| ItemWindow | Окно добавления продукции к заказу | Возможность добавления товаров в выбранный заказ |
| OrderPage | Страница, выводящая информацию о заказах, сохранение отчета по заказам в pdf, | Возможность просмотр заказов, сохранение заказа в pdf, позволяет перейти на страницу с добавлением и изменением заказа |

Продолжение таблицы 9

|  |  |  |
| --- | --- | --- |
| Название | Основное назначение | Функционал |
| ClientPage | Страница, выводящая информация о клиентах | Возможность просмотра информации о клиентах, поисковая система |
| OrderAddPage | Добавление контрактов | Возможность создания заказа для клиента |
| ClientAddPage | Добавление клиентов | Возможность добавления нового клиента |
| ItemPage | Вывод информации о продукции | Возможность просмотра данных товаров |
| OrderItemPage | Добавление продукции к заказу | Возможность добавить продукцию к заказу |

Чтобы начать работу с приложением, запустите файл “BakeryApp.exe” откроется следующее окно, где нужно ввести логин и пароль и нажать кнопку войти, после входа откроется окно MainWindow (Рисунок 6 -7):

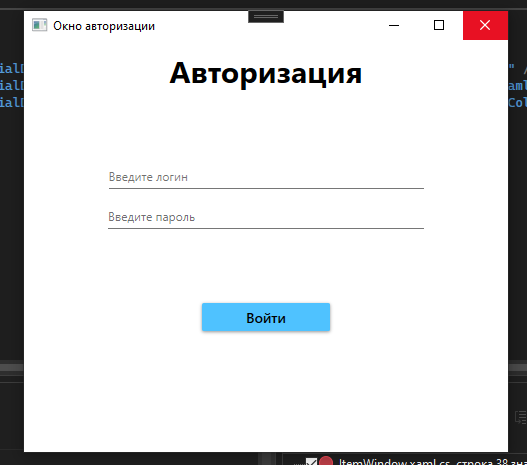


Рисунок 6 – Окно AuthWindow

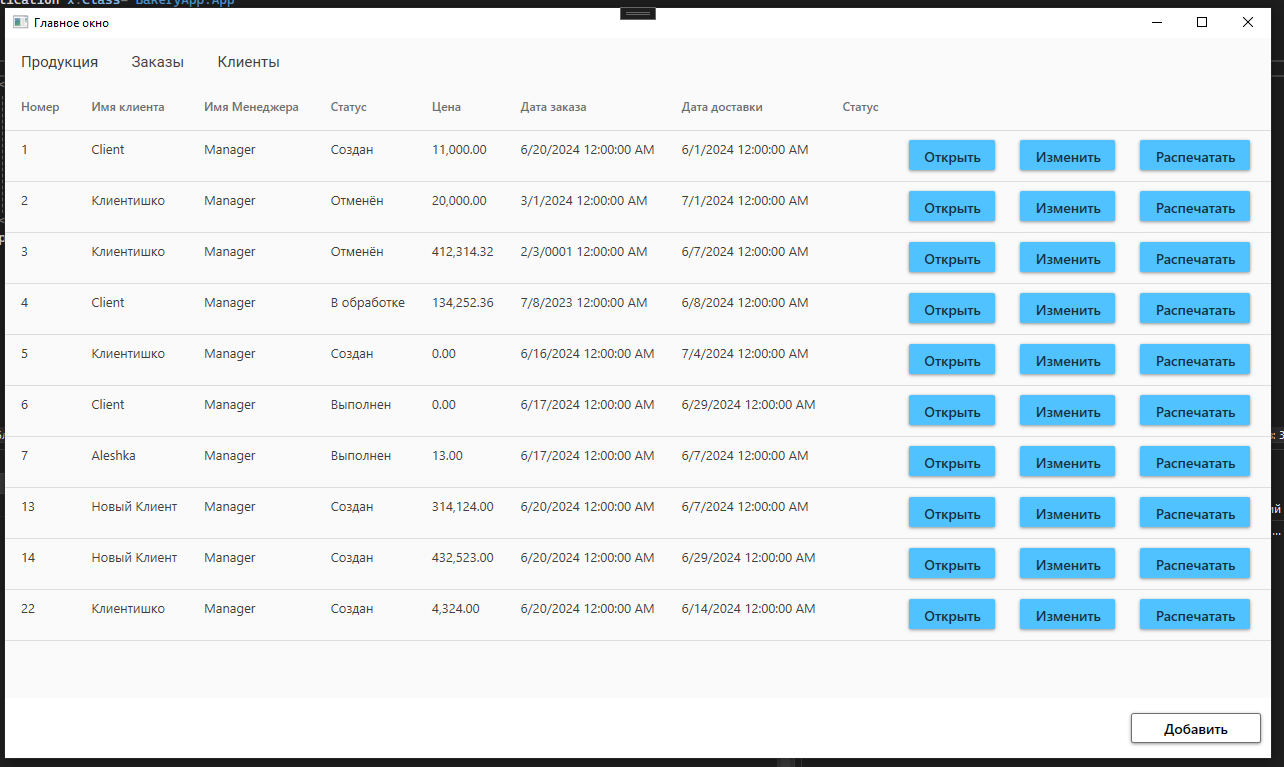


Рисунок 7 – Окно MainWindow

В MainWindow можно перемещаться по страницам OrderPage, OrderAddPage ItemPage, ClientPage, ClientAddPage с помощью меню сверху окна (Рисунок 8):

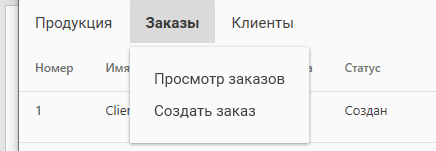


Рисунок 8 – Меню MainWindow

На странице ItemPage выводится информация о продукции, доступной для заказа (Рисунок 9):



Рисунок 9– Страница ItemPage

На странице OrderPage выводится информация о заказах, с возможностью сохранения заказа в pdf файле и перехода на страницы редактирования и добавления заказов (Рисунок 10):

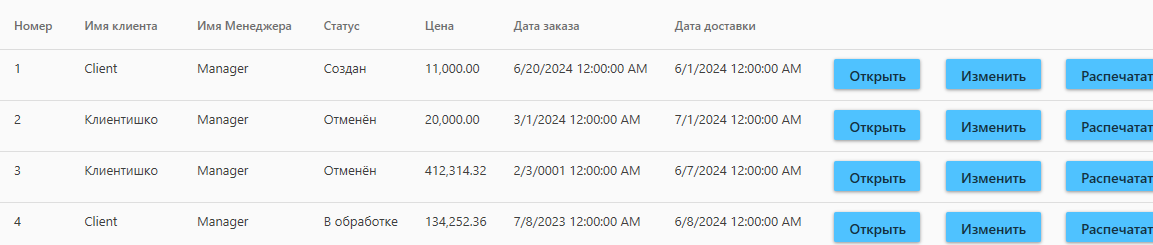


Рисунок 10 – Страница OrderPage

На странице ClientPage выводится информация о клиентах, с возможностью перехода на страницу добавления клиента (Рисунок 11):

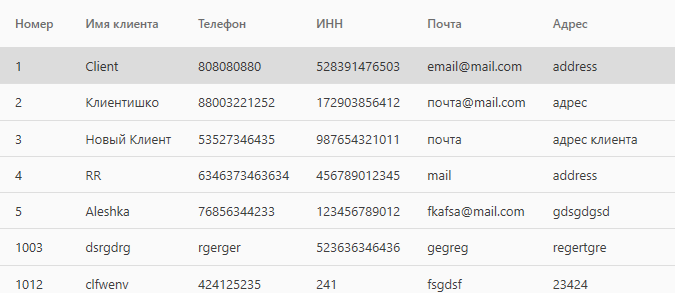


Рисунок 11 – Страница ClientPage

Также на странице присутствует функциональный поиск по наименованию клиента, номеру телефона, ИНН, почте и адресу. (Рисунок 12):

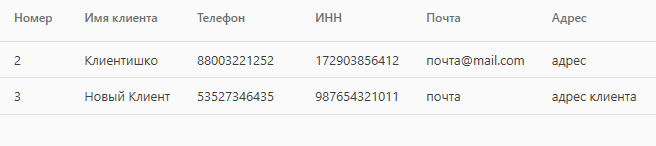


Рисунок 12 – Поиск по адресу

На странице ClientAddPage осуществляется добавление клиента в базу данных (Рисунок 13):

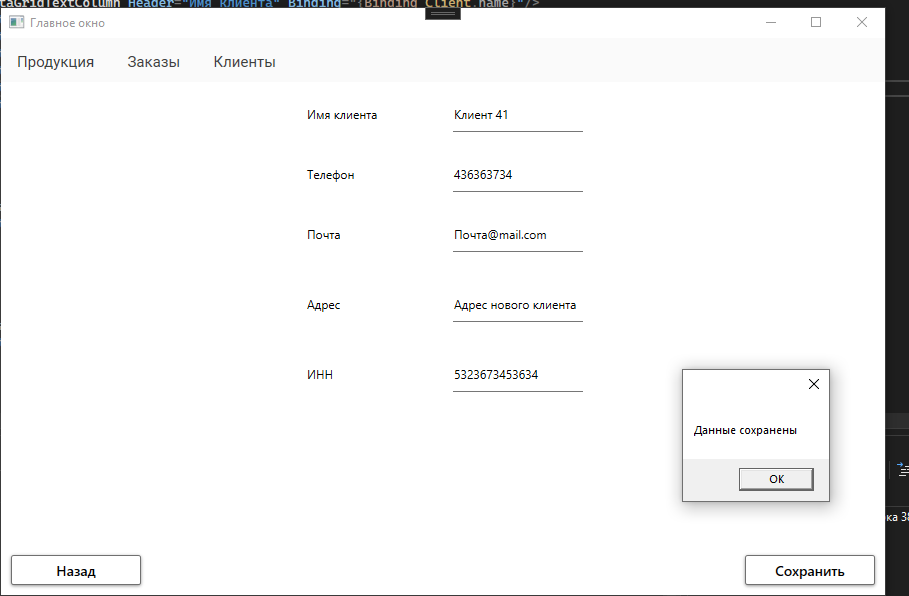


Рисунок 13 – Страница ClientAddPage

На странице OrderAddPage осуществляется создание заказа (Рисунок 14):

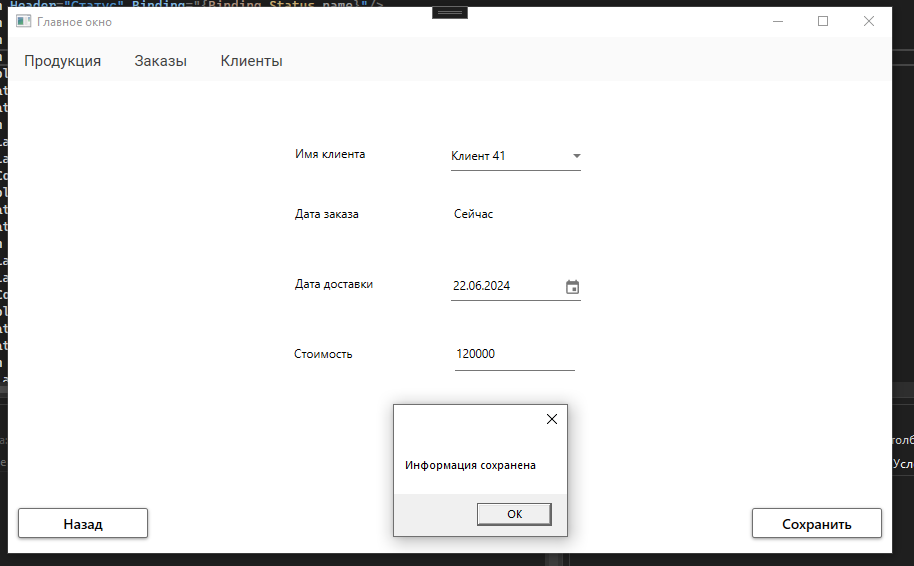


Рисунок 14 – Страница OrderAddPage

В окне ItemWindow происходит добавление товара к заказу (Рисунок 15):

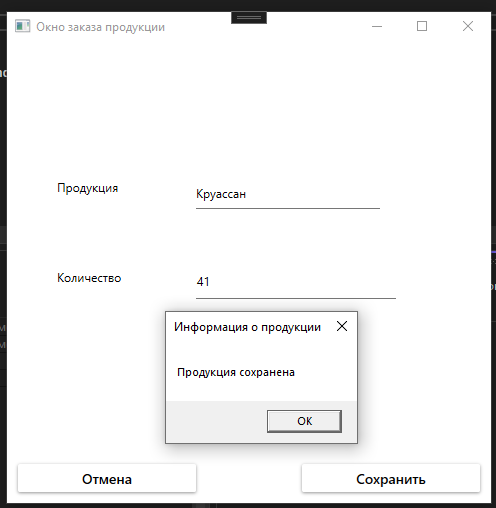


Рисунок 15 – Окно ItemWindow

На странице OrderItemPage осуществляется просмотр заказанной продукции (Рисунок 16):

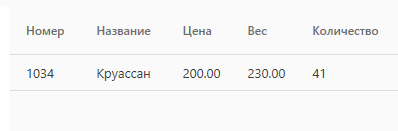


Рисунок 16 – Страница OrderItemPage

Возможность сохранение заказа в pdf файле (Рисунок 17-18):

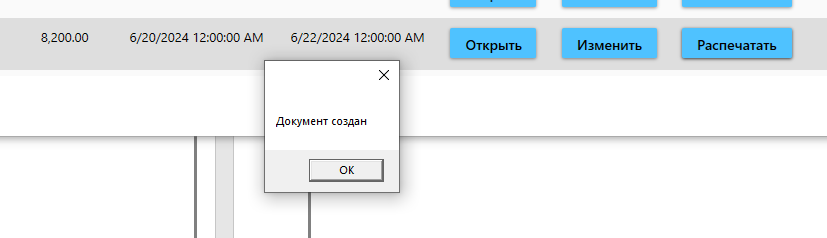


Рисунок 17 – Сохранение заказа в pdf файл

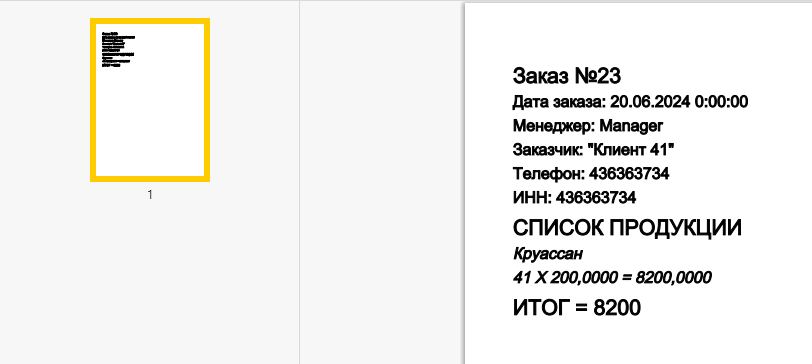


Рисунок 18 – Заказ в виде pdf файла

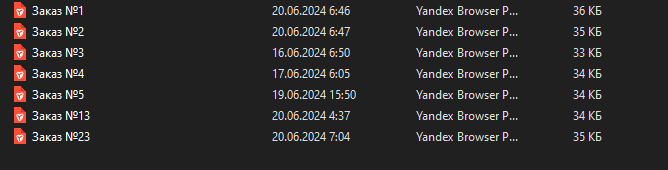


Рисунок 19 – Папка с сохраненными чеками

На странице OrderAddPage осуществляется редактирование заказа

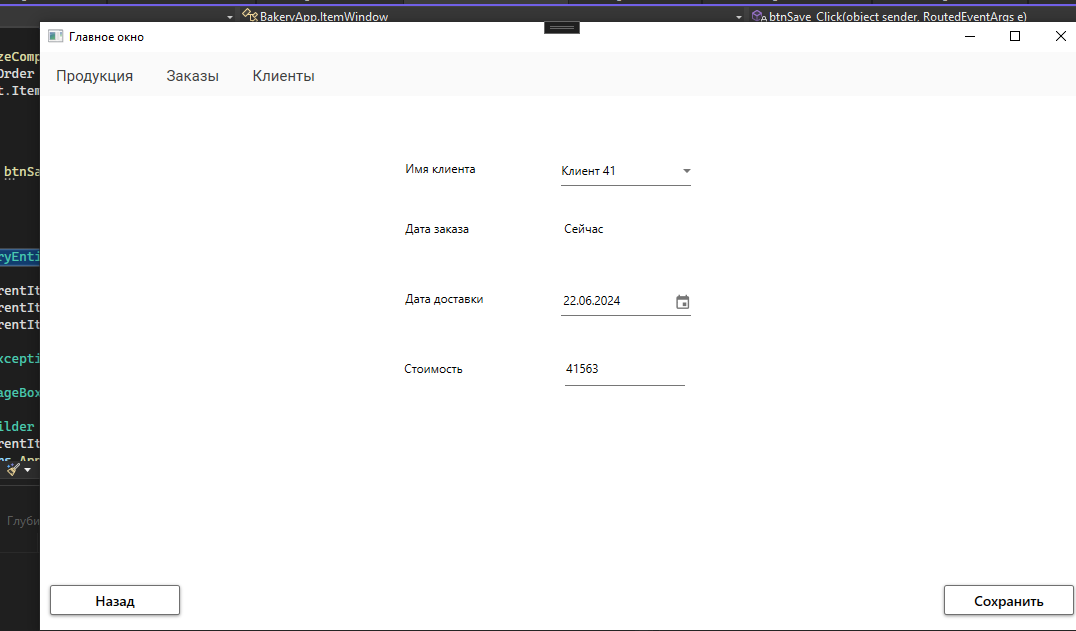


Рисунок 20 – Страница OrderAddPage

1. ЭКОНОМИЧЕСКИЙ РАЗДЕЛ

2.1 Понятие себестоимости

Себестоимость (товаров, работ, услуг) — стоимостная оценка используемых в процессе производства природных ресурсов, сырья, материалов, основных фондов, трудовых ресурсов и других затрат на ее производство и реализацию.

В себестоимости находят отражение стоимостная оценка природных ресурсов, сырья, материалов, топлива, энергии, трудовых ресурсов.

В себестоимость включается часть стоимости основных фондов, используемых для производства продукции, а также расходы, связанные с обслуживанием и управлением производством.

Себестоимость продукции образует основу цены товаров и является одним из качественных показателей работы предприятия.

От уровня себестоимости зависят прибыль, получаемая предприятиями и рентабельность продукции.

Себестоимость может быть фактической и нормативной.

При расчете фактической себестоимости берут реальные данные, т.е. исходя из фактических издержек, формируется цена товара, работы или услуги. Производить такой расчет очень неудобно, т.к. зачастую узнать себестоимость необходимо раньше, чем производится конкретное производительное действие. От этого зависит рентабельность бизнеса.

Рассчитывая нормативную себестоимость, данные берут согласно производственных норм.

Благодаря этому можно жестко контролировать расход материалов, что минимизирует возникновение неоправданных расходов.

В следующем вопросе конкретизируем затраты, включаемые в состав себестоимости.

2.2 Затраты в составе себестоимости

Перечень расходов, включаемых в себестоимость продукции (работ, услуг) в целях расчета налогооблагаемой прибыли, определяется в соответствии с законодательством Российской Федерации, в частности с главой 25 НК РФ «Налог на прибыль организаций».

Расходами признаются обоснованные и документально подтвержденные затраты, осуществленные предприятием.

Под обоснованными расходами понимаются экономически оправданные затраты, оценка которых выражена в денежной форме.

Под документально подтвержденными расходами понимаются затраты, подтвержденные документами, оформленными в соответствии с законодательством Российской Федерации. Расходами признаются любые затраты при условии, что они произведены для осуществления деятельности, направленной на получение дохода.

Расходы в зависимости от их характера, а также условий осуществления и направлений деятельности предприятия подразделяются на расходы, связанные с производством и реализацией, и внереализационные расходы.

Расходы, связанные с производством и реализацией, включают в себя:

* расходы, связанные с изготовлением (производством) или реализацией товаров (работ, услуг, имущественных прав);
* расходы на содержание и эксплуатацию, ремонт и техническое обслуживание основных средств и иного имущества, а также на поддержание их в исправном состоянии;
* расходы на обязательное и добровольное страхование;
* прочие расходы, связанные с производством и (или) реализацией.

Расходы, связанные с производством и реализацией и образующие себестоимость продукции, группируются в соответствии с их экономическим содержанием по следующим элементам:

* материальные расходы;
* расходы на оплату труда;
* суммы начисленной амортизации;
* прочие расходы.

К материальным расходам, в частности, относятся следующие затраты предприятия:

* на приобретение сырья и (или) материалов, которые входят в состав вырабатываемой продукции (основные материалы);
* приобретение материалов для обеспечения технологического процесса и для упаковки продукции или расходуемых на другие производственные и хозяйственные нужды;
* приобретение запасных частей и расходных материалов, используемых для ремонта оборудования, инструментов, приспособлений и др.;
* приобретение топлива, воды и энергии всех видов, расходуемых на технологические цели, выработку всех видов энергии, отопление зданий, а также расходы на трансформацию и передачу энергии;
* расходы, связанные с содержанием и эксплуатацией фондов природоохранного назначения, а также платежи за предельно допустимые выбросы загрязняющих веществ в природную среду.

Под возвратными отходами понимаются остатки сырья, материалов, полуфабрикатов, теплоносителей и других видов материальных ресурсов, полностью или частично утратившие потребительские качества исходного ресурса и в силу этого используемые с повышенными затратами (с понижением выхода продукции) или вовсе не используемые по прямому назначению.

К расходам на оплату труда относятся затраты предприятия на оплату труда основного производственного персонала, а также не состоящих в штате работников, участвующих в основной деятельности предприятия.

Затраты на оплату труда включают в себя заработную плату, начисляемую по сдельным расценкам, тарифным ставкам и должностным окладам в соответствии с системами оплаты труда, принятыми на предприятии; надбавки о доплаты; премии за производственные результаты; оплату очередных и дополнительных отпусков; единовременные вознаграждения за выслугу лет и другие расходы, предусмотренные законодательством Российской Федерации.

К суммам начисленной амортизации относятся суммы амортизационных отчислений по амортизируемому имуществу (основным средствам и нематериальным активам).

К прочим расходам относятся:

* расходы на сертификацию продукции и услуг;
* расходы на оплату услуг по охране имущества, обслуживанию охранной пожарной сигнализации, услуг пожарной охраны и иных услуг охранной деятельности;
* расходы по обеспечению нормальных условий труда и техники безопасности, предусмотренных законодательством РФ;
* арендные (лизинговые) платежи за арендуемое имущество;
* расходы на содержание служебного автотранспорта;
* расходы на командировки в пределах норм, утвержденных правительством РФ;
* расходы на канцелярские товары;
* расходы на подготовку и освоение новых производств, цехов и агрегатов;
* расходы на ремонт основных средств.

В состав внереализационных расходов включают:

* расходы на содержание переданного по договору аренды имущества;
* проценты по долговым обязательствам любого вида, включая проценты по кредитам и ценные бумаги, выпущенным предприятием;
* расходы, связанные с организацией выпуска и обслуживанием собственных ценных бумаг;
* судебные расходы и арбитражные сборы и др.

К внереализационным расходам в целях определения налогооблагаемой прибыли приравниваются убытки, связанные с потерями от брака, потерями от стихийных бедствий, пожаров, аварий и других чрезвычайных ситуаций, а также убытки прошлых периодов, выявленные в текущем периоде.

2.3 Расчет себестоимости информационной системы

Расчет себестоимости разрабатываемой информационной системы включает определенную последовательность действий. Покажем их содержание, применяя расчетные действия. Их результаты приведем в таблицах, используя при необходимости соответствующие формулы.

Расчет трудоемкости произведен в таблице 10.

Таблица 10 – Расчет трудоёмкости

|  |  |
| --- | --- |
| Виды работ | Трудоёмкость, дн. |
| Получение ТЗ | 1 |
| Сбор информации и ознакомление с предметной | 2 |
| областью |  |
| Разработка основного алгоритма | 2 |
| Разработка программы | 11 |
| Отладка | 1 |
| Подготовка технической документации | 2 |
| Сдача продукта | 1 |
| Итого | 20 |

Расчет затрат на разработку информационной системы.

Величина себестоимости разрабатываемого программного продукта рассчитывается по формуле 1.

С = Сзп + Сэл + Сам + См + Сн + Ссоц.ф , где: (1)

С – величина себестоимости разрабатываемого программного продукта;

Сзп – заработная плата;

СЭл – расходы на электроэнергию;

САм – сумма амортизационных отчислений;

См – затраты на расходные материалы;

Сн – накладные расходы;

Ссоц.ф – отчисления в социальные фонды;

Численность – 2 человека (студент-разработчик и преподаватель).

Зарплата персонала (ЗП) показана в таблице 11. За основу заработной платы студента, разрабатывающего программный продукт, взята величина средней заработной платы на должности «Инженер-программист» в городе Коломна.

Таблица 11 – Расчет основной заработной платы

|  |  |  |
| --- | --- | --- |
| Персонал | Количество  (чел.) | ЗП за время разработки  (руб.) |
| Руководитель  диплома | 1 | 3 200 |
| Студент | 1 | 34 920 |
| Всего | 2 | 38 120 |

Расчет заработной платы.

Заработная плата руководителя диплома за час = 400 руб.

Время работы руководителя диплома = 8 ч.

Заработная плата руководителя диплома на время разработки дипломного проекта = 8 \* 400 = 3 200руб.

Величина оплаты труда студента на своей рабочей должности составит 54 115 руб. В расчетах исходим из того, что заработная плата начисляется за 31 день. Посуточная заработная плата студента за один месяц работы в мае составит 54 115 / 31 = 1 746 руб.

Время работы студента – 160 часов за 20 дней (с учетом 8 часов работы в день) осуществления проекта. Тогда заработная плата за 20 дней работы составит 1 746\* 20 = 34 920 рублей.

Затраты на электроэнергию:

Сэл=Р\*Т\* Z, где: (2)

Р – общая мощность оборудования (КВт);

Т – общее время работы оборудования (р.ч.);

Z – цена одного КВт (В 2024 г. – 6 руб. 73 коп).

Затраты на электроэнергию показаны в таблице 12.

Таблица 12 – Расчет затрат на электроэнергию

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Оборудование | Количество,  шт. | Общая  стоимость,  руб. | Общая  мощность,  КВт | Количество часов работы, р.ч. |
| Компьютер | 1 | 861,44 | 0,8 | 160 |
| Струйный принтер | 1 | 14,13 | 0,7 | 3 |
| Итого | 2 | 875,57 | 1,5 | 163 |

Амортизационные отчисления.

Расчет амортизационных отчислений на рабочий период производится по формуле 3.

Сам = (Спер\*Н a) / (365/20), где: (3)

Спер – первоначальная стоимость оборудования;

Нa – годовая норма амортизации (20%);

365 – количество дней в году;

20 – количество дней на разработку проекта.

Амортизационные отчисления рассчитываются на основании таблицы 13.

Таблица 13 – Амортизационные отчисления

|  |  |  |  |
| --- | --- | --- | --- |
| Оборудование | Количество,  шт. | Первоначальная стоимость оборудования, руб. | Стоимость амортизационных отчислений, руб. |
| Компьютер | 1 | 58 000 | 635,62 |
| Струйный  принтер | 1 | 12 300 | 134,79 |
| Итого | 2 | 70 300 | 770,41 |

Затраты на материалы.

Расчет затрат на материалы осуществляется согласно фактическим данным в год, которые представлены в таблице 14.

Таблица 14 – Расчет затрат на материалы

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование  материала | Количество шт. | Стоимость за 1 шт., руб. | Общая стоимость, руб. |
| Бумага | 1 | 400 | 400 |
| Краска для принтера | 1 | 1 200 | 1 200 |
| Флэш-накопитель | 1 | 800 | 800 |
| Итого | | | 2 400 |

Носителем информации был выбран Флэш-накопитель, так как этот вариант является наиболее экономичным.

Произведем расчет накладных расходов для выбранного проекта и покажем его в таблице 15.

В нашей стране законодательно закреплены накладные расходы только в таких сферах как строительство и медицина и в ряде других видах деятельности. Остальные предприниматели и компании устанавливают самостоятельно перечень подобных затрат. Установим величину накладных расходов на уровне 31%.

Накладные расходы (Общепроизводственные и общехозяйственные расходы):

Сн = ЗПосн\*% накл. расх. /100, где: (4)

% накл. расх. – процент накладных расходов – 31.

Таблица 15 – Расчет накладных расходов

|  |  |  |  |
| --- | --- | --- | --- |
| Персонал | Количество | Заработная | Накладные |
|  | (чел) | плата, руб. | расходы, руб. |
| Руководитель диплома | 1 | 3 200 | 992 |
| Студент | 1 | 34 920 | 10 825,2 |
| Итого | | | 11 817,2 |

Расчет отчислений в социальные фонды показаны в таблице 16:

Ссоцф = ЗПосн \* % отч. / 100, где: (5)

% отч. – процент отчислений (В 2024 г. – 30).

Таблица 16 – Расчет отчислений в социальный фонд

|  |  |  |  |
| --- | --- | --- | --- |
| Персонал | Количество  (чел) | Заработная плата, руб. | Сумма отчислений, руб. |
| Руководитель дплома | 1 | 3 200 | 960 |
| Студент | 1 | 34 920 | 10 476 |
| Итого | | | 11 436 |

Затраты на разработку программного продукта рассчитываются по формуле (6):

С = 38 120 + 875,57 + 770,41 + 2 400 + 11 817,2 + 11 436 = 65 419,18 руб. (6)

Удельный вес по элементам затрат представлен в таблице 17.

Таблица 17 – Удельный вес по элементам затрат

|  |  |  |
| --- | --- | --- |
| Элементы затрат | Сумма (руб.) | Удельный вес (%) |
| Зарплата персонала | 38 120 | 58,27 |
| Расходы на электроэнергию | 875,57 | 1,34 |
| Сумма амортизационных отчислений | 770,41 | 1,18 |
| Сумма затрат на материалы | 2 400 | 3,67 |
| Накладные расходы | 11 817, 2 | 18,06 |
| Отчисления в соц. фонды | 11 436 | 17,48 |
| Итого | 65 419,18 | 100 |

На рисунке 21 представлена диаграмма себестоимости программного продукта.

Из диаграммы следует, что наибольший удельный вес поэлементных затрат характерен для зарплаты персонала, накладных расходов и отчислений в социальные фонды.

Изображение выглядит как текст, снимок экрана, диаграмма, круг

Автоматически созданное описание

Рисунок 21 – Себестоимость программного продукта, в процентах (%)

В ходе выполнения экономических расчетов был проведен анализ издержек, возникающих при разработке информационной системы. В результате проведенного анализа удалось выявить, что основная сумма себестоимости складывается из заработной платы (58,27%), расходов на электроэнергию (1,34%), отчислений в социальные фонды (17,48%), амортизационных отчислений (1,18%), накладных расходов (18,06%) и затрат на материалы (3,67%). Общая сумма себестоимости разработанной информационной системы составляет 65 419,18 рублей.

Окупаемость полученной разработки возможна только в случае ее коммерческого применения или реализации на основе соответствующего бюджетного заказа с учетом ожидаемой нормы прибыли.

Проведенное исследование позволило получить навык рассчитывать затраты на разработку информационной системы в целях последующего внедрения в системы автоматизации процессов на предприятии в соответствии с освоением профильной специальности.

1. ТЕХНИКА БЕЗОПАСНОСТИ ПРИ РАБОТЕ С ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКОЙ

Работа с вычислительной техникой требует соблюдения определенных мер по технике безопасности, чтобы предотвратить возможные риски и обеспечить защиту как самих устройств, так и пользователей. Ниже приведены общие требования по технике безопасности, которые следует учесть при работе с вычислительной техникой. Чем больше из них будет применено, тем лучше будет обеспечена безопасность.

Физическая безопасность:

* Обеспечение безопасности помещения, где находится вычислительная техника, с помощью контроля доступа и системы видеонаблюдения.
* Установка вычислительной техники на стабильные поверхности или крепление ее к специальным стойкам, чтобы предотвратить падение и повреждение устройств.
* Защита от несанкционированного доступа путем использования физических замков, паролей или биометрических систем идентификации.

Защита от электрических рисков:

* Правильное заземление и подключение всех компонентов вычислительной техники.
* Использование стабилизаторов напряжения и защитных предохранителей для предотвращения повреждения оборудования от перепадов напряжения или коротких замыканий.

Защита от программных угроз:

* Установка и регулярное обновление антивирусного программного обеспечения для защиты от вредоносных программ и вирусов.
* Использование брандмауэров и других средств защиты, чтобы предотвратить несанкционированный доступ к сети или компьютерным ресурсам.

Защита данных:

* Регулярное создание резервных копий важных данных и хранение их в надежном месте.
* Шифрование конфиденциальной информации и использование сильных паролей для защиты данных от несанкционированного доступа.
* Ограничение доступа к конфиденциальным данным только уполномоченным сотрудникам с помощью различных уровней доступа и паролей.

Регулярное обслуживание и проверка:

* Проведение регулярных проверок системы на наличие уязвимостей и исправление обнаруженных проблем.
* Очистка и обслуживание компонентов вычислительной техники, включая прочистку вентиляционных отверстий и замену изношенных деталей.

Обучение пользователей:

* Проведение обучающих программ для сотрудников, в которых рассказывается о мерах безопасности и правилах работы с вычислительной техникой.
* Оповещение пользователей о потенциальных угрозах безопасности, таких как фишинговые атаки или социальная инженерия, и научение их распознавать и предотвращать такие атаки.

Регулярное обновление программного обеспечения:

* Установка всех доступных обновлений операционных систем, приложений и драйверов, чтобы исправить известные уязвимости и обеспечить безопасность системы.

Рабочее место и правильная посадка:

Перед тем как приступить к работе, уделите немного времени организации своего рабочего места. Клавиатура и экран монитора должны быть расположены прямо перед вами. Если конструкция стула позволяет регулировать высоту сиденья, настройте ее таким образом, чтобы в положении сидя ваши ноги прочно опирались на пол или на специальную подставку, а поверхность, на которой расположены клавиатура и мышь, находились на уровне локтя или немного ниже. Разместите клавиатуру и мышь таким образом, чтобы при работе с ними не возникало необходимости отдалять локти от туловища.

Это лишь некоторые общие требования по технике безопасности при работе с вычислительной техникой. Рекомендуется также обращаться к рекомендациям и стандартам безопасности, связанным с конкретными типами устройств и средами, в которых они используются.

ВЫВОДЫ И ЗАКЛЮЧЕНИЕ

Цель дипломного проекта - разработка информационной системы для пекарни. В результате разработки была создана работоспособная и отказоустойчивая информационная система для хранения данных о клиентах и заказах.

Для достижения цели были выполнены следующие задачи:

Спроектирована и разработана база данных в соответствии с требованиями к функционалу системы;

Разработано десктопное приложение для работы с информационной системой;

Разработан функционал информационной системы в соответствии с требованиями к программному продукту.

Был произведен подсчет себестоимости информационной системы. Общая сумма себестоимости разработанного приложения составляет 65 419,18 рубля. Окупаемость полученной разработки возможна только в случае ее коммерческого применения или реализации на основе соответствующего бюджетного заказа с учетом ожидаемой нормы прибыли.

В разделе «Требования по технике безопасности при работе с вычислительной техникой» рассмотрены вопросы организации безопасной работы с компьютерной техникой.

СПИСОК ЛИТЕРАТУРЫ

1. Абрамян А.В. Разработка пользовательского интерфейса на основе технологии Windows Presentation Foundation: учебник для студ. Учреждений сред. Проф. Образования. – М.: Издательство Южного федерального университета, 2023.
2. Ганенко А.П. Оформление текстовых и графических материалов при подготовке дипломных проектов, курсовых и письменных экзаменационных работ (требования ЕСКД): учебно-метод. Пособие для студ. Учреждения сред. Проф. – М.: Издательский центр «Академия», 2023.
3. Виноградова Н.А. Научно-исследовательская работа студента: Технология написания и оформления доклада, реферата, курсовой и выпускной квалификационной работы: учеб. пособие для студ. Учреждений сред. Проф. Образования. – М.: Издательский центр «Академия», 2023.
4. Фёдорова Г.Н. Разработка программных модулей программного обеспечения для компьютерных систем: учебник для студ. Учреждений сред. Проф. Образования. – М.: Издетельский центр «Академия», 2023.

Электронные ресурсы:

1. Metanit – язык программирования С# и платформа .NET [Электронный ресурс] Режим доступа: свободный <https://metanit.com/sharp/> (Дата обращения 13.06.2024)
2. Национальный открытый университет [Электронный ресурс] Режим доступа: свободный https://www.intuit.ru/ (Дата обращения 12.06.2024)
3. Professor Web [Электронный ресурс] Режим доступа: свободный https://professorweb.ru/ (Дата обращения 13.06.2024)

ПРИЛОЖЕНИЕ А – Код программы

Весь проект был прикреплен и загружен на публичный репозиторий системы GitHub.

Ссылка на Github: https://github.com/Luccczi/BakeryApp

Обработка данных:

* Окно AuthWindow:

public partial class AuthWindow : Window

{

private User \_currentUser = new User();

public AuthWindow()

{

InitializeComponent();

}

* Вход на окно MainWindow:

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

{

try

{

\_currentUser = BakeryEntities.GetContext().User.FirstOrDefault(c => c.login == tbLogin.Text && c.password == pbPass.Password);

if (\_currentUser == null)

{

throw new Exception("Пользователь не найден");

}

if (\_currentUser != null)

{

MessageBox.Show("Успешный вход");

MainWindow mainWindow = new MainWindow();

mainWindow.Show();

Close();

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString(), "ОШИБКA", MessageBoxButton.OK, MessageBoxImage.Error);

}

* Окно MainWindow:

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

MainFrame.Navigate(new OrderPage());

Manager.MainFrame = MainFrame;

}

* Переход на страницу клиентов:

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

{

MainFrame.Navigate(new ClientPage());

}

* Переход на страницу заказов:

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

{

MainFrame.Navigate(new OrderPage());

}

* Переход на страницу продукции:

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

{

MainFrame.Navigate(new ItemPage());

}

* Страница ItemPage:

public partial class ItemPage : Page

{

public ItemPage()

{

InitializeComponent();

dGridItems.ItemsSource = BakeryEntities.GetContext().Item.ToList();

}

}

* Переход на страницу создания заказа:

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

{

Manager.MainFrame.Navigate(new OrderAddPage(null));

}

* Переход на страницу добавления клиентов:

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

{

Manager.MainFrame.Navigate(new ClientAddPage());

}

* Окно ItemWindow:

public partial class ItemWindow : Window

{

private Order \_currentOrder = new Order();

private ItemOrder \_currentItemOrder = new ItemOrder();

public ItemWindow(Order selectedOrder)

{

InitializeComponent();

\_currentOrder = selectedOrder;

cbProduct.ItemsSource = BakeryEntities.GetContext().Item.ToList();

}

* Кнопка добавления товара в заказ:

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

{

try

{

BakeryEntities.GetContext().ItemOrder.Add(\_currentItemOrder);

\_currentItemOrder.idOrder = \_currentOrder.id;

\_currentItemOrder.idItem = Convert.ToInt32(cbProduct.SelectedIndex)+1;

\_currentItemOrder.quantity = Convert.ToInt32(tbCount.Text);

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString());

}

StringBuilder errors = new StringBuilder();

if (\_currentItemOrder.idItem == 0)

errors.AppendLine("Выберите продукцию");

if (\_currentItemOrder.quantity == 0)

errors.AppendLine("Укажите количество");

if (\_currentItemOrder.quantity <= 0)

errors.AppendLine("Количество должно быть больше 0");

if (errors.Length > 0)

{

MessageBox.Show(errors.ToString());

return;

}

try

{

BakeryEntities.GetContext().ItemOrder.Add(\_currentItemOrder);

BakeryEntities.GetContext().SaveChanges();

var rez = MessageBox.Show("Продукция сохранена", "Информация о продукции", MessageBoxButton.OK);

if (rez == MessageBoxResult.OK)

{

Close();

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString());

}

this.Close();

}

* Кнопка отмены:

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

{

MessageBox.Show("Данные не были сохранены");

this.Close();

}

* Страница OrderPage:

public OrderPage()

{

InitializeComponent();

dGridOrders.ItemsSource = BakeryEntities.GetContext().Order.ToList();

}

* Кнопка сохранения pdf документа:

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

{

Manager.MainFrame.Navigate(new OrderItemPage((sender as Button).DataContext as Order));

}

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

{

Order selectedOrder = dGridOrders.SelectedItem as Order;

string path = $"C:/Other/Soft/BakeryApp/Заказ №{selectedOrder.id}.pdf";

iTextSharp.text.Document doc = new iTextSharp.text.Document();

BaseFont baseFont = BaseFont.CreateFont("C:/Windows/Fonts/arial.ttf", BaseFont.IDENTITY\_H, BaseFont.NOT\_EMBEDDED);

iTextSharp.text.Font font = new iTextSharp.text.Font(baseFont, iTextSharp.text.Font.DEFAULTSIZE, iTextSharp.text.Font.BOLD);

iTextSharp.text.Font font1 = new iTextSharp.text.Font(baseFont, 16, iTextSharp.text.Font.BOLD);

iTextSharp.text.Font font2 = new iTextSharp.text.Font(baseFont, iTextSharp.text.Font.DEFAULTSIZE, iTextSharp.text.Font.BOLDITALIC);

using (var writer = PdfWriter.GetInstance(doc, new FileStream(path, FileMode.Create)))

{

doc.Open();

doc.AddTitle($"Заказ №{selectedOrder.id}");

doc.NewPage();

doc.Add(new iTextSharp.text.Paragraph($"Заказ №{selectedOrder.id}", font1));

doc.Add(new iTextSharp.text.Paragraph($"Дата заказа: {selectedOrder.dateOrder}", font));

doc.Add(new iTextSharp.text.Paragraph($"Менеджер: {selectedOrder.User.name}", font));

doc.Add(new iTextSharp.text.Paragraph($"Заказчик: \"{selectedOrder.Client.name}\"", font));

doc.Add(new iTextSharp.text.Paragraph($"Телефон: {selectedOrder.Client.phone}", font));

doc.Add(new iTextSharp.text.Paragraph($"ИНН: {selectedOrder.Client.phone}", font));

doc.Add(new iTextSharp.text.Paragraph("СПИСОК ПРОДУКЦИИ", font1));

List<ItemOrder> itemOrders = BakeryEntities.GetContext().ItemOrder.Where(p => p.idOrder == selectedOrder.id ).ToList();

for (int i = 1; i <= itemOrders.Count; i++)

{

ItemOrder selectedItem = BakeryEntities.GetContext().ItemOrder.Where(p => p.idOrder == selectedOrder.id).FirstOrDefault();

doc.Add(new iTextSharp.text.Paragraph($"{selectedItem.Item.name}", font2));

decimal allPrice = selectedItem.quantity \* selectedItem.Item.price;

doc.Add(new iTextSharp.text.Paragraph($"{selectedItem.quantity} Х {selectedItem.Item.price} = {allPrice}", font2));

}

doc.Add(new iTextSharp.text.Paragraph($"ИТОГ = {selectedOrder.cost}", font1));

doc.Close();

}

MessageBox.Show("Документ создан");

}

* Переход на страницу редактирования заказа:

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

{

Manager.MainFrame.Navigate(new OrderAddPage((sender as Button).DataContext as Order));

}

* Страница OrderItemPage :

public partial class OrderItemPage : Page

{

private Order \_currentOrder = new Order();

public OrderItemPage(Order selectedOrder)

{

InitializeComponent();

\_currentOrder = selectedOrder;

UpLoad();

}

* Метод обновления данных:

public void UpLoad()

{

dGridItemOrder.ItemsSource = BakeryEntities.GetContext().ItemOrder.ToList().Where(p => p.idOrder == \_currentOrder.id);

}

* Кнопка добавления данных:

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

{

ItemWindow itemWindow = new ItemWindow(\_currentOrder);

itemWindow.ShowDialog();

UpLoad();

}

* Переход назад :

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

{

Manager.MainFrame.Navigate(new OrderPage());

}

* Страница ClientPage:

public partial class ClientPage : Page

{

public ClientPage()

{

InitializeComponent();

dGridClients.ItemsSource = BakeryEntities.GetContext().Client.ToList();

}

* Поисковое поле клиентов:

private void tbSearh\_TextChanged(object sender, TextChangedEventArgs e)

{

var \_currentClient = BakeryEntities.GetContext().Client.ToList();

if (tbSearh.Text != null)

{

\_currentClient = BakeryEntities.GetContext().Client.Where(p => p.name.ToLower().Contains(tbSearh.Text.ToLower()) ||

p.phone.ToLower().Contains(tbSearh.Text.ToLower()) ||

p.email.ToLower().Contains(tbSearh.Text.ToLower()) ||

p.address.ToLower().Contains(tbSearh.Text.ToLower()) ||

p.id.ToString().Contains(tbSearh.Text.ToLower()) ||

p.INN.ToString().Contains(tbSearh.Text.ToLower())).ToList();

}

dGridClients.ItemsSource = \_currentClient;

}

* Очистка поискового запроса:

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

{

tbSearh.Text = null;

dGridClients.ItemsSource = BakeryEntities.GetContext().Client.ToList();

}

* Страница OrderAddPage:

public partial class OrderAddPage : Page

{

private Order \_currentOrder = new Order();

public OrderAddPage(Order selectedOrder)

{

InitializeComponent();

if (selectedOrder != null)

\_currentOrder = selectedOrder;

DataContext = \_currentOrder;

cbClient.ItemsSource = BakeryEntities.GetContext().Client.ToList();

dpDateEnd.Text = \_currentOrder.dateEnd.ToString();

}

* Кнопка добавления заказа:

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

{

if (\_currentOrder.id == 0)

{

BakeryEntities.GetContext().Order.Add(\_currentOrder);

\_currentOrder.dateOrder = DateTime.Now;

\_currentOrder.idStatus = 1;

\_currentOrder.cost = 0;

}

try

{

\_currentOrder.idUser = 1;

\_currentOrder.cost = Convert.ToDecimal(tbCost.Text);

\_currentOrder.dateEnd = dpDateEnd.SelectedDate;

BakeryEntities.GetContext().SaveChanges();

MessageBox.Show("Информация сохранена");

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString());

}

Manager.MainFrame.Navigate(new OrderPage());

}

* Страница ClientAddPage:

public partial class ClientAddPage : Page

{

private Client \_currentClient = new Client();

public ClientAddPage()

{

InitializeComponent();

}

* Страница ItemPage:

public partial class ItemPage : Page

{

public ItemPage()

{

InitializeComponent();

dGridItems.ItemsSource = BakeryEntities.GetContext().Item.ToList();

}

}

* Добавление и сохранение клиента:

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

{

StringBuilder errors = new StringBuilder();

if (tbName.Text == "")

errors.AppendLine("Укажите имя");

if (tbPhone.Text == "")

errors.AppendLine("Укажите телефон");

if (tbINN.Text == "")

errors.AppendLine("Укажите ИНН");

if (errors.Length > 0)

{

MessageBox.Show(errors.ToString());

return;

}

\_currentClient.name = tbName.Text;

\_currentClient.phone = tbPhone.Text;

\_currentClient.email = tbEmail.Text;

\_currentClient.address = tbAddress.Text;

\_currentClient.INN = tbINN.Text;

try

{

BakeryEntities.GetContext().Client.Add(\_currentClient);

BakeryEntities.GetContext().SaveChanges();

MessageBox.Show("Данные сохранены");

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString());

}

Manager.MainFrame.Navigate(new ClientPage());

}

* Переход на страницу клиентов:

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

{

Manager.MainFrame.Navigate(new ClientPage());

}

}

ПРИЛОЖЕНИЕ Б – Флэш - накопитель с программой