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 управления заказами и обслуживания клиентов в пиццерии

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

ККОО.ДП2188.000

|  |  |  |
| --- | --- | --- |
| Студент | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2025 г. | Михеев В. А. |
| Руководитель проекта | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2025 г. | Бунаков П. Ю. |
| Консультант по экономическому разделу | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2025 г. | Караваев А.В. |
| Нормоконтролер | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2025 г. | Грушникова Т.Н. |
| Рецензент | «\_\_\_\_»\_\_\_\_\_\_\_\_\_2025 г. | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

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

2025

Министерство образования Московской области

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

|  |  |
| --- | --- |
|  | «Утверждаю»  Зам. директора по УР  \_\_\_\_\_\_\_\_\_\_ Э.Б. Ромашкина  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2025 г. |

ЗАДАНИЕ

на дипломный проект по специальности 09.02.07 «Информационные системы и программирование» студенту очного отделения, группы 42111

Кузнецову Николаю Сергеевичу

Тема проекта Разработка информационной системы магазина канцелярских товаров

1. Постановка задачи:

Разработать на языке C# в среде MS Visual Studio 2017 информационную систему оптимизирования работы магазина канцелярских товаров с использованием технологии Windows Forms.

Информационная система должны решать следующие задачи:

- Поисковая система товара;

- Фильтрация по различным критериям;

- Возможность оперативного добавления и изменения данных.

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

— Информация о товаре;

— Информация о контрагентах;

— Информация о категориях товара;

— Журнал операций;

— Журнал учета остатков товара;

— Журнал изменения цены поступившего товара.

Для возможности просмотра информации о товаре необходимы поля: артикул, название, цена закупочная, цена розничная, остаток товара, описание товара.

Для возможности просмотра информации о контрагентах необходимы поля: название организации, адрес, телефон, описание организации.

Для возможности просмотра информации о категориях товара необходимы поля: название.

Для возможности просмотра журнала операций необходимы поля: название, название товара, артикул, цена товара, количество товара, номер документа, название операции, название организации, дата, время.

Для возможности просмотра журнала учета остатков необходимы поля: название, артикул, номер документа, остаток товара, дата, время.

Для возможности просмотра журнала изменения цены поступившего товара остатков необходимы поля: название, артикул, цена закупочная, цена розничная, дата.

2. Состав и объем дипломного проекта

2.1 Пояснительная записка, содержащая следующие обязательные разделы:

Введение

* Основная часть
  + Разработка системного проекта
    - Описание предметной области
    - Анализ предметной области
    - Техническое задание
    - Описание используемых технологий
  + Проектирование информационной системы
    - Функциональное моделирование
    - Проектирование интерфейсов приложения
  + Разработка информационной системы
    - Модель данных
    - Структура проекта
  + Оценка качества программного обеспечения

2 Экономический раздел

3 Охрана труда 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**:**

Тут должны быть перечислены подразделы экономического раздела. Шрифт TimesNewRoman 14 пт

5. Содержание раздела по охране труда:

Охрана труда 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. Федорова Г.Н.. Осуществление интеграции программных модулей. : учебник для студ. учреждений сред. Проф. Образования / Г.Н. Федорова. – 3-е изд., исправленное. – М.: Издательский центр «Академия», 2023.
6. Литература по охране труда
7. Литература по экономике

Дата выдачи задания « 07 » апреля 2025 г.

Срок окончания дипломного проекта «14» июня 2025 г.

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

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

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

Задание утверждено на заседании цикловой комиссии

Протокол № 7 от «12» марта 2025 г.

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

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2025 года.

Содержание

|  |  |  |  |
| --- | --- | --- | --- |
| Введение | | |  |
| 1 | Основная часть | | 9 |
|  | 1.1 | Постановка задачи |  |
|  | 1.2 | Описание использованных технологий | 10 |
|  | 1.3 | Проектирование базы данных | 14 |
|  | 1.4 | Проектирование приложения | 19 |
|  | 1.5 | Разработка приложения | 23 |
|  | 1.6 | Тестирование | 34 |
| 2 | Экономический раздел | | 37 |
|  | 2.1 | Понятие себестоимости |  |
|  | 2.2 | Затраты в составе себестоимости | 38 |
|  | 2.3 | Расчет себестоимости информационной системы | 41 |
| 3 Требования по технике безопасности при работе с вычислительной техникой | | | 48 |
|  | Выводы и заключение | | 52 |
|  | Список литературы | | 53 |
|  | Приложение А – Код программы | | 54 |
|  | Приложение Б – Диск с программой | | 57 |

В современном мире сфера общественного питания сталкивается с необходимостью оптимизации внутренних процессов для повышения эффективности работы заведений. Одним из ключевых аспектов успешного функционирования пиццерии является грамотное управление меню, контроль заказов и анализ статистики, что позволяет минимизировать издержки и улучшить качество обслуживания клиентов. Однако во многих заведениях эти процессы до сих пор ведутся вручную или с использованием устаревших систем, что приводит к ошибкам, замедлению работы и потере данных.

Актуальность данной работы заключается в разработке desktop-приложения, предназначенного для автоматизации рабочих процессов сотрудников пиццерии. Программа позволяет упростить добавление и редактирование продуктов в меню, отслеживать статусы заказов, оперативно завершать их, а также анализировать статистику по продажам, популярности блюд и активности клиентов. Это способствует снижению нагрузки на персонал, уменьшению количества ошибок и повышению общей эффективности работы заведения.

Целью дипломного проекта является создание удобного и функционального desktop-приложения для сотрудников пиццерии, которое позволит автоматизировать ключевые бизнес-процессы. Основными задачами стали разработка интуитивно понятного интерфейса, реализация механизма управления меню (добавление, редактирование и удаление продуктов), создание системы контроля заказов (отслеживание статусов и завершение), а также внедрение модуля аналитики для оценки эффективности работы заведения.

Приложение предоставляет сотрудникам пиццерии удобный инструмент для управления ассортиментом. Они могут добавлять новые позиции в меню, изменять их описание и стоимость, а также скрывать временно недоступные блюда. Система заказов позволяет в реальном времени отслеживать поступившие запросы от клиентов, фиксировать их выполнение и закрывать после завершения.

Особое внимание уделено модулю статистики, который собирает и анализирует данные о продажах, популярности продуктов и поведении клиентов. Это помогает руководству принимать обоснованные решения по корректировке меню, оптимизации закупок и улучшению сервиса.

Разработанное desktop-приложение представляет собой современное решение для автоматизации работы пиццерии. Оно не только упрощает ежедневные задачи сотрудников, но и способствует повышению эффективности бизнеса за счёт анализа ключевых показателей. Внедрение подобной системы позволяет сократить время обслуживания клиентов, минимизировать ошибки, связанные с ручным вводом данных, и в целом улучшить качество работы заведения.

Для достижения цели были выделены следующие задачи:

* Проектирование базы данных в соответствии с требованиями к функционалу системы;
* Разработка базы данных;
* Разработка функционала информационной системы в соответствии с требованиями к программному продукту;
* Создание пользовательского интерфейса, обеспечивающего интуитивно понятное взаимодействие с системой;

ОСНОВНАЯ ЧАСТЬ

1. Постановка задачи

Требуется разработать desktop-приложение, предназначенное для сотрудников пиццерии, которое позволит эффективно управлять меню, отслеживать заказы клиентов и анализировать статистику. Приложение должно обеспечивать удобный интерфейс для работы с продуктами, заказами и данными о клиентах, а также предоставлять инструменты для визуализации статистики.

Главный экран приложения содержит боковое меню с навигационными кнопками, позволяющими переходить между разделами системы. На домашней странице отображается краткая статистика за текущий день, включая количество новых заказов, активных заказов и выручку. Это позволяет сотрудникам быстро оценить текущую загрузку пиццерии.

В разделе **"Склад продуктов"** представлен каталог всех доступных продуктов. Пользователи могут искать товары по названию или типу с помощью поисковой строки. Продукты отображаются в виде карточек, каждая из которых содержит основную информацию: название, тип, количество на складе и цену. Для удобства управления на карточках предусмотрены кнопки редактирования и удаления продукта.

Раздел **"Заказы"** предоставляет возможность просмотра и управления заказами клиентов. Для фильтрации заказов по дате используется календарь. Заказы выводятся в виде карточек с краткой информацией: номер заказа, статус, сумма и время оформления. На каждой карточке есть кнопки для печати чека и перехода к детальному просмотру заказа. На странице подробной информации о заказе можно завершить заказ, а также повторно распечатать чек.

Для анализа эффективности работы пиццерии в разделе **"Статистика"** представлены данные в виде диаграмм (круговых и столбчатых) и текстовых отчетов. Статистика включает информацию по клиентам (частота заказов, предпочтения), продуктам (популярность, остатки на складе), заказам (динамика продаж) и общей прибыли. Это позволяет руководству принимать обоснованные решения по управлению ассортиментом и маркетинговым активностям.

В разделе **"Профиль"** сотрудник может просматривать и редактировать свои личные данные: ФИО, фотографию, почту, номер телефона и место работы. Также здесь доступны функции смены пароля и выхода из системы.

Приложение обеспечивает удобный и интуитивно понятный интерфейс, что позволяет сотрудникам пиццерии быстро осваивать систему и эффективно выполнять свои задачи. Автоматизация процессов учета заказов и управления складом сокращает время обработки данных и минимизирует ошибки, связанные с ручным вводом информации.

1.2 Описание использованных технологий

В данном проекте реализовано desktop-приложение для сотрудников пиццерии, построенное на архитектуре **MVVM (Model-View-ViewModel)**, что обеспечивает четкое разделение логики представления, бизнес-логики и данных. Приложение позволяет управлять продуктами, обрабатывать заказы и анализировать статистику, предоставляя удобный интерфейс для работы.

**Слой представления (View)** отвечает за пользовательский интерфейс и взаимодействие с сотрудником. Он включает в себя главное меню с навигационными кнопками, страницы для работы с продуктами, заказами, статистикой и профилем. Все элементы интерфейса связаны с **ViewModel**, которая обрабатывает действия пользователя и обновляет данные в реальном времени.

**Слой ViewModel** выступает промежуточным звеном между **View** и бизнес-логикой. Он преобразует данные из сервисов в удобный для отображения формат, обрабатывает пользовательские команды (например, добавление продукта или завершение заказа) и обеспечивает реактивность интерфейса. Благодаря этому достигается высокая степень тестируемости и гибкости при изменении UI.

**Бизнес-логика приложения** вынесена в отдельные **сервисы**, которые выполняют все ключевые операции: управление продуктами, обработку заказов, формирование статистики и работу с профилями сотрудников. В отличие от классической трехслойной архитектуры, доступ к данным (запросы к базе) также осуществляется непосредственно в сервисах, что упрощает структуру кода без ущерба для функциональности.

На рисунке 1 представлена схема архитектуры MVVM.

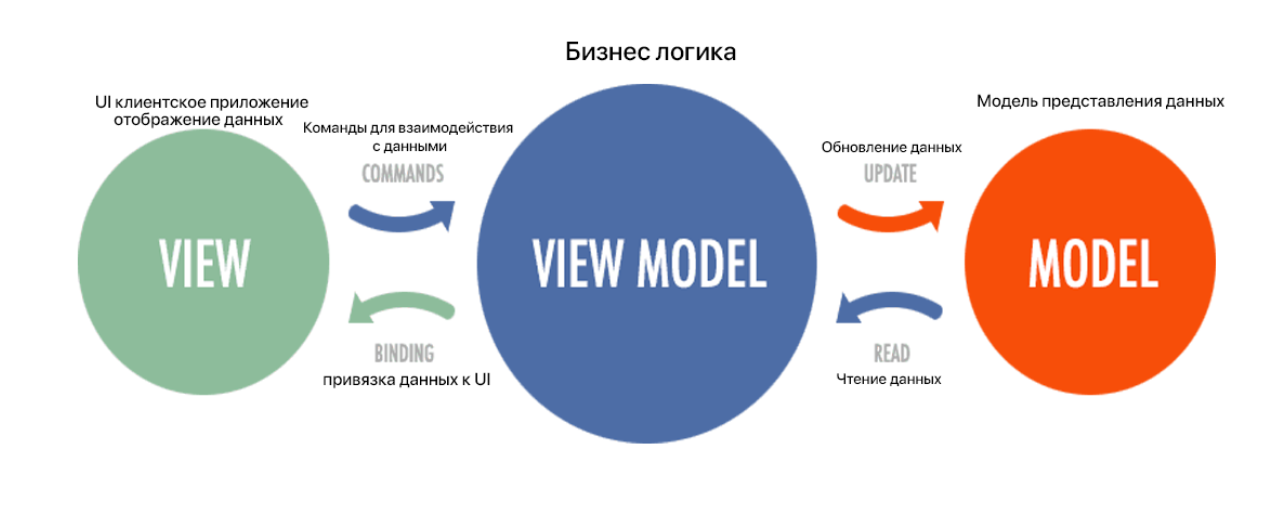


Рисунок 1 – Схема архитектуры MVVM

**SQL Server** – это профессиональная система управления базами данных, широко используемая в проектах любого масштаба: от небольших сервисов до сложных корпоративных решений. Благодаря высокой производительности, встроенным средствам защиты данных и тесной интеграции с экосистемой Microsoft, SQL Server остается одним из ключевых инструментов для разработчиков и IT-специалистов.

В рамках этого проекта выбор SQL Server обусловлен его надежностью, мощными механизмами обработки транзакций и удобными инструментами администрирования. Система эффективно справляется с хранением и управлением данными клиентов, арендой автомобилей, финансовыми операциями и другими бизнес-процессами. Кроме того, SQL Server оптимизирован для работы с высокой нагрузкой, поддерживая параллельное выполнение запросов. Это позволяет приложению стабильно работать даже при большом количестве пользователей, обеспечивая быстрый отклик и бесперебойную работу, что напрямую влияет на удобство и лояльность клиентов.

Для работы с СУБД было выбрано приложение Microsoft SQL Server Management Studio.

**Microsoft SQL Server Management Studio (SSMS)** – это многофункциональная среда для работы с базами данных, которая предоставляет разработчикам, администраторам и аналитикам удобные инструменты для проектирования, настройки, мониторинга и оптимизации SQL Server.

Разработка программной части информационной системы осуществлялась в Microsoft Visual Studio 2022. Эта интегрированная среда разработки предоставляет обширный набор инструментов для создания приложений различного типа. В ходе работы были особенно полезны многофункциональность платформы, интуитивно понятный интерфейс и широкие возможности, которые существенно ускоряют процесс разработки.

Клиентская часть приложения разрабатывалась в Microsoft Visual Studio 2022. Выбор этой среды обусловлен её высокой производительностью, удобством работы с различными технологиями и поддержкой современных инструментов разработки. Visual Studio 2022 обеспечивает мощную функциональность, встроенные средства отладки и гибкие настройки, что позволяет эффективно реализовывать клиентские решения и адаптировать их под конкретные требования проекта.

Для разработки пользовательского интерфейса был выбран WPF (Windows Presentation Foundation). Этот фреймворк предоставляет мощные инструменты для создания современных десктоп-приложений с богатой графикой и гибкими возможностями кастомизации. WPF был выбран благодаря его тесной интеграции с платформой .NET, что обеспечивает высокую производительность, надежность и удобство разработки. Использование WPF позволило реализовать сложные интерфейсы с поддержкой анимаций, стилей и шаблонов, обеспечивая при этом отзывчивость и плавность работы. Технология XAML значительно упростила декларативное описание интерфейса, а встроенные механизмы привязки данных (data binding) и команд (commands) помогли эффективно разделить логику приложения и его представление. WPF доказал свою эффективность в создании масштабируемых и поддерживаемых решений для настольных приложений.

В качестве основного языка программирования был выбран C#, так как он хорошо подходит для разработки десктоп-приложений на платформе .NET. Наличие опыта работы с этим языком позволило ускорить процесс разработки и минимизировать время на освоение новых технологий. Благодаря знакомству с синтаксисом и инструментарием C#, удавалось эффективнее решать задачи, связанные с реализацией бизнес-логики и взаимодействием с интерфейсом WPF. Язык предоставляет строгую типизацию, современные возможности программирования и управляемый код, что повышает надежность и безопасность приложения. Кроме того, богатая стандартная библиотека .NET и поддержка со стороны Visual Studio 2022 сделали C# оптимальным выбором для данного проекта.

1.3 Проектирование базы данных

Основной задачей при проектировании базы данных являлось создание оптимальной структуры для надежного хранения и быстрого доступа к информации, необходимой для работы приложения. База данных построена в виде набора связанных таблиц, структура которых тщательно продумана с учетом всех функциональных требований системы. Разработанная схема данных обеспечивает четкую организацию сведений о товарах, пользователях, заказах и транзакциях, что позволяет эффективно реализовывать основной функционал платформы и гарантировать удобство взаимодействия пользователей с сервисом.

Модель «Сотрудник»:

Модель «Сотрудник» представляет собой основную сущность системы. Каждый сотрудник имеет уникальный идентификатор, который используется для идентификации в системе. В дополнение к идентификатору, в модели хранятся основные личные данные, такие как электронная почта, имя, фамилия, телефон, фотография профиля, место работы и логин. Пароль пользователя хранится в зашифрованном виде для обеспечения безопасности.

Модель «Заказ»:

Модель «Заказ» представляет собой информацию о заказе, который клиент размещает в системе. Каждый заказ имеет уникальный идентификатор, дату заказа, адрес доставки, статус заказа и стоимость заказа. Дополнительные атрибуты включают заметки для доставки, точное и примерное время доставки, а также метод оплаты заказа. В модели хранится информация о клиенте сделавший заказ и продукты содержащиеся в заказе.

Модель «Продукт»:

Модель «Продукт» представляет собой информацию о товаре, который доступен для заказа в системе. Каждый продукт имеет уникальный идентификатор, название, тип и стоимость. Дополнительно модель включает описание товара, его вес, калорийность и фотографию.

Модель "Клиент":

Модель "Клиент" Модель «Клиент» содержит информацию о пользователе, который совершает заказы в системе. Каждый клиент имеет уникальный идентификатор, имя, фамилия, отчество, контактный телефон и дату регистрации.

В рамках объектно-ориентированное проектирование разрабатываемой системы было произведено проектирование базы данных. Все проведено с учетом необходимости обеспечения эффективного хранения данных, поддержки функционала desktop-приложения для удобства сотрудников.

Все поля спроектированной базы данных описаны в таблице 1 – в словаре данных.

Словарь данных – структурированный набор терминов и их определений, используемых для описания данных, используемых в системе или проекте.

Таблица 1 – Словарь данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| City | | | | |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
| Первичный | id | int | Да | Уникальный идентификатор |
|  | city\_name | Varchar | Да | Название города |
| Customers | | | | |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
| Первичный | id | int | Да | Уникальный идентификатор |

Продолжение таблицы 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Ключ | Поле | | Тип данных | Обязательное | | Примечание |
|  | full\_name | | Nvarchar | Да | | ФИО |
|  | phone | | Nvarchar | Да | | Номер телефона |
|  | registration\_date | | Datetime | Да | | Время регистрации |
| Employee | | | | | | |
| Ключ | | Поле | Тип данных | Обязательное | Примечание | |
| Первичный | | id | int | Да | Уникальный идентификатор | |
|  | | login | Varchar | Да | Логин | |
|  | | password | Varchar | Да | Хешированный пароль | |
| Внешний | | workplace\_id | int | Нет | Идентификатор места работы | |
| Внешний | | role\_id | int | Нет | Идентификатор роли сотрудника | |
|  | | full\_name | Varchar | Да | Имя | |
|  | | phone | Varchar | Да | Номер телефона | |
|  | | email | Varchar | Да | Почта | |
|  | | photo | Varbinary | Нет | Фото | |
| Employee\_role | | | | | | |
| Ключ | | Поле | Тип данных | Обязательное | Примечание | |
| Первичный | | id | int | Да | Уникальный идентификатор | |
|  | | role\_name | Varchar | Да | Имя | |
| Order\_items | | | | | | |
| Ключ | | Поле | Тип данных | Обязательное | Примечание | |
| Первичный | | id | int | Да | Уникальный идентификатор | |

Продолжение таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
| Внешний | order\_id | int | Нет | Идентификатор заказа |
| Внешний | product\_id | int | Нет | Идентификатор продукта |
| Orders | | | | |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
| Первичный | id | int | Да | Уникальный идентификатор |
| Внешний | customer\_id | int | Да | Идентификатор клиента |
|  | order\_date | Datetime | Да | Время заказа |
|  | delivery\_address | Varchar | Да | Адрес заказа |
|  | total\_amount | Decimal | Да | Стоимость |
|  | payment\_method | Nvarchar | Да | Метод оплаты |
|  | order\_status | Nvarchar | Да | Статус |
|  | delivery\_notes | Text | Нет | Заметки заказа |
|  | estimated\_delivery\_time | Datetime | Нет | Примерное время доставки |
|  | actual\_delivery\_time | Datetime | Нет | Точное время доставки |
| Products | | | | |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
| Первичный | id | int | Да | Уникальный идентификатор |
|  | product\_name | Nvarchar | Да | Имя |
|  | type | Nvarchar | Да | Тип |
|  | cost | Decimal | Да | Стоимость |

Продолжение таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
|  | description | Nvarchar | Нет | Описание |
|  | wheight | Decimal | Да | Вес |
|  | ccal | Decimal | Да | Калории |
|  | photo | Varbinary | Нет | Фото |
| Street\_type | | | | |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
| Первичный | id | int | Да | Уникальный идентификатор |
|  | name | Nvarchar | Да | Имя |
|  | short\_name | Nvarchar | Нет | Сокращенное имя |
| Workplace | | | | |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
| Первичный | id | int | Да | Уникальный идентификатор |
| Внешний | city\_id | int | Да | Идентификатор города |
| Внешний | street\_type\_id | int | Да | Идентификатор типа улицы |
|  | street | Varchar | Да | Улица |
|  | home | Varchar | Да | Дом |
|  | apartment | Varchar | Нет | Квартира |
|  | address | Varchar | Да | Адрес |

Исходя из предметной области была построена ER-диаграмма для базы данных, предназначенной для эффективного управления обменом продуктами питания.

ER-диаграмма (сокращение от Entity-Relationship diagram) – это графическое представление структуры базы данных, которое помогает моделировать и анализировать взаимосвязи между сущностями в системе. Этот метод дает возможность визуализировать основные сущности, их атрибуты и связи между ними. На рисунке 2 изображена ER-диаграмма для базы данных.

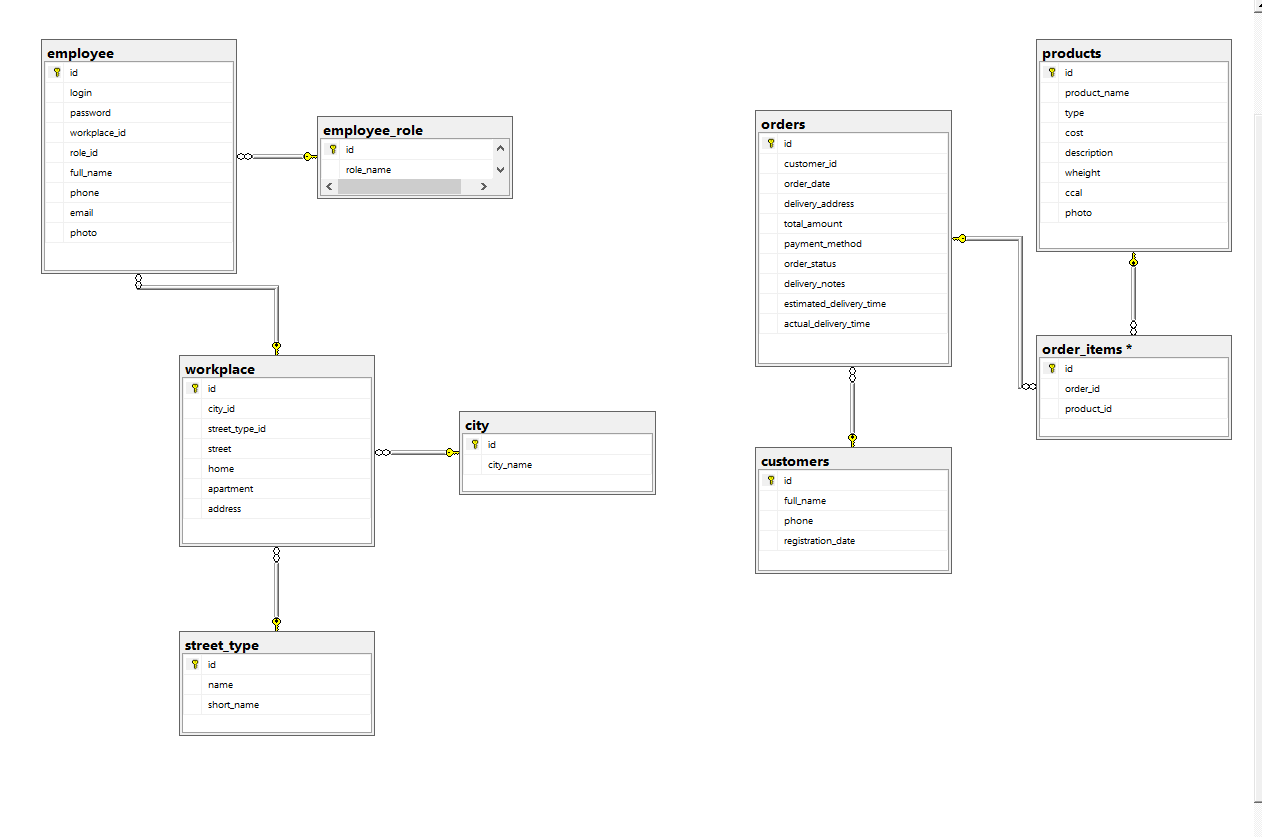


Рисунок 2 – ER диаграмма

1.4 Проектирование приложения

Перед разработкой проекта были спроектированы следующие диаграммы, каждая из которых играет важную роль в описании функциональности и структуры системы.

Диаграмма вариантов использования (прецендентов) помогла определить основные действия, которые пользователи могут выполнять в системе. Это включает в себя описание функциональности системы с точки зрения ее пользователей и предоставляет общее представление о взаимодействии между системой и ее окружением. Спроектированная диаграмма представлена на рисунке 3.

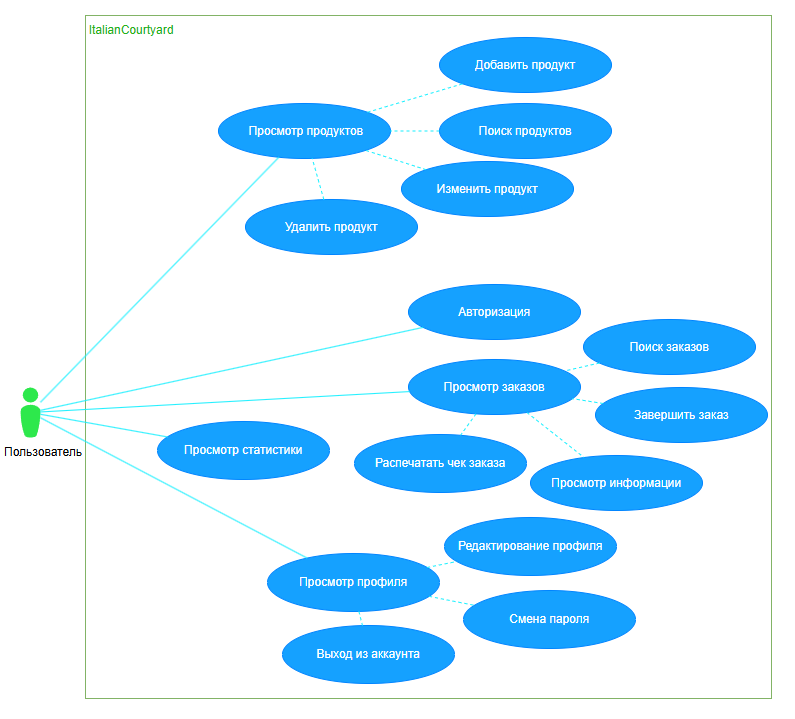


Рисунок 3 – Диаграмма вариантов использования

Диаграмма последовательности разработана для иллюстрации последовательности взаимодействия между объектами в системе в конкретном сценарии использования. Данная диаграмма помогает описать, как объекты взаимодействуют друг с другом во времени и какие сообщения или операции они обмениваются в рамках конкретного сценария. Разработанная диаграмма представлена на рисунке 4.

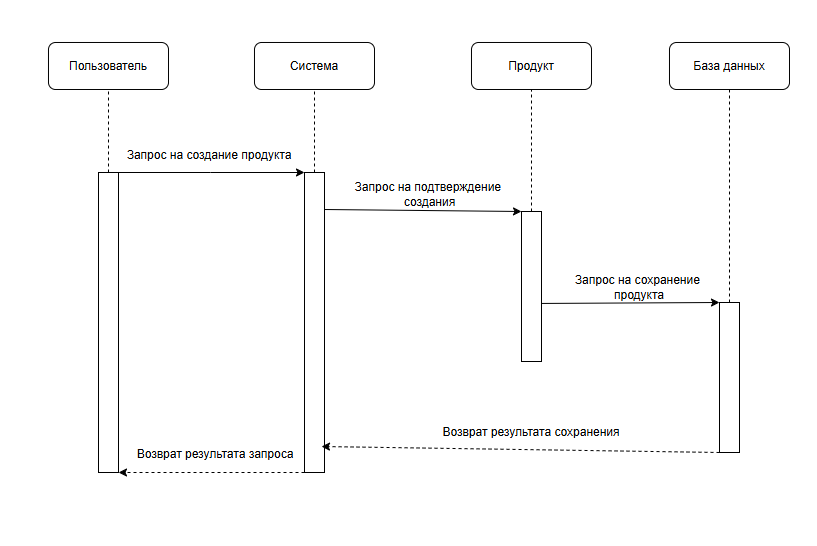


Рисунок 4 – Диаграмма последовательности

Диаграмма деятельности была использована для визуализации последовательности действий или процессов в системе. Это позволило описать порядок выполнения операций или задач в системе и выделить основные этапы взаимодействия пользователей с системой. Разработанная диаграмма представлена на рисунке 5.

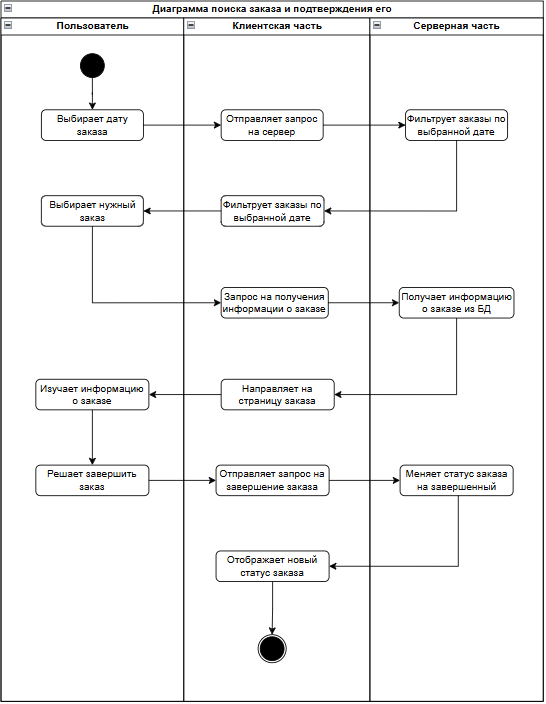


Рисунок 5 – Диаграмма деятельности

1.5 Разработка приложения

В рамках проекта разработаны модели представления данных, которые являются важной частью информационной системы. Эти модели созданы с учетом основных бизнес-процессов и данных, необходимых для работы системы. На рисунке 6 представлены классы моделей данных.

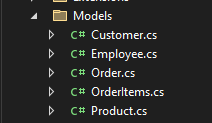


Рисунок 6 – Доменные модели

Разрабатываемый проект должен реализовывать такой функционал как:

* добавление данных;
* редактирование данных;
* удаление данных;

Обработка данных для списка выше представлена в Приложении А.

В информационной системе предусмотрено 10 страниц, которые описаны в Таблице 2:

Таблица 2 – Страницы приложения

|  |  |  |
| --- | --- | --- |
| Название | Основное назначение | Функционал |
| AuthorizationWindow | Окно авторизации в приложении | При вводе корректных данных во все необходимые поля происходит успешная авторизация пользователя в систему |
| DashboardPage | Домашняя страница приложения | Возможность просмотра краткой статистики за текущей день |

Продолжение таблицы 2

|  |  |  |
| --- | --- | --- |
| Название | Основное назначение | Функционал |
| ProductWarehousePage | Страница склада продуктов | Возможность просматривать и осуществлять поиск продуктов, редактировать и удалять их |
| EditProductPage | Страница редактирования продукта | При вводе корректных данных во все необходимые поля происходит изменение данных продукта в базе данных |
| AddProductPage | Страница добавления продукта | При вводе корректных данных во все необходимые поля происходит добавление продукта в базу данных |
| OrderPage | Страница со списком всех заказов за выбранную дату | Возможность просмотра заказов за выбранных период, переход на страницу заказа и печати чека |
| OrderInfoPage | Страница просмотра информации по заказу | Возможность просмотра детальной информации по заказу, возможность завершить заказ и распечатать чек по заказу |
| StatisticPage | Страница для вывода статистики по заказам, продуктам, клиентам и выручки | Возможность просмотра детальной статистики по заказам, продуктам, клиентам и выручки |
| ProfilePage | Страница профиля пользователя | При вводе корректных данных во все необходимые поля происходит изменение данных профиля в базе данных |
| ChangePasswordPage | Страница изменения пароля | При вводе корректных данных во все необходимые поля происходит изменения пароля в базе данных |

Таблица 3 – Сервисы в приложении

|  |  |
| --- | --- |
| Название | Описание функционала |
| AuthorizationService | Управляет процессами аутентификации пользователей в приложении. Обрабатывает запросы для входа в систему пользователей и выхода из системы. |
| CustomerService | Предназначен для управления запросами, связанными с клиентами. Он обрабатывает запросы для получения информации о клиентах и для формирования статистики по клиентам. |
| EmployeeService | Предназначен для управления запросами, связанными с сотрудниками в приложении. Обрабатывает различные запросы для получения сотрудников, смены пароля, смены фотографии профиля и обновления данных в профиле. |
| OrderService | Предназначен для управления запросами, связанными с заказами клиентов. Обрабатывает запросы для получения заказов, завершения заказа, печати чека и формирование статистики по заказам. |
| ProductService | Предназначен для управления запросами, связанными с заказами продуктами. Обрабатывает запросы для получения продуктов, поиск продукта, удаления продукта, сохранения продукта, обновление продукта и формирование статистики по продуктам. |

Для начала работы с приложением необходимо открыть стартовое окно авторизации, изображенную на рисунке 7.

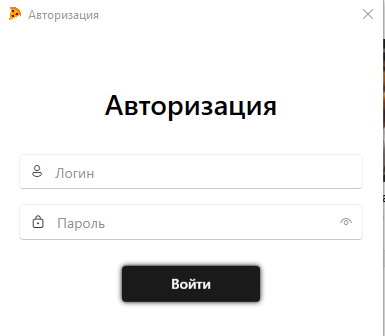


Рисунок 7 – окно авторизации

После ввода логина и пароля нажмите на кнопку “Войти”. После входа в приложение откроется домашняя страница с краткой статистикой за сегодняшний день, которая изображена на рисунке 8

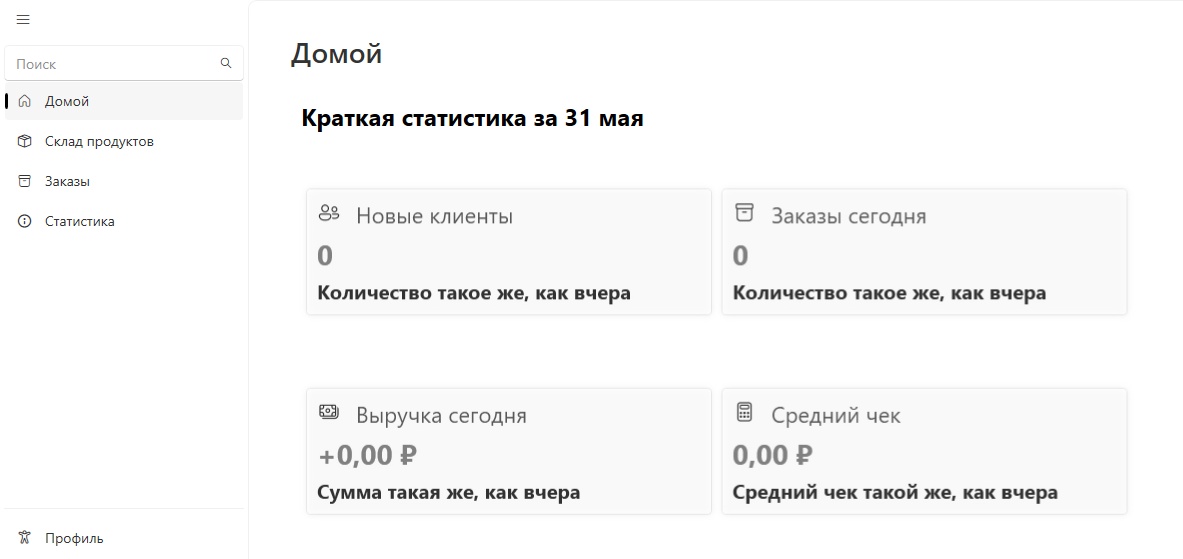


Рисунок 8 – Домашняя страница

После нажатия кнопки “Склад продуктов” откроется страница со всеми продуктами и поиском по продуктам. (Рисунок 9)

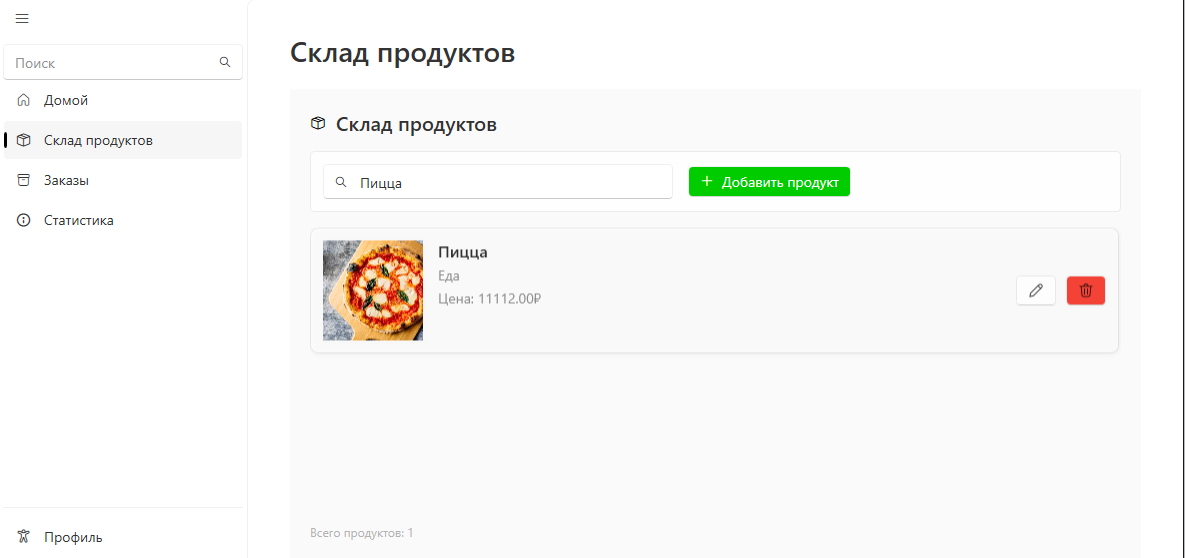


Рисунок 9 – Страница со всеми продуктами

Пользователь может нажать кнопку “+ Добавить продукт” и перейти на страницу создания нового продукта. (Рисунок 10)

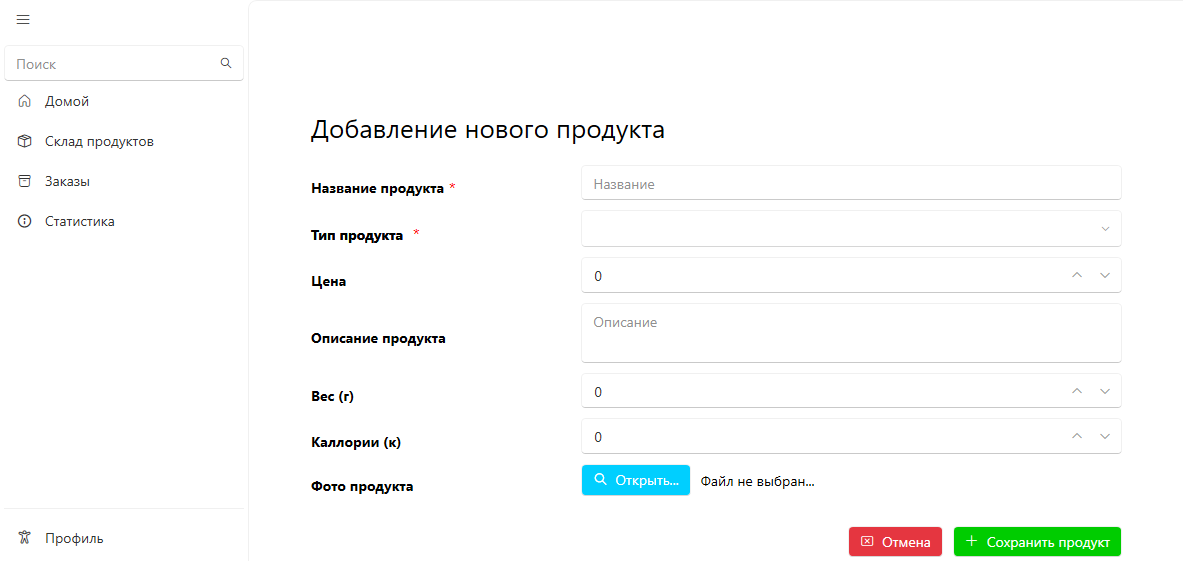


Рисунок 10 – Страница добавления нового продукта

Пользователь может нажать на кнопку “С карандашом” и перейти на страницу редактирования продукта. (Рисунок 11)

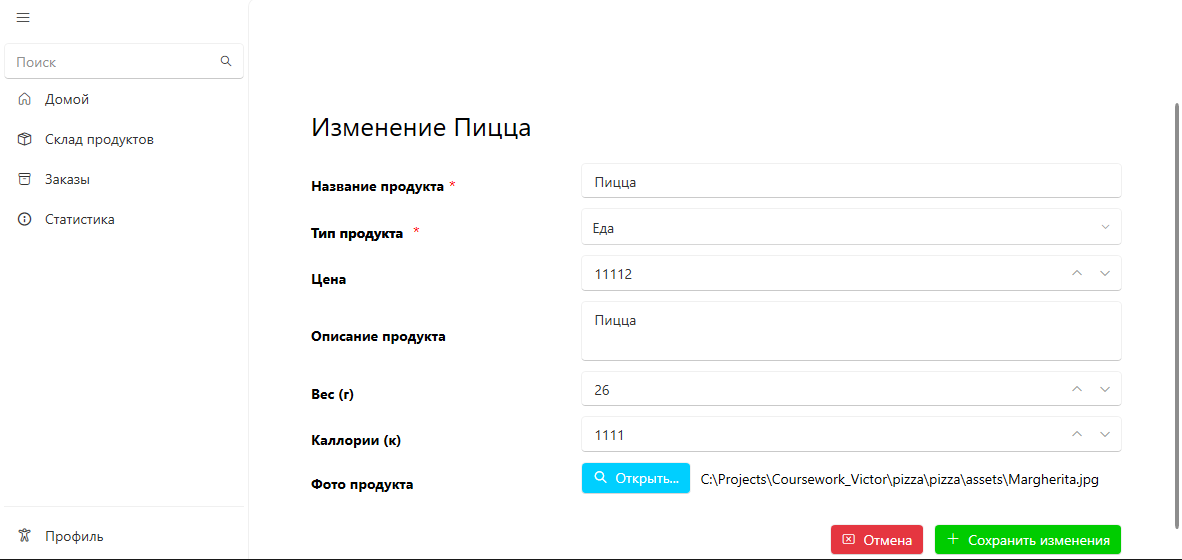


Рисунок 11 – Редактирование продукта

Если пользователь нажмет на “Мусорку”, то вылезет окно подтверждающие удаление продукта. (Рисунок 12)

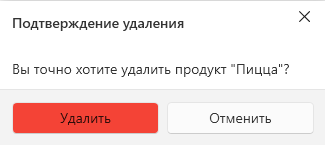


Рисунок 12 – Удаление продукта

Пользователь может нажать кнопку “Заказы” в меню и перейти на страницу заказов. (Рисунок 13)

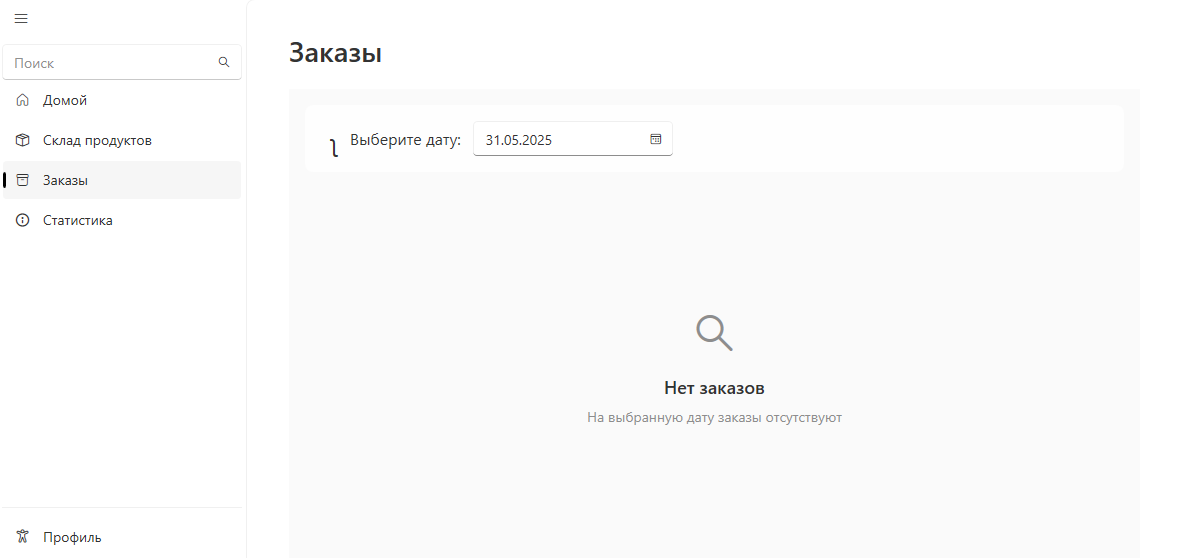


Рисунок 13 – Страница заказов

На странице с заказами можно использовать календарь чтобы найти заказы в нужный день. (Рисунок 14)

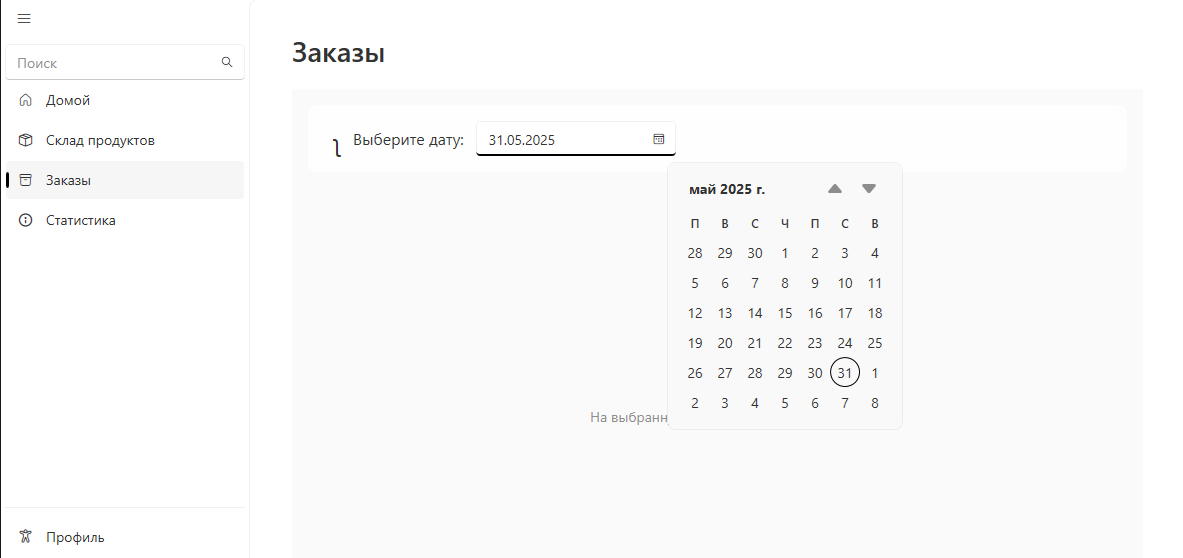


Рисунок 14 – Использование календаря

Пользователь может перейти на страницу детальной информации по заказу нажав соответствующую кнопку “Детали заказа” (Рисунок 15)

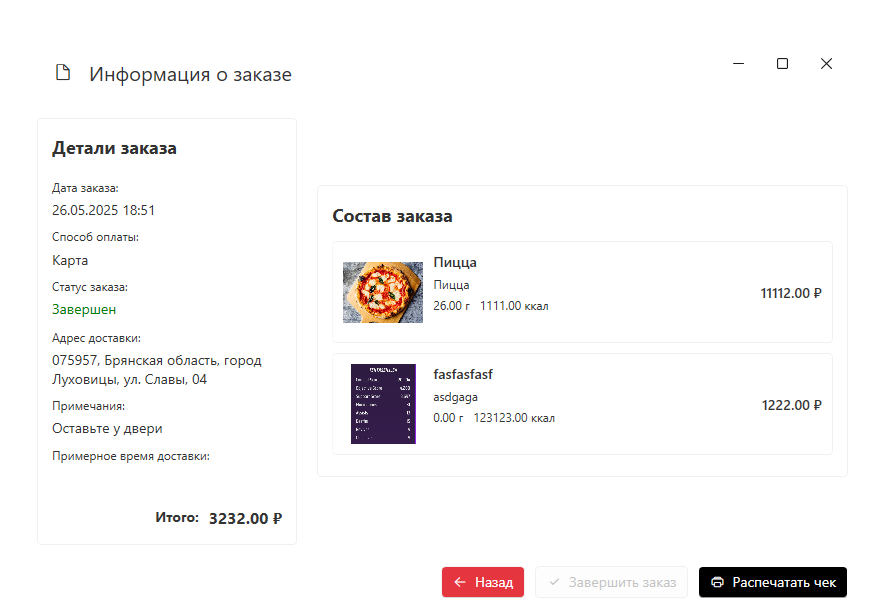


Рисунок 15 – Страница информации о заказе

Пользователь может нажать на кнопку “Завершить заказ”, после нажатия статус заказа изменится на “Завершен”. (Рисунок 16)

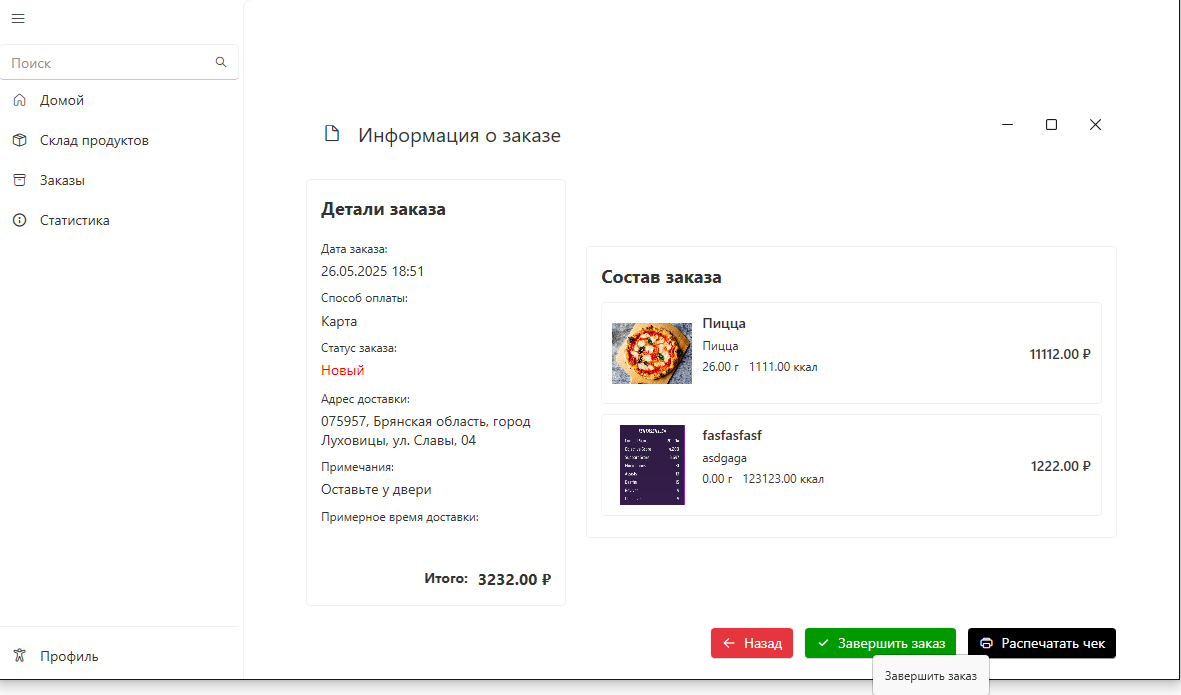


Рисунок 16 – Завершение заказа

Также есть возможность распечатать чек заказа после нажатия кнопки “Распечатать чек”(Рисунок 17)

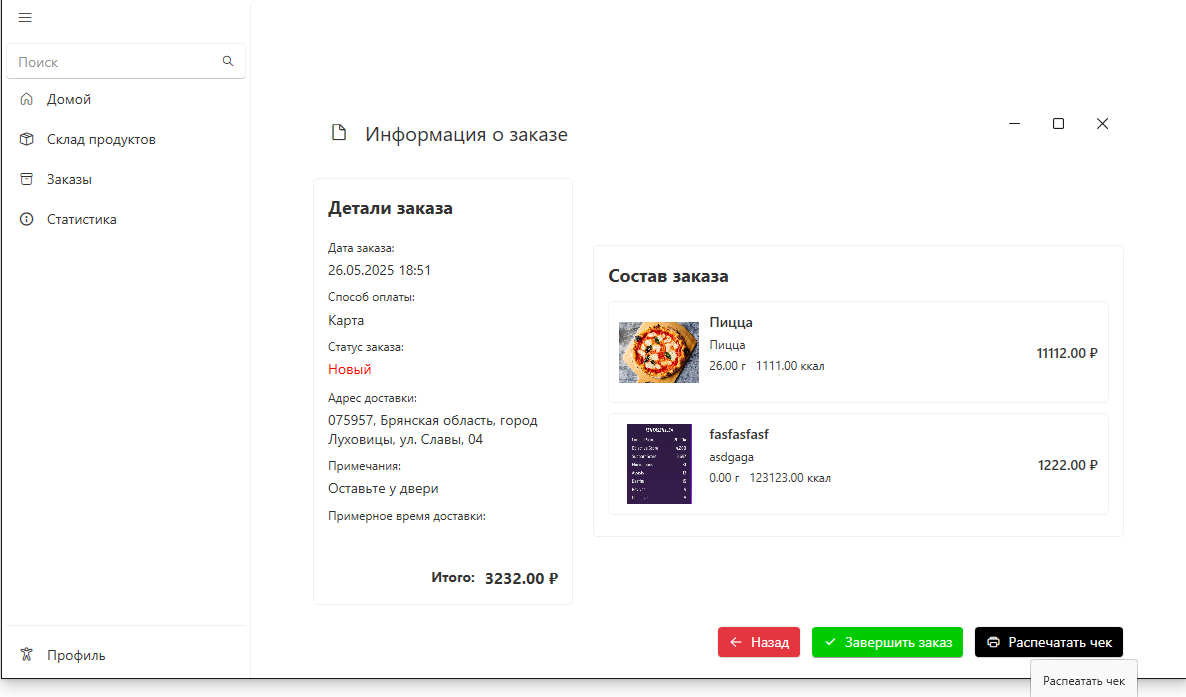


Рисунок 17 – Печать чека

На странице со всеми заказами пользователь также может нажать кнопку “Распечатать чек”.

Пользователь может нажать кнопку “Статистика” в меню и перейти на страницу статистики, где он может посмотреть всю статистике по заказам, продуктам, клиентам и выручке. (Рисунок 18)

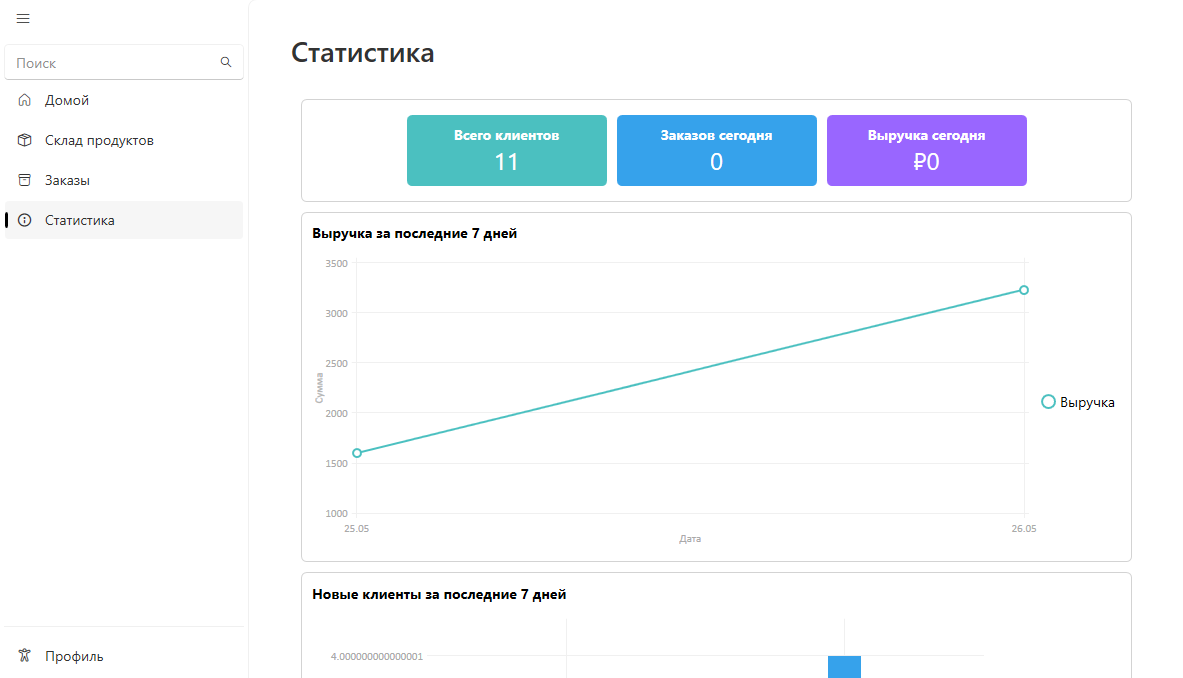


Рисунок 18 – Страница статистики

Пользователь может нажать кнопку “Профиль” в меню и перейти на страницу своего профиля. (Рисунок 19)

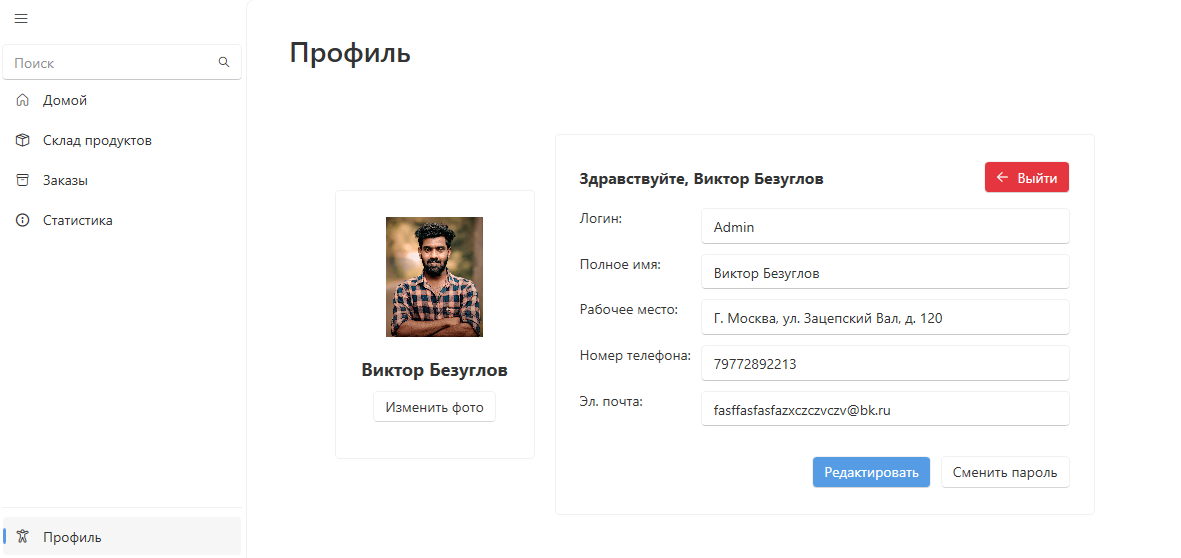


Рисунок 19 – Страница профиля

На странице профиля можно изменить пароль и почту пользователя. (Рисунок 20)

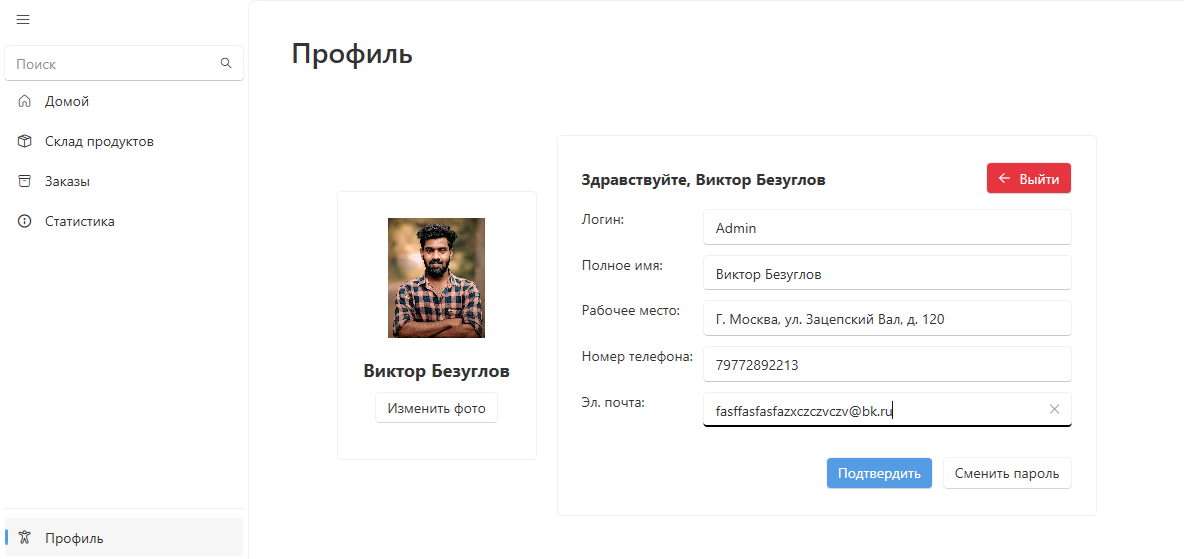


Рисунок 20 – Изменение почты и пароля

Из страницы профиля пользователь может открыть страницу со сменой пароля нажав кнопку “Сменить пароль” (Рисунок 21)

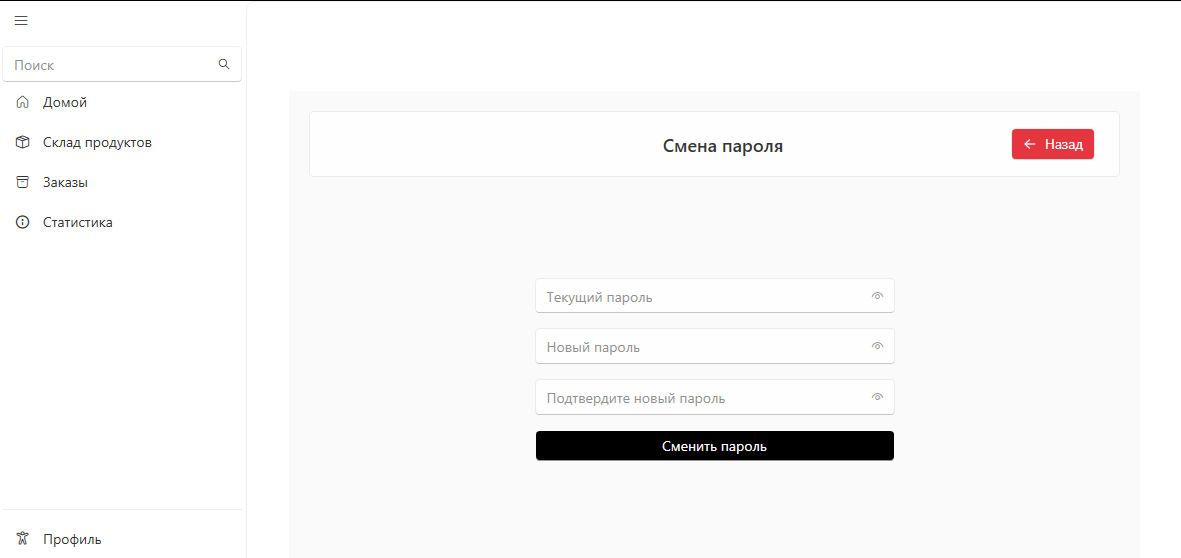


Рисунок 21 – Страница со сменой пароля пользователя

1.6 Тестирование

Для обеспечения надежной работы проекта было проведено тестирование функциональности с использованием инструмента Xunit. (Рисунок 22) Xunit предоставляет удобную библиотеку для тестирования, что позволяет эффективно проверить работоспособность и соответствие ожидаемым результатам.

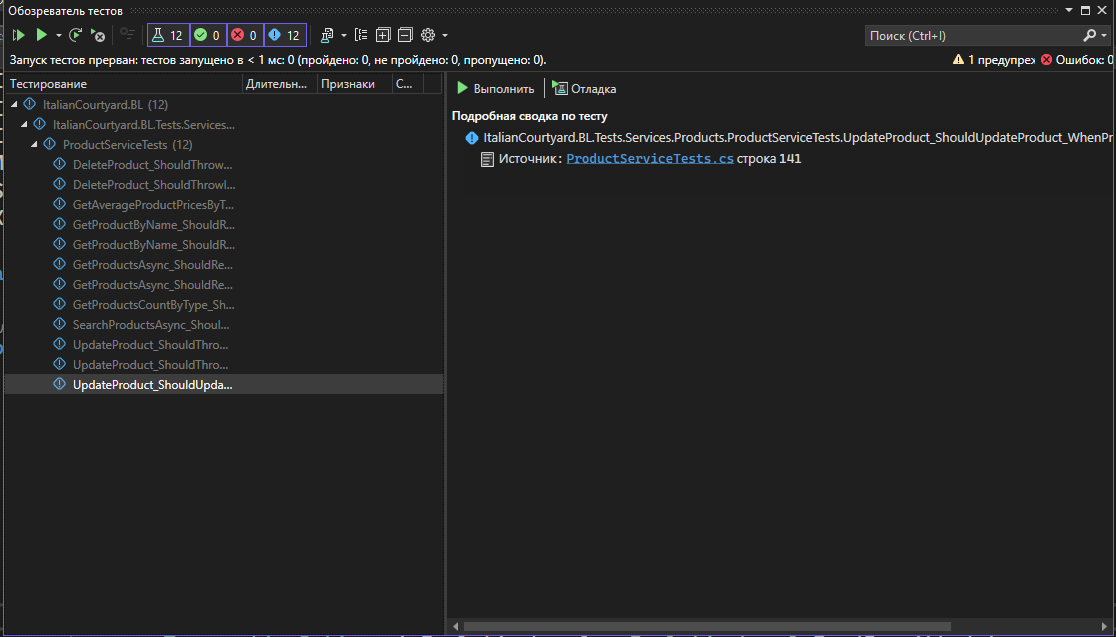


Рисунок 22 – Обозреватель тестов Xunit

В процессе тестирования были протестированы следующие категории методов:

Методы управления объявлениями:

* Добавление, редактирование и удаление продуктов.
* Поиск продуктов по тексту и категории.
* Получение информации о конкретном продукте.
* Получение списка продуктов и статистики.

Пример тестирования метода поиска продукта по имени через Xunit.

При вводе несуществующего имени продукта сервер возвращает пустой продукт (Рисунок 23)

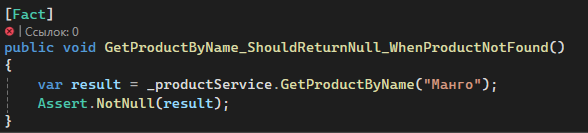


Рисунок 23 – Запрос на поиск несуществующего продукта

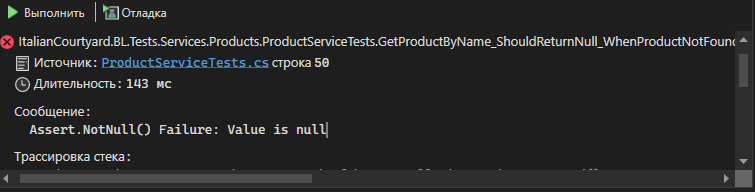


Рисунок 24 – Тест не прошел

При вводе существующего продукта сервер должен вернуть существующий продукт. (Рисунок 25)

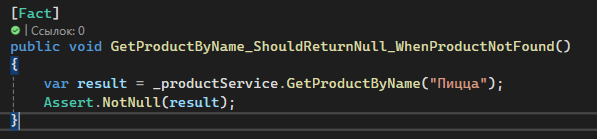


Рисунок 25 – Запрос на поиск существующего продукта

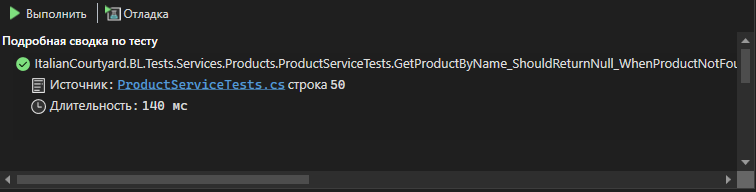


Рисунок 26 – Тест прошел

Тестирование с использованием Xunit позволило выявить и устранить различные проблемы, такие как некорректная обработка запросов к базе данных, ошибки валидации данных и непредвиденное поведение. В результате тестирования подтверждена корректная работа всех методов, что обеспечивает высокий уровень функциональности и надежности приложения

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 20.

Таблица 20 – 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 |

Продолжение таблицы 20

|  |  |
| --- | --- |
| Виды работ | Трудоёмкость, дн. |
| 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 21. За основу заработной платы студента, разрабатывающего программный продукт, взята величина его заработной платы на должности «Слесарь МСР» в компании «ООО ВиртуумЛаб».

Таблица 21 – Расчет основной заработной платы

|  |  |  |
| --- | --- | --- |
| Персонал | Количество  (чел.) | ЗП за время разработки  (руб.) |
| Руководитель  диплома | 1 | 3 200 |
| Студент | 1 | 23 800 |
| Всего | 2 | 27 000 |

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 = 3 200 руб.

Величина оплаты труда студента на своей рабочей должности составит 27 000 руб. В расчетах исходим из того, что заработная плата начисляется за 31 день. Посуточная заработная плата студента за один месяц работы в мае составит 27 000 / 31 = 871 руб.

Время работы студента – 128 часов за 16 дней (с учетом 8 часов работы в день) осуществления проекта. Тогда зарплата за 16 дней работы составит 871 \* 16 = 13 936 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 2025 г. – 7 руб. 33 коп).

Затраты на электроэнергию показаны в таблице 22.

Таблица 22 – Расчет затрат на электроэнергию

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Оборудование | Количество,  шт. | Общая  стоимость,  руб. | Общая  мощность,  КВт | Количество часов работы, р.ч. |
| Персональный компьютер | 1 | 750,59 | 0,8 | 128 |
| Лазерный принтер | 1 | 15,39 | 0,7 | 3 |
| Итого | 2 | 765,98 | 1,5 | 131 |

Амортизационные отчисления.

Расчет амортизационных отчислений на рабочий период производится по формуле 3.

Сам = (Спер\*Н a) / (365/16), где (3)

Спер – первоначальная стоимость оборудования;

Нa – годовая норма амортизации (20%);

365 – количество дней в году;

16 – количество дней на разработку проекта.

Амортизационные отчисления рассчитываются на основании таблицы 23.

Таблица 23 – Амортизационные отчисления

|  |  |  |  |
| --- | --- | --- | --- |
| Оборудование | Количество,  шт. | Первоначальная стоимость оборудования, руб. | Стоимость амортизационных отчислений, руб. |
| Ноутбук | 1 | 115 000 | 1 008,22 |
| Лазерный  принтер | 1 | 13 800 | 120,99 |
| Итого | 2 | 128 800 | 1 129,21 |

Затраты на материалы.

Расчет затрат на материалы осуществляется согласно фактическим данным в год, которые представлены в таблице 24.

Таблица 24 – Расчет затрат на материалы

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование  материала | Количество шт. | Стоимость за 1 шт., руб. | Общая стоимость, руб. |
| Бумага | 1 | 319 | 319 |
| Картридж для принтера | 1 | 3 900 | 3 900 |
| Флеш-накопитель | 1 | 332 | 332 |
| Итого | | | 4 551 |

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

Произведем расчет накладных расходов для выбранного проекта и покажем его в табл. 25.

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 (4)

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

Таблица 25 – Расчет накладных расходов

|  |  |  |  |
| --- | --- | --- | --- |
| Персонал | Количество | Заработная | Накладные |
|  | (чел) | плата, руб. | расходы, руб. |
| Руководитель диплома | 1 | 3 200 | 992 |
| Студент | 1 | 23 800 | 7 378 |
| Итого | | | 8 370 |

Расчет отчислений в социальные фонды (таблица 26):

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

% отч. – процент отчислений (В 2025 г. – 30).

Таблица 26 – Расчет отчислений в социальный фонд

|  |  |  |  |
| --- | --- | --- | --- |
| Персонал | Количество  (чел) | Заработная плата, руб. | Сумма  отчислений, руб. |
| Руководитель дплома | 1 | 3 200 | 960 |

Продолжение таблицы 26

|  |  |  |  |
| --- | --- | --- | --- |
| Студент | 1 | 23 800 | 7 140 |
| Итого | | | 8 100 |

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

С = 27 000 + 766 + 1 129 + 4 551 + 8 370 + 8 100 = 49 916 руб. (6)

Удельный вес по элементам затрат представлен в таблице 27.

Таблица 27 – Удельный вес по элементам затрат

|  |  |  |
| --- | --- | --- |
| Элементы затрат | Сумма (руб.) | Удельный вес (%) |
| Зарплата персонала | 27 000 | 54 |
| Расходы на электроэнергию | 766 | 1,5 |
| Сумма амортизационных отчислений | 1 129 | 2,3 |
| Сумма затрат на материалы | 4 551 | 9,1 |
| Накладные расходы | 8 370 | 16,8 |
| Отчисления в соц. фонды | 8 100 | 16,2 |
| Итого | 49 916 | 100 |

На рисунке 20 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.



Рисунок 20 – 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 (54%), расходов на электроэнергию (1,5%), отчислений в социальные фонды (16,2%), амортизационных отчислений (2,3%), накладных расходов (16,8%) и затрат на материалы (9,1%). Общая сумма себестоимости разработанной информационной системы составляет 49 916 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.

3 ТРЕБОВАНИЯ ПО ТЕХНИКЕ БЕЗОПАСНОСТИ ПРИ РАБОТЕ С ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКОЙ

Только длительная работа за компьютером может оказать существенное влияние на здоровье человека.

Рассмотрим основные аспекты длительной работы за компьютером:

* 1. Работающий за компьютером человек длительное время должен сохранять относительно неподвижное положение, что негативно сказывается на позвоночнике и циркуляции крови во всем организме (застой крови). При длительных нарушениях циркуляции крови нарушается питание тканей и повреждаются стенки сосудов, что в свою очередь приводит к их необратимому расширению. Такое расширение сосудов наблюдается, например, при геморрое.
  2. Чтение информации с монитора вызывает перенапряжение глаз. Возникает это главным образом потому, что во время чтения с монитора расстояние от текста до глаз постоянно остается одним и тем же, из-за этого мышцы глаз, регулирующие аккомодацию, находятся в постоянном напряжении. Со временем это может привести к нарушению аккомодативной способности глаз и, следовательно, к нарушениям зрения.

Дисплейная болезнь, характеризуется нарушением аккомодации глаз из-за длительного перенапряжения ресничного тела. Синдром сухого глаза – собирательное название заболевания, вызванного нарушением увлажнения передней поверхности глаза (роговицы) слезной жидкостью. Также длительная работа за компьютером может увеличить риск таких глазных заболеваний как миопия (близорукость), дальнозоркость, глаукома.

* 1. Длительная работа на клавиатуре приводит к перенапряжению суставов кисти и мышц предплечья, что вызывает развитие туннельного синдрома запястья.
  2. Работа за компьютером предполагает переработку большого массива информации и постоянную концентрацию внимания, поэтому при длительной работе за компьютером нередко развивается умственная усталость и нарушение внимания.
  3. Человек, работающий за компьютером, вынужден все время принимать решения, от которых зависит эффективность его работы. Поэтому, длительная работа за компьютером, часто является причиной хронического стресса.
  4. Все чаще появляются сообщения о возникновении компьютерной зависимости. Действительно, длительная работа за компьютером, работа в Интернете и компьютерные игры могут вызвать подобные психические расстройства.
  5. Работа за компьютером нередко поглощает все внимание работающего человека и потому, такие люди часто пренебрегают нормальным питанием и работают впроголодь весь день. Неправильное питание приводит к снижению интеллектуальных способностей человека.
  6. Гиподинамия, стресс, вредные привычки и неправильное питание являются основными причинами сердечно-сосудистых заболеваний и диабета. Таким образом, человек длительное время, работающий за компьютером подвергается реальному риску сердечно-сосудистых заболеваний, различных заболеваний глаз, двигательного аппарата, органов желудочно-кишечного тракта, психических расстройств.

Лечебная гимнастика.

Для развития мышц спины существует множество упражнений, в основном это наклоны в разные стороны. Очень полезно плавание и упражнения на турнике. В том случае, если вы работаете в офисе, старайтесь почаще потягивайтесь и вообще, делайте как можно больше движений, даже без надобности.

Упражнение для рук.

Чем чаще вы будете прерываться для выполнения упражнений, тем больше они принесут пользы.

* Встряхните руки;
* Сжимайте пальцы в кулаки (10 раз);
* Вращайте кулаки вокруг своей оси;
* Надавливая одной рукой на пальцы другой руки со стороны ладони, как бы выворачивая ладонь и запястье наружу.

Положения рук при работе за компьютером.

В процессе работы старайтесь удерживать туловище в вертикальном положении, опираясь на спинку стула или кресла. Ступни ног должны прочно стоять на полу или на специальной подставке, плечи следует держать в расслабленном состоянии. В оптимальной позиции руки должны быть согнуты в локте под углом 90°. Локти должны располагаться вблизи туловища.

Профилактика заболеваний органов дыхания.

* Как можно чаще делайте влажную уборку помещения и проветривайте его;
* Для увеличения влажности можно ставить открытую емкость с водой.

Профилактика нервных расстройств.

* Постарайтесь сделать так, чтобы при работе компьютер давал как можно меньше сбоев и меньше раздражал вас;
* Как можно чаще прерывайте работу с компьютером.

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.

Экран монитора должен располагаться на расстоянии вытянутой руки сидящего за столом пользователя. Отрегулируйте высоту подставки монитора таким образом, чтобы центр экрана находился чуть ниже уровня глаз. При отсутствии такой возможности желательно установить монитор на специальную полку или подставку.

При работе с бумажными документами рекомендуется установить на корпус монитора специальный держатель для бумаг.

Упражнения для глаз.

* Зажмурьте глаза на ~ 10 секунд;
* Быстро моргайте в течение ~5-10 сек;
* Сделайте комплекс упражнений:

ВЫВОДЫ И ЗАКЛЮЧЕНИЕ

В рамках данного проекта было разработано desktop-приложение для сотрудников пиццерии, позволяющее управлять меню, отслеживать и завершать заказы клиентов, а также анализировать статистику по заказам, продуктам и покупателям.

Целью проекта являлась разработка удобного и функционального приложения для автоматизации работы пиццерии. В результате было создано работоспособное решение, упрощающее управление заказами и контроль продаж.

Для достижения поставленной цели были выполнены ключевые задачи:

* Спроектирована база данных
* Разработана база данных
* Разработана информационная система
* Ограничен доступ к системе неавторизованному пользователю

Приложение обеспечивает удобство взаимодействия сотрудников с системой и повышает эффективность работы заведения.

Были учтены требования по безопасности и надежности приложения, а также проведен расчет себестоимости разработки. Приложение может быть внедрено в работу пиццерии для оптимизации процессов и улучшения качества обслуживания клиентов.

СПИСОК ЛИТЕРАТУРЫ

1. Федорова Г.Н. Разработка модулей программного обеспечения для компьютерных систем: учебник для студ. учреждений сред. проф. образования / Г.Н. Федорова – 3-е изд., испр. – М.: Издательский центр «Академия», 2019. – 384 с.
2. Кудрина, Е.В. Основы алгоритмизации и программирования на языке С#: учеб. пособие для СПО / Е.В. Кудрина, М.В. Огнева. – М.: Издательство Юрайт, 2019.
3. Н.А. Виноградова. 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», 2017.
4. Беляков, Г. И. Охрана труда и техника безопасности: учебник для среднего профессионального образования / Г. И. Беляков. — 3-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2019. — 404 с.
5. Экономика отрасли информационных систем : учебное пособие для среднего профессионального образования / А. Л. Рыжко, Н. А. Рыжко, Н. М. Лобанова, Е. О. Кучинская. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 176 с.

Электронные ресурсы:

1. Национальный открытый университет [Электронный ресурс] Режим доступа: свободный <https://www.intuit.ru/> (Дата обращения 12.05.2022)
2. Metanit – язык программирования C# и платформа .NET. [Электронный ресурс] Режим доступа: свободный <https://metanit.com/sharp/> (Дата обращения 20.05.2022)

ПРИЛОЖЕНИЕ А – Код программы

Весь проект содержит много строчек кода, поэтому прикрепить весь код не представляется возможным. Для этого приложение загружено в публичный репозиторий системы GitHub.

Ссылка на GitHub: Пришлю ссылку

Обработка данных:

* Сохранение продукта

public void SaveProduct(Product product)

{

using (var connection = new SqlConnection(DataBaseHelper.ConnectionString))

{

connection.Open();

var command = new SqlCommand(

@"INSERT INTO products

(product\_name, type, cost, description, wheight, ccal, photo)

VALUES

(@productName, @type, @cost, @description, @wheight, @ccal, @photo)",

connection);

command.Parameters.AddWithValue("@productName", product.ProductName);

command.Parameters.AddWithValue("@type", product.Type);

command.Parameters.AddWithValue("@cost", product.Cost);

command.Parameters.AddWithValue("@wheight", product.Weight);

command.Parameters.AddWithValue("@ccal", product.Ccal);

command.Parameters.AddWithValue("@description",

string.IsNullOrEmpty(product.Description) ? DBNull.Value : (object)product.Description);

command.Parameters.AddWithValue("@photo",

product.Photo == null || product.Photo.Length == 0 ? DBNull.Value : (object)product.Photo);

command.ExecuteNonQuery();

}

} }

* Поиск продукта

public async Task<List<Product>> SearchProductsAsync(string searchText)

{

var products = new List<Product>();

using (var connection = new SqlConnection(DataBaseHelper.ConnectionString))

{

await connection.OpenAsync();

var command = new SqlCommand(

"SELECT \* FROM products WHERE product\_name LIKE @search OR type LIKE @search",

connection);

command.Parameters.AddWithValue("@search", $"%{searchText}%");

using (var reader = await command.ExecuteReaderAsync())

{

while (await reader.ReadAsync())

{

products.Add(new Product

{

Id = reader.GetInt32(0),

ProductName = reader.GetString(1),

Type = reader.GetString(2),

Cost = reader.GetDecimal(3),

Description = reader.IsDBNull(4) ? null : reader.GetString(4),

Weight = reader.GetDecimal(5),

Ccal = reader.GetDecimal(6),

Photo = reader.IsDBNull(7) ? null : (byte[])reader[7]

});

}

}

}

return products;

}

* Завершение заказа

public void FinishOrder(int id)

{

using var connection = new SqlConnection(DataBaseHelper.ConnectionString);

connection.Open();

using var command = new SqlCommand(

@"UPDATE orders

SET order\_status = 'Завершен',

actual\_delivery\_time = GETDATE()

WHERE id = @orderId",

connection);

command.Parameters.AddWithValue("@orderId", id);

command.ExecuteNonQuery();

}

* Обновление почты и номера телефона пользователя

public void UpdateEmailAndPhone(int employeeId, string email, string phone)

{

using (var connection = new SqlConnection(DataBaseHelper.ConnectionString))

{

connection.Open();

using (var command = new SqlCommand(

"UPDATE employee SET email = @email, phone = @phone WHERE id = @id",

connection))

{

command.Parameters.AddWithValue("@id", employeeId);

command.Parameters.AddWithValue("@email", email);

command.Parameters.AddWithValue("@phone", phone);

int rowsAffected = command.ExecuteNonQuery();

if (rowsAffected == 0)

{

throw new Exception($"Сотрудник с ID {employeeId} не найден");

}

}

}

}

ПРИЛОЖЕНИЕ Б – Флэш-накопитель с программой