

РЕФЕРАТ

Данная работа содержит 58 страниц. В работе использованы 16 иллюстраций, 33 таблицы, 30 источников литературы.

Перечень ключевых слов: заказ, главная, завтраки, обеды, ужины, корзина, профиль, авторизация.

Объектом разработки является программно-информационная система приёма и обработки заказов на предприятии быстрого питания.

Целью работы является проектирование и разработка программно-информационная система приёма и обработки заказов на предприятии быстрого питания.

Основными задачами при разработке данной программно-информационной системы являются:

1. Разработать модель данных для хранения информации о товарах, продуктах, сотрудниках, покупателях, продажах, заказах и их состоянии.
2. Разработать алгоритм для осуществления продажи.
3. Разработать алгоритм для осуществления продажи.
4. Разработать алгоритм для управления каталогом товаров, склада, учетными данными сотрудников.
5. Разработать программно-информационную систему приёма и обработки заказов на предприятии быстрого питания.

Во введении обоснована актуальность темы, сформулирована цель работы, поставлены задачи, а также приведено краткое содержание каждого из разделов.

В техническом задании отражено основание и назначение разработки, указаны требования к системе и оформлению.

В техническом проекте приводится описание общих сведений, проект данных, проектирование архитектуры и пользовательского интерфейса.

В заключении расписаны главные результаты работы, полученные в ходе разработки.

ABSTRACT

This work contains 58 pages. The work uses 16 illustrations, 33 tables, 30 sources of literature.

List of keywords: main, basket, orders, profile, exit, accounting, quantity of goods.

The object of development is a software and information system for receiving and processing orders at a fast food enterprise.

The purpose of the work is to design and develop a software and information system for receiving and processing orders at a fast food enterprise.

The main tasks in developing this software and information system are:

1. Develop a data model for storing information about goods, products, employees, Users, sales, orders and their status.
2. Develop an algorithm for making a sale.
3. Develop an algorithm for managing a product catalog, warehouse, employee credentials.
4. Develop a software and information system for receiving and processing orders at a fast food enterprise.

The main tools for designing the architecture and interface of the software and information system are Visual Studio.

The introduction substantiates the relevance of the topic, formulates the purpose of the work, sets the tasks, and provides a brief summary of each section.

The technical task reflects the basis and purpose of the development, specifies the requirements for the system and design.

The technical project provides a description of general information, data design, architecture and user interface design.

The conclusion describes the main results of the work obtained during the development.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	7
1 Анализ предметной области	9
1.1 Назначение предприятий быстрого питания	9
1.2 Особенности системы управления заказами на предприятиях быстрого питания	13
1.3 Использование средств РНР для создания программно-информационной системы приёма и обработки заказов на предприятии быстрого питания	19
2 Техническое задание	23
2.1 Основание для разработки	23
2.2 Назначение разработки	23
2.3 Требования к программной системе	26
2.3.1 Требования к данным программно-информационной системы	26
2.3.2 Функциональные требования к программной системе	30
2.3.2.1 Вариант использования «Авторизация»	32
2.3.2.2 Вариант использования «Регистрация»	33
2.3.2.3 Вариант использования «Добавить товар»	33
2.3.2.4 Вариант использования «Добавить еду»	33
2.3.2.5 Вариант использования «Добавить заказ»	34
2.3.2.6 Вариант использования «Очистить корзину»	34
2.3.2.7 Вариант использования «Выйти из аккаунта»	34
2.3.2.8 Вариант использования «Оплатить»	35
2.3.2.9 Вариант использования «Добавление учетной записи пользователя»	35
2.3.2.10 Вариант использования «Редактирование учетной записи покупателя»	35
2.3.2.11 Вариант использования «Удалить учетную запись профиля»	36
2.3.2.12 Вариант использования «Закрыть сайт»	36
2.3.3 Требования к пользовательскому интерфейсу программной си-	

стемы	36
2.3.4 Нефункциональные требования к программной системе	37
2.4 Требования к оформлению документации	41
3 Технический проект	42
3.1 Общие сведения о программно-информационной системе	42
3.2 Проект данных программно-информационной системы	42
3.2.1 Структура базы данных	42
3.3 Проектирование архитектуры программно-информационной системы	45
3.3.1 Компоненты программно-информационной системы	45
3.3.2 Архитектура программно-информационной системы	48
3.4 Проектирование пользовательского интерфейса программно-информационной системы	49
ЗАКЛЮЧЕНИЕ	58
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	58

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

Меню – Список товаров.

Товар – Единица товара(пицца) для продажи.

Склад – Список продуктов на складе.

Корзина – Формирующийся продавцом список товаров.

База данных – Программа для сбора и структурирования и хранения информации.

Авторизация – операция, определяющая права доступа пользователя к программе.

Учетная запись – информация данных о сотруднике, необходимая для работы в программе.

Microsoft SQL Server – система управления реляционными базами данных.

ВВЕДЕНИЕ

Актуальность разработки программно-информационной системы (ПИС) для приёма и обработки заказов в предприятии быстрого питания заключается в необходимости повышения эффективности работы бизнеса и улучшения качества обслуживания клиентов. В условиях современного рынка, где конкуренция в сфере общественного питания крайне высокая, важно предложить клиентам максимально быстрый и удобный сервис. Внедрение ПИС позволяет значительно ускорить процесс приёма заказов, что особенно важно для предприятий быстрого питания, где скорость обслуживания является ключевым фактором удовлетворенности клиентов.

Кроме того, использование автоматизированных систем помогает снизить количество ошибок, связанных с ручным вводом информации. Это особенно актуально в условиях большой загруженности и многозадачности сотрудников, когда вероятность ошибок увеличивается. ПИС позволяет автоматически фиксировать все данные, исключая недоразумения и ускоряя обработку заказов. Это, в свою очередь, снижает нагрузку на персонал и позволяет сосредоточиться на других аспектах работы, таких как обслуживание клиентов и контроль качества.

Интеграция программно-информационной системы с другими системами предприятия также является важным аспектом её разработки. Она может объединять данные о заказах с финансовыми и учётными системами, а также с системой управления запасами. Это позволяет автоматизировать не только процесс приёма и обработки заказов, но и все сопутствующие операции, такие как учёт продуктов, расчёт прибыли, анализ продаж и прогнозирование потребностей в товарных остатках.

Дополнительным преимуществом является сбор и анализ данных. С помощью ПИС можно легко отслеживать информацию о заказах, времени обслуживания, предпочтениях клиентов и других важных показателях. Это даёт возможность владельцам бизнеса принимать обоснованные решения на основе фактических данных, что способствует улучшению качества обслужи-

вания и повышению прибыли. Программно-информационная система также может помочь в создании лояльности среди клиентов, например, через реализацию программ лояльности и индивидуальных предложений.

Таким образом, разработка ПИС для приёма и обработки заказов в предприятии быстрого питания представляет собой не только необходимость для эффективного функционирования предприятия, но и важный инструмент для повышения его конкурентоспособности.

Целью данной работы является проектирование и разработка архитектуры программно-информационной системы приёма и обработки заказов на предприятии быстрого питания.

Основными задачами при разработке данной программно-информационной системы являются:

1. Разработать модель данных для хранения информации о товарах, продуктах, сотрудниках, покупателях, продажах, заказах и их состоянии.
2. Разработать алгоритм для осуществления продажи.
3. Разработать алгоритм для управления каталогом товаров, склада, учетными данными сотрудников.
4. Разработать программно-информационную систему приёма и обработки заказов на предприятии быстрого питания.

Основными средствами проектирования архитектуры и интерфейса программно-информационной системы являются Visual Studio.

Во введении обоснована актуальность темы, сформулирована цель работы, поставлены задачи, а также приведено краткое содержание каждого из разделов.

В техническом задании отражено основание и назначение разработки, указаны требования к системе и оформлению.

В техническом проекте приводится описание общих сведений, проект данных, проектирование архитектуры и пользовательского интерфейса.

В заключении расписаны главные результаты работы, полученные в ходе разработки.

1 Анализ предметной области

1.1 Назначение предприятий быстрого питания

Индустрия быстрого питания зародилась в начале 20-го века, когда социально-экономические изменения, такие как урбанизация и индустриализация, создали потребность в удобстве, доступности и быстром увеличении производства. В первые десятилетия 20 века предприятия общественного питания существовали преимущественно в ресторанах и барах, но для удовлетворения растущих потребностей населения, особенно в городах, в 1990 году возникла необходимость разработки новой системы, которая могла бы обеспечивать питанием не только быстро, но и быстро. также по разумной цене [3, с.65].

Первая волна ресторанов быстрого питания появилась в США, где бурный рост автомобильной промышленности и изменение образа жизни потребовали нового подхода к обслуживанию клиентов. Одним из первых таких зданий стал ресторан White Castle, открывшийся в 1921 году в Канзас-Сити. Это место стало известно своим устоявшимся меню с небольшими закусками, которые быстро готовятся и по фиксированным ценам. В отличие от ресторанов того времени, которые готовили еду на заказ и отнимали много времени, White Castle упростил процесс, создав систему меню, основанную на потоке.

В 1940-х годах в США возникла новая волна ресторанов быстрого питания, которая позже распространилась по всему миру. Рестораны McDonald's, основанные братьями Ричардом и Морисом Макдональдами в 1940 году в Сан-Бернардино, штат Калифорния, сыграли ключевую роль в развитии сети быстрого питания [9, с.102].

Их инновационная система, включающая стандартизированные процессы и оптимизированные методы обработки пищевых продуктов, сделала обслуживание намного быстрее и эффективнее. Это стало основой гибкой системы, которая позже была адаптирована и внедрена в ресторанах по всему миру.

В 1955 году Рэй Крок, предприниматель, открывший франшизу «Макдональдс», начал активно развивать сеть, превратив ее в глобальную компанию. После успеха McDonald's в США бизнес быстрого питания начал расширяться и в других странах.

Ресторанная сфера продолжала расти, и новые бренды, такие как Burger King, KFC, Wendy's и Taco Bell, появились во многих странах, став популярными не только в США, но и на международной арене. Эти компании также активно внедрили процесс франчайзинга, что значительно ускорило их расширение и привело к стабильному росту.

С 1970-х годов точки быстрого питания стали неотъемлемой частью городского ландшафта. За это время стали появляться новые варианты, в том числе кафе и рестораны, ориентированные на более молодую аудиторию, а также сети, специализирующиеся на определенных видах еды, например, пицце или китайской кухне [11, с.87].

Основное внимание уделяется расширенному меню и дополнительным позициям, таким как салаты, десерты, а также вариантам для людей с особыми потребностями, например, не введенным веганским и безглютеновым продуктам. В последние десятилетия индустрия быстрого питания продолжала расти, уделяя особое внимание таким инновациям, как здоровое питание, устойчивое развитие и улучшенное питание.

Достижения в области технологий также привели к появлению новых возможностей, таких как службы доставки еды, мобильные приложения для заказа и даже рестораны без персонала, полностью изменив традиционное восприятие фаст-фуда. Таким образом, индустрия быстрого питания – это история адаптации и приспособления в ответ на требования времени. От первых обычных ресторанов до глобальных сетей — эти предприятия продолжают расти, обеспечивая удобство и доступ к еде миллионам людей по всему миру.

Кроме того, компании быстрого питания играют важную роль в удовлетворении потребности в стандартизированном и качественном продукте, приготовленном быстро и по определенным стандартам. Благодаря строгим

правилам и внутренним правилам таких компаний вкус, внешний вид и качество продуктов питания всегда одинаковы, что дает клиентам уверенность и удовлетворение [5, с.90].

Это также снижает риск ошибок и разочарований, связанных с вариациями, как это может случиться с домашней кухней или неформальным рестораном. Центральное место занимает способность таких фирм быстро реагировать на изменения потребительских предпочтений и рыночных тенденций. В последние годы компании быстрого питания адаптировались к требованиям здорового питания, предлагая в своих меню низкокалорийные, безглютеновые или веганские блюда.

Это помогает расширить потребительскую базу и привлечь новые сегменты рынка, ориентированные на здоровый образ жизни. Индустрия быстрого питания также создает рабочие места и стимулирует экономику. Сети ресторанов быстрого питания могут обслуживать несколько мест, что делает их важной частью городской инфраструктуры.

В этих проектах часто нанимаются молодые люди, студенты и граждане, ищущие временную или гибкую работу, помогая решать местные проблемы занятости. Таким образом, индустрия быстрого питания является не только прибыльным бизнесом, удовлетворяющим потребности в продуктах питания, но и способствующим экономическому и социальному развитию общества.

В них люди находят решения своих проблем с питанием, а компании получают возможность удовлетворить более широкий спектр потребностей потребителей, что, в свою очередь, позволяет таким компаниям расширять и развивать свои сети.

Бизнес быстрого питания включает в себя различные области дизайна услуг, концепций и предложения продуктов, но все они преследуют одну и ту же цель — предоставить клиентам быструю и доступную еду. Индустрию быстрого питания можно разделить на различные характеристики, каждая из которых ориентирована на определенную аудиторию и потребности.

Одним из наиболее распространенных вариантов быстрого питания является традиционный фастфуд, предлагающий ограниченное, но популярное меню, состоящее в основном из гамбургеров, картофеля фри, напитков и десертов. Как правило, эти заведения работают по принципу самообслуживания: клиент делает заказ на кассе, а затем получает приготовленную еду в течение короткого периода времени [2, с.102].

Основное внимание уделяется скорости процесса и стандартам продукции. Примером таких компаний является всемирно известная сеть McDonald's. В последние десятилетия популярность сайтов, специализирующихся на здоровом питании, возросла.

В отличие от традиционных ресторанов быстрого питания, они предлагают более разнообразное меню, ориентированное на людей, заботящихся о своем здоровье или соблюдающих диету. Здесь можно найти салаты, овощные блюда, низкокалорийную еду и напитки. Эти предприятия сосредоточены на инновационных и высококачественных продуктах, отвечающих современным инициативам здорового образа жизни.

Специализированные точки быстрого питания — это кафе и рестораны, которые обычно специализируются на определенной кухне, например, на продукте, китайской еде, мексиканской еде или блюдах. В этих заведениях может быть собственный бизнес или служба официантов.

Хотя такие кафе и рестораны также делают упор на быстрое обслуживание, их меню разнообразны и неповторимы. Они, как правило, предлагают блюда, на приготовление которых уходит немного меньше времени, но при этом они готовятся быстро и удобно для потребителя. Еще одна интересная тенденция — появление так называемых «ресторанов без персонала» или автоматизированного фастфуда.

В этих заведениях используются высокотехнологичные технологии для обеспечения быстрого обслуживания: заказы размещаются через мобильное приложение или киоск самообслуживания, а еда готовится и доставляется клиенту с помощью автоматизированных систем. Это значительно снижает трудозатраты и увеличивает скорость [5, с.77].

Такие рестораны также могут использовать роботизированную систему доставки или дрон. Компании, которые специализируются на доставке еды на дом или в офис, также становятся все более популярными. В этих ресторанах, как правило, меньше шагов, а их бизнес-модель ориентирована на быструю доставку заказов. Примеры таких услуг можно найти в пиццериях или ресторанах, предлагающих разнообразные блюда, доставляемые в удобном формате сообщений.

Ключевой особенностью всех видов предприятий быстрого питания является их способность адаптироваться к современным технологиям и потребностям потребителей. В последние годы предприятия быстрого питания начали уделять больше внимания цифровым решениям, таким как мобильные приложения для заказа еды, автоматические кассы и улучшенные системы доставки. Это не только увеличивает скорость обработки, но и улучшает обслуживание клиентов, делая заказ более простым и персонализированным.

Таким образом, предприятия быстрого питания могут различаться по стилю, специализации и концепции, но их объединяет один ключевой фактор – они подают еду быстро, эффективно и удобно, думая о современных нуждах клиентов.

1.2 Особенности системы управления заказами на предприятиях быстрого питания

Первые системы управления заказами для заведений быстрого питания начали развиваться в середине 20 века, когда начал расти спрос на быстрое и индивидуальное обслуживание при поставках продуктов питания. Одним из первых примеров использования такой системы стала сеть McDonald's, которая в 1940-х годах начала искать пути улучшения процессов обслуживания клиентов. В этот период важнейшей задачей было сокращение простоев, требующих технических решений по автоматизации работы ресторана. Изначально системы обработки заказов в сетях быстрого питания не были такими сложными, как их современные аналоги.

Основной целью было упростить процесс приема заказов, доставки их на кухню и доставки клиенту. В ресторанах McDonald's братья Ричард и Морис Макдональд разработали так называемую «систему братства», которая обеспечивала систематические способы приготовления еды, ускоряла обслуживание и уменьшала количество ошибок. Важным аспектом этого процесса является принцип «жидкости», при котором все сотрудники выполняют строго определенные задачи, что обеспечивает эффективность и скорость [6, с.90].

Однако сама система проверки заказов еще не была полностью работоспособна — кассиры и другой персонал по-прежнему активно работали, вручную принимая заказы на кухню и проверяя их выполнение. По мере роста отрасли, особенно после того, как Рэй Крок превратил McDonald's в международную сеть, рестораны начали использовать более сложные системы заказов. К этому времени сети ресторанов начали внедрять стратегии по ускорению обслуживания, включая специальные ингредиенты и дисплеи для кухонного персонала, а также быстрый ввод данных банками питания.

Эти ранние системы позволяли значительно ускорить обмен информацией между банком и кухней, что, в свою очередь, сокращало время ожидания заказов и повышало точность заказов. Автоматизация процессов началась в 1960-х годах.

К началу 1970-х годов, по мере совершенствования компьютерных технологий, начали появляться первые компьютеризированные системы управления закупками. Значительно улучшена скорость и точность обработки данных.

Такая система начала интегрировать банковские, кухонные и складские операции, позволяя не только получать и контролировать заказы, но и эффективно управлять запасами, в результате чего излишки или дефицит уменьшались.

Эти системы стали основой для дальнейшей автоматизации ресторанного бизнеса и внедрения более надежных решений в сфере услуг. Управление закупками в индустрии быстрого питания приняло множество основных

форм: от классических методов организации бизнеса середины 20 века до современных высокотехнологичных решений [10, с.201].

Это явление было тесно связано с развитием информационных технологий, увеличением размера бизнеса и изменением потребностей потребителей. Вначале, в 1940-х годах, рестораны быстрого питания, такие как McDonald's, имели очень простую систему управления заказами.

В 1960-х и 1970-х годах, с ростом сетей быстрого питания и развитием компьютерных технологий, начали появляться первые устройства для отслеживания заказов. Стали популярными решения, основанные на значительных технологических достижениях, такие как кассовые аппараты, способные быстро вводить данные. Эти устройства позволили системе быстрее рассчитывать и обрабатывать заказы, уменьшая количество ошибок, связанных с ручным вводом.

Такие системы позволили улучшить координацию между банком и кухней, ускорив перемещение заказов и повысив точность производства. Они позволили индустрии быстрого питания увеличить количество обслуживаемых клиентов, уменьшить количество человеческих ошибок и повысить общую операционную эффективность.

В 1980-х годах, с развитием персональных компьютеров и программного обеспечения, рестораны быстрого питания начали использовать сложные автоматизированные системы. Эти системы интегрировали различные аспекты операций: от получения заказов и их соблюдения до учета запасов и финансирования.

Внедрение интеллектуальных систем прогнозирования, которые анализируют поведение, предпочтения и тенденции клиентов, позволило ресторанам лучше адаптировать свои предложения и услуги к меняющимся обстоятельствам.

Это было особенно важно в условиях постоянного изменения спроса при карантинных ограничениях и изменении структуры закупок. Кроме того, эпидемия ускорила внедрение автоматизации кухни и логистики.

Роботы и автоматизация обработки пищевых продуктов, заказа и упа-

ковки стали важными элементами в снижении зависимости от человеческого труда, повышении эффективности и сокращении взаимодействия с клиентами и общении. Автоматизация позволила быстрее выполнять заказы и обеспечить безопасность сотрудников, поскольку сократила их физическое присутствие в определенных процессах [21, с.108].

Рост использования технологий, таких как искусственный интеллект, стал ещё одним важным шагом в развитии систем управления заказами в условиях пандемии. Адаптивные алгоритмы позволяли не только улучшить качество обслуживания, но и оптимизировать внутренние процессы, например, управление запасами продуктов, сокращение времени ожидания заказов и предотвращение ошибок.

Таким образом, период ковида стал катализатором роста и развития систем управления заказами на предприятиях быстрого питания. Потребность в инновационных решениях, обеспечивающих безопасность, ускорение обслуживания и адаптацию к новым условиям, вынудила рестораны быстрого питания активно внедрять новые технологии. Системы управления заказами стали ключевыми для сохранения бизнес-операций, улучшения клиентского опыта и успешной работы в условиях кризиса.

Зарубежный опыт управления заказами на предприятиях быстрого питания в период пандемии COVID-19 демонстрирует целый ряд инновационных решений, направленных на адаптацию к изменившимся условиям работы и на обеспечение безопасности как для клиентов, так и для персонала. Пандемия стала настоящим вызовом для ресторанного бизнеса, однако она также предоставила возможности для внедрения новых технологий и оптимизации бизнес-процессов, что, в свою очередь, сыграло важную роль в сохранении конкурентоспособности предприятий.

С началом пандемии в странах по всему миру предприятия быстрого питания, в первую очередь, начали активно переходить на модели, ориентированные на доставку и заказы на вынос. Одним из наиболее заметных трендов стало массовое внедрение мобильных приложений и онлайн-платформ для заказов.

Сеть McDonald's, например, усилила своё присутствие в цифровом пространстве, запустив улучшенные версии мобильного приложения в различных странах. Это приложение позволяло не только заказывать еду с доставкой, но и выбрать блюда, оплатить заказ и отслеживать его выполнение, минимизируя физический контакт с персоналом [17, с.109].

В странах, таких как США и Великобритания, такие изменения позволили сети значительно повысить свою долю на рынке доставки, что стало возможным благодаря разработке удобных и интуитивно понятных цифровых интерфейсов.

Еще одним важным событием стало внедрение технологии бесконтактных платежей и самообслуживания. Во Франции и Германии, например, компании быстрого питания начали создавать киоски самообслуживания, позволяющие клиентам размещать заказы без участия кассиров и обеспечивая доступность сотрудников, сокращая общение и ускоряя обработку заказов.

Такие системы также улучшили качество обслуживания клиентов, обеспечив большую гибкость процесса заказа. В то же время в ресторанах стали набирать популярность бесконтактные способы оплаты, такие как мобильные кошельки, обеспечивающие уровень безопасности как для клиентов, так и для сотрудников.

В Великобритании и Австралии наблюдается всплеск использования служб доставки еды, таких как Uber Eats, Deliveroo и DoorDash, что повысило их популярность на фоне эпидемии. Рестораны быстрого питания интегрировали свои заказы в эти платформы, что позволило им расширить клиентскую базу и обеспечить непрерывность заказов.

В некоторых странах сети быстрого питания начали разрабатывать собственные системы доставки, что позволяет им контролировать структуру доставки и повышать скорость.

При этом клиенты могут не только размещать заказы через официальные сайты или мобильные приложения, но и иметь постоянную информацию о статусе своих заказов в режиме реального времени [22, с.209].

Технологии для улучшения кухни и логистики также стали востребо-

ванными во время пандемии. В некоторых ресторанах были внедрены автоматизированные системы обработки пищевых продуктов, в том числе роботизированные системы приготовления пищи и линии упаковки, чтобы немного сократить физический контакт с сотрудниками.

В Японии, например, компания Fried Chicken использовала роботов для сборки заказов, что ускоряло процесс и снижало риски для сотрудников. Подобные решения активно тестируются в других странах, таких как Южная Корея и Сингапур, где рестораны начали использовать роботизированные устройства для доставки еды.

В то же время, чтобы более эффективно управлять заказами и поддерживать более высокий уровень обслуживания, поскольку количество заказов резко увеличивается, многие компании внедрили системы по требованию.

В США и Канаде рестораны использовали инструменты поиска для отслеживания спроса в определенные часы и дни недели. Эти системы помогали планировать работу рабочих и своевременно пополнять запасы сырья. Такие методы не только повысили операционную эффективность, но и снизили затраты за счет сокращения потерь и улучшения транспортировки.

Не менее важной была реализация планов реагирования, которая стала еще более важной во время пандемии. В различных странах рестораны начали активно внедрять инструменты для сбора обратной связи от клиентов после оформления заказа через мобильные приложения. Это позволило нам оперативно реагировать на изменение предпочтений клиентов и улучшить предоставление услуг в условиях жесткой конкуренции.

Таким образом, зарубежный опыт управления системами в индустрии быстрого питания во время Covid разнообразен и инновационен. Ключевые стратегии включают технологии бесконтактного заказа, повышение эффективности процессов за счет автоматизации и прогнозирования, расширение охвата платформ онлайн-доставки и улучшение качества обслуживания клиентов с помощью мобильных приложений.

Это помогло бизнесу сохранить свою деятельность, адаптироваться к новым обстоятельствам и продолжить обслуживание клиентов во время пан-

демии, что, в свою очередь, спасло их от спада и укрепило позиции на рынке.

Система должна быть гибкой и легко масштабируемой, чтобы адаптироваться к меняющимся обстоятельствам и требованиям бизнеса. Таким образом, особенностями системы управления заказами в индустрии быстрого питания являются скорость обработки информации, точность заказов, интеграция с другими системами и возможности работы со службами онлайн-заказа и доставки. Это делает систему неотъемлемой частью бизнес-процессов завода, повышая эффективность и качество обслуживания клиентов.

1.3 Использование средств PHP для создания программно-информационной системы приёма и обработки заказов на предприятии быстрого питания

PHP - популярный язык программирования для создания веб-приложений и динамических веб-страниц. История его возникновения и развития тесно связана с необходимостью упростить разработку веб-сайтов и создать гибкие интерактивные решения для Интернета. PHP изначально разрабатывался как небольшая утилита для одних из личных нужд пользователей, но со временем он стал одним из ведущих языков программирования в веб-разработке.

История PHP начинается в 1994 году, когда датский программист Расмус Лерддорф разработал простую программу для отслеживания посещений своей веб-страницы. Эта утилита, которую он назвал «Инструменты личной домашней страницы», была написана на языке C и предназначена для выполнения основных задач, таких как обработка форм и взаимодействие с базами данных.

Хотя изначально возможности этой системы были ограничены, она оказалась полезной и распространилась среди других веб-разработчиков, которые начали использовать ее для создания динамических веб-страниц.

В 1995 году Lerdorff начал активно развивать свою утилиту, добавляя новые функции, такие как поддержка баз данных, которые позволяли исполь-

зовать PHP для создания более сложных веб-приложений.

Тем временем была выпущена первая версия, которая получила название PHP/FI (Personal Home Page/Forms Interpreter). Этот выпуск не представлял собой полноценный язык программирования, а скорее набор скриптов, предназначенных для простого взаимодействия с веб-страницами и их документами.

В 1997 году Лердорф решил перепроектировать PHP, чтобы сделать его более универсальным и расширяемым. Они начали работать с другими разработчиками, в том числе с Андием Фредериком и Зеевой Штрайцер, над разработкой новой версии PHP под названием PHP 3. Эта версия работала лучше, чем предыдущая, и имела полную поддержку работы в базе данных и динамической сети, включая -page.

PHP 3 стал первым шагом на пути к созданию мощного современного инструмента веб-разработки и завоевал все большую популярность среди разработчиков. Благодаря своей простоте, гибкости и возможности интеграции с несколькими базами данных PHP быстро завоевал рынок и стал основным инструментом для создания динамических веб-сайтов.

В 2000 году был выпущен PHP 4, внесший значительные улучшения в производительность и расширивший возможности языка. Одним из главных нововведений стал новый механизм обработки запросов и расширенная поддержка сеансов, что позволило создавать надежные и масштабируемые веб-приложения [7, с.101].

PHP 4 также стал первой версией, официально поддерживающей объектно-ориентированное программирование, что сделало язык еще более мощным и упростило разработку крупномасштабных проектов.

В 2004 году был выпущен PHP 5, ставший еще более важным шагом в развитии языка. Одной из наиболее важных новых функций этой версии является полная поддержка объектно-ориентированного программирования, включая наследование, абстракцию и методы интерфейса.

В PHP 5 также были улучшены базы данных, расширена поддержка XML и улучшена производительность, что сделало его еще более мощным

инструментом для веб-разработки. В этот же период PHP стал активно использоваться для разработки масштабных веб-приложений, таких как системы управления контентом (CMS), интернет-магазины и форумы. С каждым новым выпуском PHP становился все лучше и функциональнее.

В 2015 году был выпущен PHP 7, который значительно улучшил производительность по сравнению с предыдущими версиями, позволил ускорить обработку запросов и снизить нагрузку на серверы. PHP 7 также обеспечивает улучшенную поддержку типов данных и новые возможности для работы с современными веб-технологиями.

PHP 7 был одним из самых успешных и популярных выпусков, поскольку он значительно повысил производительность и сделал язык более привлекательным для разработчиков. Со временем PHP продолжал развиваться: с каждым новым выпуском его функциональность улучшалась, расширялась и добавлялась поддержка современных веб-стандартов.

Программисты стали активно использовать PHP не только для создания веб-страниц, но и для разработки более сложных веб-приложений, включая социальные сети, платформы электронной коммерции и корпоративные системы. Сегодня большинство веб-сайтов используют PHP, включая популярные платформы, такие как WordPress, Facebook и Wikipedia.

Таким образом, история PHP — это история языка, который начинался как небольшой инструмент для отслеживания посещений и превратился в один из самых популярных инструментов для создания динамических и масштабируемых веб-приложений [12, с.208].

Благодаря своей гибкости, гибкости и широкой поддержке PHP продолжает оставаться одним из самых популярных языков программирования для веб-разработки, позволяя миллионам веб-сайтов по всему миру хорошо оптимизировать производительность и функциональность.

Использование инструментов PHP для разработки программного обеспечения и информационных систем приема и обработки заказов в сфере быстрого питания является эффективным решением для автоматизации и эффективного обслуживания клиентов. PHP, как популярный язык программи-

рования для веб-разработки, предлагает множество возможностей для создания гибких и масштабируемых программ, способных удовлетворить потребности ресторанной индустрии.

Разработка программного обеспечения и информационных систем на базе РНР для приема и обработки заказов начинается с проектирования прикладной системы, которая будет включать в себя несколько ключевых компонентов: интерфейс для клиентов, интерфейс для администраторов, базу данных для хранения информации о заказах и взаимодействия с внешними системами, такими как способы оплаты и реферальные системы.

Основой системы является веб-интерфейс, доступный как клиентам, так и персоналу ресторана. Для клиентов система может обеспечить онлайн-заказ через веб-страницу или мобильное приложение. На стороне РНР создаются страницы с опциями, настройками настроек и формами заказа. РНР обрабатывает входные данные, взаимодействует с базой данных для добавления информации о заказе, а также обеспечивает наличие и доставку товара. Одна из основных задач — сделать интерфейс простым и удобным в использовании, чтобы клиент мог сделать заказ быстро и без ошибок.

2 Техническое задание

2.1 Основание для разработки

Основанием для разработки программно-информационной системы является задание на выпускную квалификационную работу бакалавра «Программно-информационная система приёма и обработки заказов на предприятии быстрого питания».

2.2 Назначение разработки

Разработка программного обеспечения и информационных систем приема и обработки заказов в сфере быстрого питания призвана автоматизировать и оптимизировать все аспекты приема и обработки заказов, позволила улучшить коммуникацию с клиентами и повысить эффективность работы компании.

Такая система необходима для создания гибкой и быстро реагирующей системы, которая ускорит обслуживание клиентов, улучшит качество обслуживания и уменьшит количество ошибок, которые могут возникнуть при обработке заказов вручную.

Система в первую очередь предназначена для облегчения взаимодействия с клиентами на всех этапах: от оформления заказа до его получения и оплаты. Это включает в себя прием заказов через такие каналы, как мобильные приложения, веб-сайты, терминалы самообслуживания и кассы, а также выполнение этих заказов с минимальным вмешательством человека, что увеличивает скорость и снижает вероятность ошибки.

Кроме того, система позволяет эффективно отслеживать запасы и управлять ими, что особенно важно для предприятий быстрого питания, где скорость и точность доставки являются ключевыми факторами.

Интегрированная с базой данных, система может автоматически обновлять информацию об ингредиентах и мгновенно указывать на необходимость пополнения запасов. Это также позволяет измерять предпочтения клиентов, предоставляя информацию о скидках, рекламных акциях и нали-

чии программ лояльности, что помогает повысить уровень удовлетворенности клиентов.

Также необходима разработка программно-информационной системы для улучшения связи между объектами. Например, система может стандартизировать работу на кухне, кассиров и курьеров, обеспечивая четкое и эффективное разделение обязанностей. Это важно для предотвращения раздутия штата, ускорения обработки заказов и обеспечения более высокого уровня обслуживания, что, в свою очередь, влияет на удержание клиентов и репутацию бренда.

Дополнительно такие системы могут быть подключены к внешним платежным системам, позволяющим клиентам оплачивать заказы онлайн, а также к каналам доставки еды. Это обеспечивает дополнительное удобство для клиентов, так как они могут выбрать гораздо более удобный для них способ оплаты и доставки, а также отслеживать статус своих заказов в режиме реального времени.

Еще одним важным аспектом разработки системы является возможность формирования аналитических отчетов и статистики по заказам, продажам, предпочтениям клиентов и эффективности работы сотрудников. Это позволяет руководству компании принимать соответствующие решения по улучшению предоставления услуг, поддержанию линейки продуктов и оптимизации внутренних процессов, что в конечном итоге приводит к повышению прибыльности и операционной эффективности.

Программное обеспечение и информационные системы приема и обработки заказов в заведении быстрого питания подходят для различных участников этого процесса, так как влияют на работу ресторана и его взаимодействие с внешними и внутренними системами на многие аспекты вида.

В первую очередь такая система действительно необходима предприятию быстрого питания, поскольку она позволяет существенно улучшить внутренние процессы, повысить эффективность обслуживания клиентов и снизить затраты.

Это важный инструмент для мониторинга закупок, запасов, персонала и движения денежных средств. Для владельцев и менеджеров предприятий быстрого питания такая система является незаменимым инструментом стратегического управления бизнесом.

Это помогает оптимизировать бизнес-процессы, улучшить контроль качества обслуживания и снизить человеческий фактор. Для клиентов программное обеспечение информационных систем является важным компонентом улучшения предоставления услуг. Он обеспечивает возможность быстрого оформления заказов через различные каналы – мобильные приложения, сайты, терминалы самообслуживания, а также позволяет отслеживать статус заказа в режиме реального времени.

Это повышает удобство и удовлетворенность, поскольку клиенты могут выбирать предпочтительные способы оплаты и доставки. Благодаря таким программам клиенты могут получать персональные предложения, скидки, что делает их пребывание более приятным и побуждает вернуться в ресторан.

Кроме того, для поставщиков программное обеспечение и информационные системы являются важным инструментом эффективного взаимодействия с рестораном.

Система позволяет точно отслеживать заказы на продукты питания и ингредиенты, помогая поставщикам быстро реагировать на потребности ресторана и вовремя поставлять нужные продукты. Это помогает улучшить управление запасами и сократить сбои в цепочке поставок, что важно для бесперебойной работы компании.

Таким образом, программное обеспечение и информационные системы приема и обработки заказов в заведении быстрого питания подходят для разных типов людей: владельцев и руководителей бизнеса, операторов ресторанов, клиентов и поставщиков.

Для каждой из этих групп оно представляет собой важный инструмент улучшения процессов и коммуникации, что в конечном итоге приводит к повышению эффективности и улучшению предоставления услуг.

Данная программно-информационная система предназначена для организации деятельности предприятия быстрого питания.

Для достижения цели необходимо решить следующие задачи:

1. Разработать модель данных для хранения информации о товарах, продуктах, сотрудниках, покупателях, продажах, заказах и их состоянии.
2. Разработать алгоритм для осуществления продажи.
3. Разработать алгоритм для управления каталогом товаров, склада, учетными данными сотрудников.
4. Разработать программно-информационную систему для организации деятельности предприятия быстрого питания.

2.3 Требования к программной системе

2.3.1 Требования к данным программно-информационной системы

Требования к данным программного обеспечения информационных систем приема и обработки заказов в сфере быстрого питания являются ключевыми для разработки и внедрения этой системы, так как от общей эффективности системы зависит качество обслуживания клиентов. Основными требованиями к данным являются точность, доступность, безопасность, актуальность и целостность.

Точность данных является ключевым требованием, поскольку ошибки в данных могут привести к неправильной обработке заказов, поломкам кухни или недостаточным инвестициям. Вся информация, поступающая в систему, должна быть проверена на корректность данных во избежание возможных недоразумений при обработке заказа.

Например, если покупатель выбрал конкретный продукт питания, система должна точно отображать его название, ингредиенты и цену, чтобы избежать путаницы. Аналогичным образом, информация на упаковке пищевых продуктов должна быть актуальной, чтобы повара и менеджеры могли принимать решения на основе достоверной информации.

Доступность данных также является важным требованием. Все пользователи системы, такие как кассиры, повара, менеджеры и администраторы, всегда должны иметь доступ к необходимой информации для эффективного выполнения своей работы.

Для этого система должна быть построена так, чтобы к данным можно было получить доступ и легко получить доступ за разумное время, при этом скорость обработки запросов должна быть высокой. Важно, чтобы информация о текущих заказах, наличии и обработке была доступна всему персоналу, задействованному в проекте, что позволяет избежать ошибок и задержек. Безопасность данных — еще одна важная область.

Система обрабатывает конфиденциальную информацию, такую как информация о платежах клиентов, информация о заказах и запасах, а также информация о персонале компании. Чтобы защитить эти данные от несанкционированного доступа, необходимо реализовать методы шифрования и авторизации.

Информационная система программного обеспечения должна включать в себя многоуровневую систему безопасности данных, чтобы гарантировать, что только авторизованные пользователи могут получить доступ к определенной информации, и что данные не уязвимы для атак или компрометации. Актуальность данных — это требование, которое требует постоянного обновления и точности.

Программное обеспечение и информационные системы должны быть настроены так, чтобы информация о складских запасах, статусе заказов и информации об оплате всегда была актуальной и точной. Например, при размещении заказа система должна немедленно обновить информацию о запасах, а также отобразить информацию о статусе оплаты. Это требует постоянной синхронизации компонентов системы, чтобы все данные были согласованными в любой момент времени.

Целостность данных означает, что данные должны сохраняться в исходном формате и не изменяться произвольно. Для этого в системе долж-

ны быть механизмы предотвращения потери и повреждения данных, а также обеспечения целостности базы данных.

Это может включать в себя постоянное резервное копирование данных, использование алгоритмов для обеспечения отсутствия ошибок и проверку правильности работы всей системы. Кроме того, важным требованием является возможность интеграции с другими системами, такими как платежные шлюзы, системы инвентаризации или службы доставки.

Чтобы вся экосистема предприятия функционировала нормально, данные из внешних источников должны быть должным образом обработаны и интегрированы в общую систему. Важно, чтобы вся информация, полученная от поставщиков или других внешних служб, была правильно отформатирована и соответствовала внутренним требованиям.

В целом требования к данным для программного обеспечения информационных систем приема и обработки заказов в сфере быстрого питания обеспечивают надежную и бесперебойную работу, а также способствуют снижению потенциальных ошибок и улучшению процессов обслуживания клиентов.

Система должна обеспечивать целостность, безопасность и актуальность данных, проверять целостность информации и обеспечивать легкий доступ для всех пользователей. На рисунке 2.1 представлена концептуальная модель данных программно-информационной системы в виде диаграммы сущность-связь.

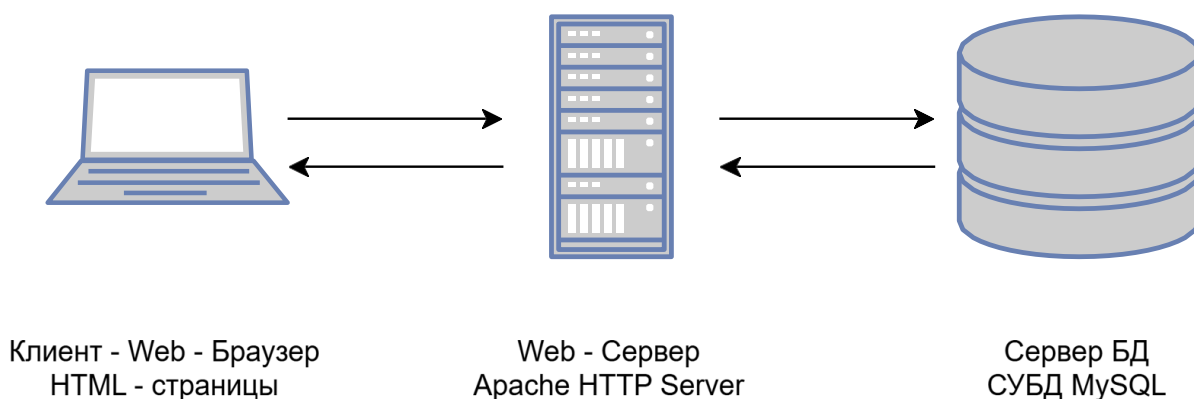


Рисунок 2.1 – Концептуальная модель данных

Входными данными для программной системы является база данных.

Выходными данными для программной системы является графическое изображение приложения.



Рисунок 2.2 – Схема веб-сервиса

Страница меню предназначена для просмотра ассортимента предприятия быстрого питания, цен. Также на странице с меню есть возможность добавить быстрое питание в корзину.

На рисунке 2.3 представлена диаграмма последовательности для кнопки «Добавить товар».

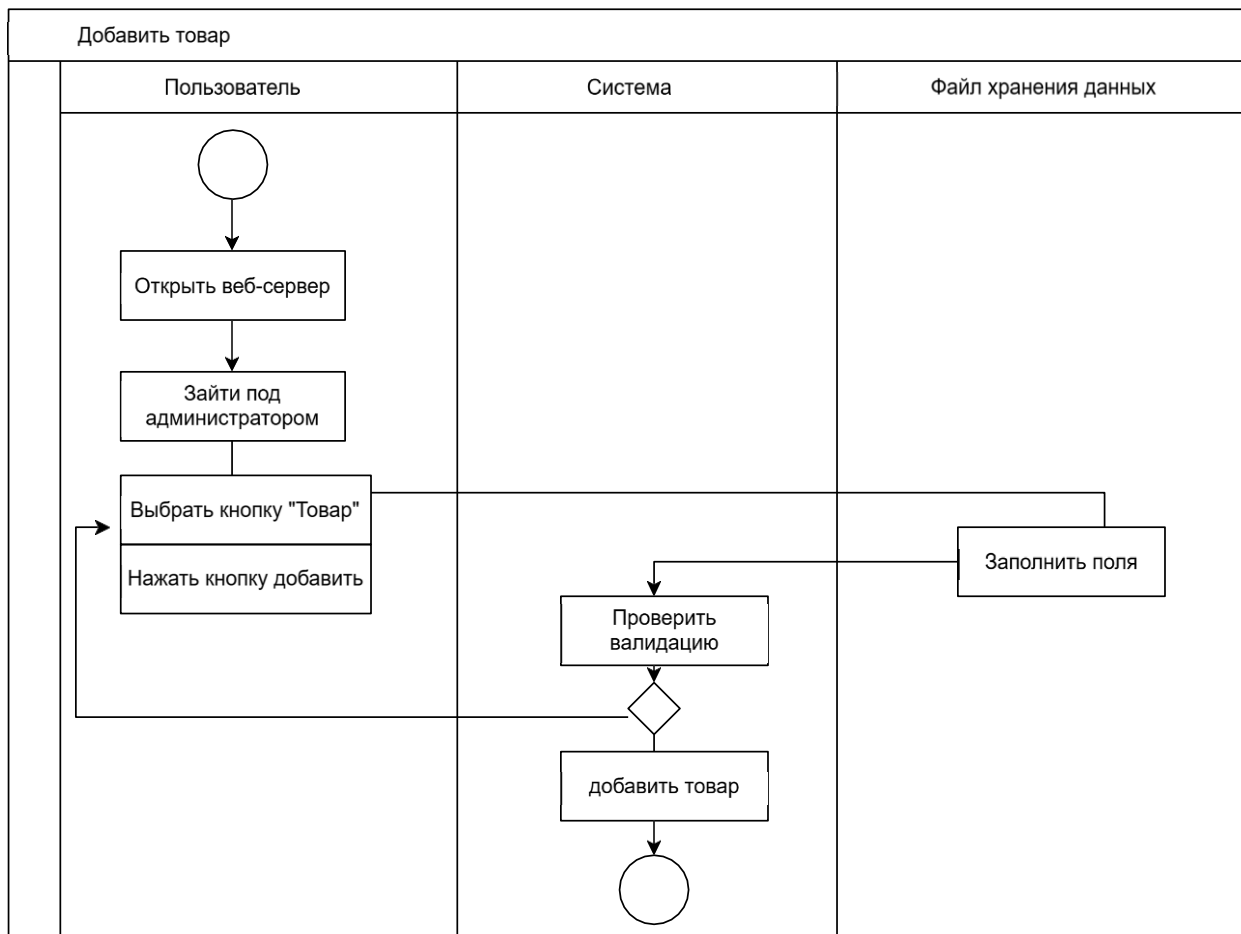


Рисунок 2.3 – Добавить товар

2.3.2 Функциональные требования к программной системе

Разрабатываемая программно-информационная система – это десктопное приложение, которое представляет следующие функции в программе:

- авторизация;
- добавление/редактирование/удаление позиции из меню;
- добавление/редактирование/удаление учетной записи клиента;
- просмотр списка пользователей;
- просмотр выполненных продаж;
- оформление заказа (продажи);
- просмотр очереди заказов;
- изменение статуса заказа;
- добавление/списание товаров со склада.

В данной системе присутствует два типа пользователей, с различными уровнями доступа к функционалу программного продукта: диаграмма прецедентов администратора (рис. 2.4), диаграмма прецедентов покупателя (2.5).



Рисунок 2.4 – Диаграмма прецедентов администратора

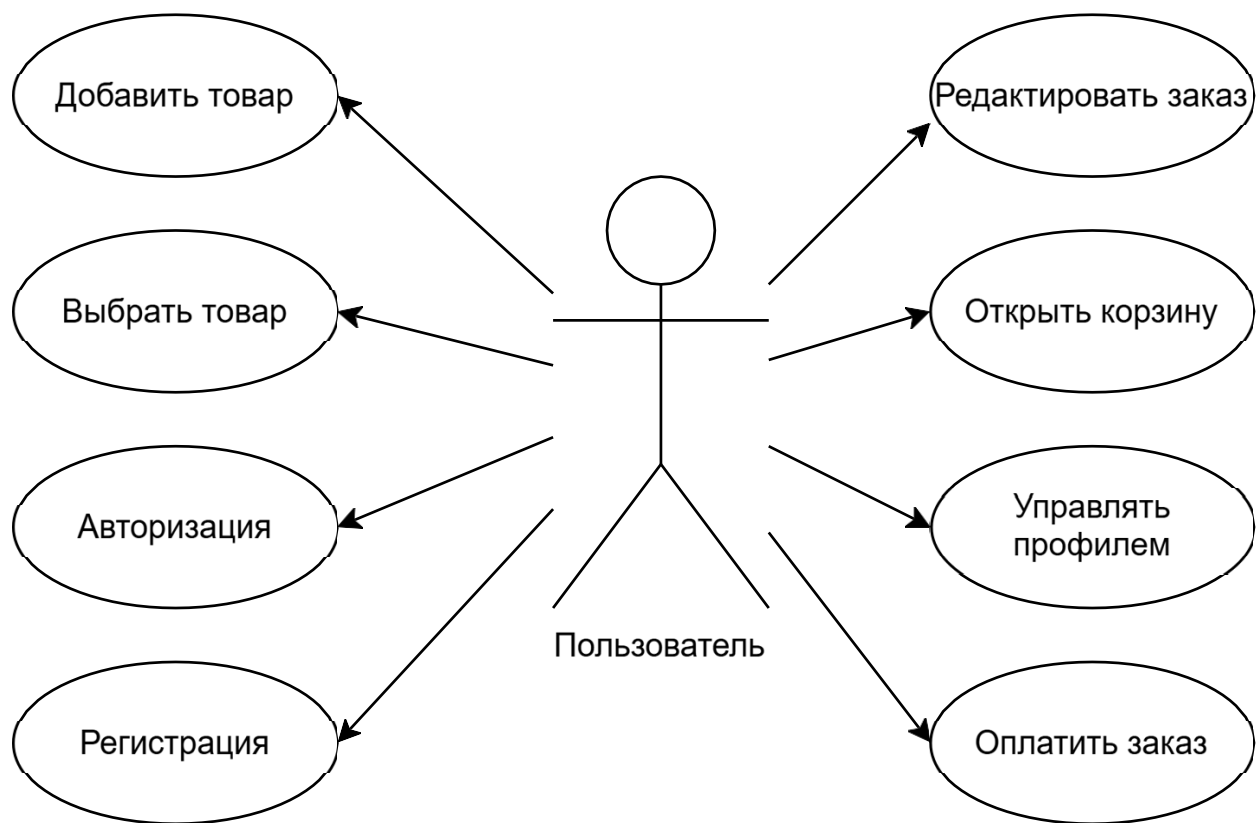


Рисунок 2.5 – Диаграмма прецедентов клиента

2.3.2.1 Вариант использования «Авторизация»

Заинтересованные лица и их требования: пользователь желает авторизоваться в системе.

Предусловие: открыта вкладка «Авторизация».

Постусловие: переход в главное меню.

Основной успешный сценарий:

1. Пользователь нажимает кнопку авторизация.
2. Пользователь вводит логин и пароль.
3. Пользователь нажимает кнопку войти.
4. Сервер формирует запрос в базу данных и передает результат в приложения.
5. Пользователь заходит в приложение.

2.3.2.2 Вариант использования «Регистрация»

Заинтересованные лица и их требования: пользователь, который хочет зарегистрироваться на сайте.

Предусловие: открыта вкладка «Регистрация».

Основной успешный сценарий:

1. Пользователь нажимает кнопку регистрация.
2. Пользователь вводит логин и пароль, адрес электронной почты.
3. Пользователь нажимает кнопку зарегистрироваться.
4. Сайт добавляет данные в базу данных.

2.3.2.3 Вариант использования «Добавить товар»

Заинтересованные лица и их требования: пользователь хочет добавить товар в корзину.

Предусловие: открыта вкладка товар.

Постусловие: пройдена авторизация.

Основной успешный сценарий:

1. Пользователь нажимает кнопку добавить товар.
2. Система перемещает выбранный товар в корзину.

2.3.2.4 Вариант использования «Добавить еду»

Заинтересованные лица и их требования: пользователь хочет добавить еду в корзину.

Предусловие: открыта вкладка «Еда».

Постусловие: пройдена авторизация.

Основной успешный сценарий:

1. Пользователь нажимает кнопку добавить еду.
2. Система перемещает выбранный вид еду в корзину.

2.3.2.5 Вариант использования «Добавить заказ»

Заинтересованные лица и их требования: пользователь хочет добавить заказ в корзину.

Предусловие: открыта вкладка заказ.

Постусловие: пройдена авторизация.

Основной успешный сценарий:

1. Пользователь нажимает кнопку добавить заказ.
2. Система перемещает выбранный вид заказ в корзину.

2.3.2.6 Вариант использования «Очистить корзину»

Заинтересованные лица и их требования: пользователь желает удалить все из корзины.

Предусловие: открыта вкладка «корзина».

Постусловие: очищена корзина.

Основной успешный сценарий:

1. Пользователь нажимает на корзину.
2. Пользователь очищает корзину, удалив услугу.
3. Веб-сервис показывает пустую корзину.

2.3.2.7 Вариант использования «Выйти из аккаунта»

Заинтересованные лица и их требования: пользователь желает выйти из аккаунта.

Предусловие: открыта вкладка аккаунт.

Постусловие: кнопка войти и зарегистрироваться.

Основной успешный сценарий:

1. Пользователь нажимает на кнопку «Аккаунт».
2. Пользователь видит информацию о себе.
3. Пользователь нажимает на кнопку выйти из аккаунта.

2.3.2.8 Вариант использования «Оплатить»

Заинтересованные лица и их требования: пользователь желает оплатить выбранную услугу.

Предусловие: открыта вкладка корзина.

Постусловие: оплачен заказ.

Основной успешный сценарий:

1. Пользователь нажимает на кнопку «Корзина».
2. Пользователь нажимает оплатить заказ.
3. Пользователь заполняет нужные поля.
4. Система фиксирует заказ.

2.3.2.9 Вариант использования «Добавление учетной записи пользователя»

Заинтересованные лица и их требования: администратор желает добавить учетную запись клиента.

Предусловие: открыта вкладка профиль.

Постусловие: добавлен покупатель.

Основной успешный сценарий:

1. Администратор нажимает кнопку «Редактировать профиль».
2. Администратор выбирает кнопку «Добавить профиль».
3. Администратор добавляет данные о профиле нового покупателя.

2.3.2.10 Вариант использования «Редактирование учетной записи покупателя»

Заинтересованные лица и их требования: администратор желает редактировать учетную запись клиента.

Предусловие: открыта вкладка профиль.

Постусловие: редактированы данные профиля.

Основной успешный сценарий:

1. Администратор нажимает кнопку «Редактировать профиль».

2. Администратор выбирает кнопку «Редактировать».
3. Администратор изменяет данные о профиле нового покупателя.

2.3.2.11 Вариант использования «Удалить учетную запись профиля»

Заинтересованные лица и их требования: администратор желает удалить учетную запись клиента.

Предусловие: открыта вкладка профиль.

Постусловие: удален профиль клиента.

Основной успешный сценарий:

1. Администратор нажимает кнопку «Редактировать профиль».
2. Администратор выбирает кнопку «Удалить профиль».
3. Администратор удаляет профиль из системы.
4. В базе данных исчезает запись о профиле.

2.3.2.12 Вариант использования «Закрыть сайт»

Заинтересованные лица и их требования: пользователь желает выйти из сайта.

Предусловие: открыт сайт.

Основной успешный сценарий:

1. Пользователь нажимает на кнопку «Выйти с сайта».
2. Пользователь выходит из сайта.

2.3.3 Требования к пользовательскому интерфейсу программной системы

В программно-информационной системе для организации деятельности предприятия быстрого питания должен быть реализован интерфейс.

Интерфейс включает:

- форма «авторизации»;
- форма «регистрации»;
- вкладка «Меню» для добавления и редактирования позиций меню;

- вкладка «Заказ» для добавления и редактирования заказов;
- вкладка «Еда» для добавления и редактирования еды;
- вкладка «Корзина» для просмотра своего заказа и его оформлении;
- вкладка «Профиль» для просмотра информации о клиенте;
- вкладка «Настройки» для редактирования заказа.

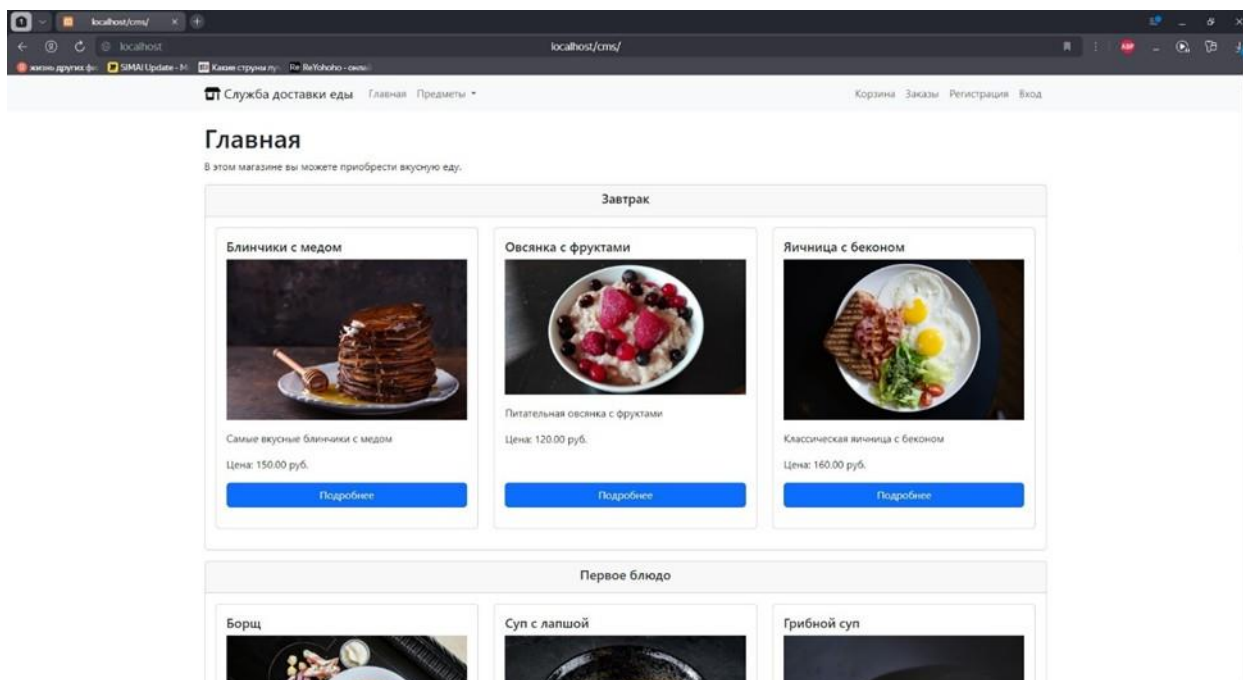


Рисунок 2.6 – Макет интерфейса окна «Главное меню программы».

2.3.4 Нефункциональные требования к программной системе

Требования к аппаратной совместимости:

Пассивные требования к программному обеспечению и информационным системам приема и обработки заказов в заведении быстрого питания играют важную роль в обеспечении его эффективной работы и удовлетворения потребностей своих пользователей в сфере управления. Эти требования не касаются конкретных функций системы, а скорее определяют, как система должна работать с точки зрения производительности, надежности, безопасности и других важных аспектов.

Одним из ключевых нефункциональных требований является производительность системы. Программно-информационная система должна обес-

печивать быструю обработку запросов и заказов. В среде предприятий быстрого питания, где заказы поступают постоянно и их необходимо обрабатывать без задержек, важно, чтобы система могла эффективно обрабатывать большие объемы данных и одновременно обслуживать большое количество клиентов.

Это особенно важно в часы пик, когда нагрузка на систему увеличивается и вам необходимо без сбоев и задержек поддерживать сложный рабочий процесс. Надежность системы также является важным требованием.

Безопасность — еще один важный аспект пассивных требований. Программное обеспечение и информационные системы для приема и обработки заказов в заведениях быстрого питания работают с конфиденциальной информацией, такой как информация о платежах клиентов, информация о заказах и персонале, что требует применения высоких стандартов безопасности.

Система должна защищать данные от несанкционированного доступа, взлома или утечек. Для достижения этой цели необходимо использовать методы шифрования, а также системы авторизации и аутентификации пользователей, чтобы гарантировать, что только авторизованные пользователи могут получить доступ к их важной информации.

Гибкость системы также важна, особенно если компания расширяет свою деятельность или увеличивает объем заказов. Программно-информационная система должна быть масштабируемой, то есть эффективно справляться с увеличением объема данных, количества пользователей и нагрузки без потери производительности.

Это важно как для растущего бизнеса, так и для крупных сетей быстрого питания с несколькими точками продаж. Масштабируемость позволяет системе адаптироваться к меняющимся обстоятельствам, не требуя полной перестройки.

Важным требованием также является простота использования системы. Интерфейс системы должен быть интуитивно понятным и простым в использовании, чтобы персонал компании мог быстро и эффективно оптимизировать его при выполнении заказов.

Важно адаптировать пользовательский интерфейс под разные уровни пользователей: от инвесторов до менеджеров, каждый из которых использует систему для выполнения определенных задач.

Это, в свою очередь, влияет на скорость обучения сотрудников, что, в свою очередь, влияет на общее обслуживание клиентов. Функциональная совместимость — еще одно требование, означающее, что система должна быть совместима с другими программными решениями, которые можно использовать на предприятии.

Например, система должна быстро интегрироваться с платежными шлюзами, системами инвентаризации, транспортными путями или системами учета. Это упрощает взаимодействие между различными бизнес-компонентами и обеспечивает единую экосистему для обработки данных и управления системами. Также важным пассивным требованием является поддержка программы.

Информационная система программного обеспечения должна легко поддерживаться, а это означает, что модификации, исправления ошибок и улучшения могут быть внесены без затрат времени и ресурсов.

Система должна быть спроектирована таким образом, чтобы новые версии можно было использовать без остановки проекта, а потенциальные ошибки можно было быстро исправить с минимальными последствиями для пользователей.

Наконец, важным требованием является совместимость с различными устройствами и операционными системами. При использовании терминалов самообслуживания, мобильных приложений и веб-платформ система должна обеспечивать бесперебойную работу всех устройств и операционных систем, позволяя обслуживать клиентов по всем каналам.

В результате пассивные требования к программным информационным системам для приема и обработки заказов в индустрии быстрого питания влияют на такие факторы, как производительность, надежность, безопасность, масштабируемость, несоответствие требованиям, сложность использования, возможность подключения, поддержку и совместимость устройств.

Эти требования помогают обеспечить эффективность и безопасность системы, что, в свою очередь, способствует качественному обслуживанию клиентов и успешной работе. Приложение можно установить на персональный компьютер с процессором Intel Celeron или более мощным, минимум 2 ГБ оперативной памяти и минимум 50 МБ свободного места на жестком диске.

Требования к программной совместимости:

Операционная система: Windows 10

Информация о меню, сотрудниках, клиентах, заказах и запасах на складе должна предоставляться через базу данных. Для использования базы данных необходимо использовать СУБД Microsoft SQL Server.

Требования к программному обеспечению информационной системы приема и обработки заказов в заведении быстрого питания означают, что система должна иметь возможность эффективно взаимодействовать и функционировать с другими программными решениями и платформами в различных сферах деятельности.

В современной бизнес-среде, где часто разворачивается несколько программных продуктов, обеспечение совместимости продуктов становится ключом к успеху в бизнесе. Прежде всего, система должна быть совместима с операционными системами, на которых она будет работать.

Сюда входит поддержка популярных операционных систем для настольных компьютеров, таких как Windows, Linux и macOS, а также мобильных операционных систем, таких как Android и iOS. Совместимость с этими операционными системами позволяет вам более гибко выбирать, какие инструменты и программные платформы использовать, а также расширять возможности системы для разных пользователей (например, для инвесторов, сотрудников или клиентов через мобильные приложения).

Кроме того, система программного обеспечения должна быть совместима с базами данных. Это позволяет системе беспрепятственно работать с различными решениями для хранения и обработки данных, такими как MySQL, PostgreSQL, Microsoft SQL Server, Oracle и другими.

Важно, что система может интегрироваться с текущими базами данных предприятия, обеспечивая бесперебойность работы и согласованность данных, а также возможность масштабирования по мере увеличения объема информации. Также важна совместимость с внешними сервисами и приложениями.

Системы приема и обработки заказов должны иметь возможность интегрироваться с платежными шлюзами, системами инвентаризации, CRM-системами, системами доставки, а также системами бухгалтерского учета и бюджетирования. Это помогает автоматизировать процессы, улучшает взаимодействие с внешними сервисами и снижает количество ошибок, которые могут возникнуть при ручной передаче данных между системами.

Совместимость программного обеспечения также включает в себя способность системы работать в различных сетевых конфигурациях. Система должна поддерживать как локальные соединения, так и удаленные соединения, что особенно важно для крупных сетей ресторанов быстрого питания, работающих в разных локациях. Для этого необходимо обеспечить производительность и безопасность системы во всех сетях, а также возможность удаленного мониторинга и управления системой.

Поэтому эти требования гарантируют бесперебойную работу системы в различных условиях, обеспечивая бесперебойное обслуживание клиентов и бесперебойное выполнение всех процессов проекта.

2.4 Требования к оформлению документации

Требования к стадиям разработки программ и программной документации для вычислительных машин, комплексов и систем независимо от их назначения и области применения, этапам и содержанию работ устанавливаются ГОСТ 19.102-77 и ГОСТ 34.601-90.

Программная документация должна включать в себя:

- техническое задание;
- технический проект;
- рабочий проект.

3 Технический проект

3.1 Общие сведения о программно-информационной системе

Программно-информационная система заказов на предприятии быстрого питания. После нажатия кнопки «Войти» отправляются запрос в базу данных для сопоставления введенных данных имеющемуся списку сотрудников. При успешном результате в зависимости от занимаемой должности сотрудником загрузится интерфейс программы с необходимым функционалом.

3.2 Проект данных программно-информационной системы

3.2.1 Структура базы данных

На основании анализа предметной области из технического задания была разработана база данных.

База данных включает в себя следующие таблицы:

- Administrator – администратор;
- Users – покупатели;
- Products – меню;
- Profile – профиль;
- Basket – корзина;
- Orders – статус заказов;
- Sells – связывает товар с чеком.

В таблице 3.1 представлены атрибуты таблицы Administrator.

Таблица 3.1 – Атрибуты таблиц Administrator

Название атрибута	Описание
AdministratorID	Идентификатор администратора
Name	Имя
Password	Пароль
Phone	Номер телефона

Продолжение таблицы 3.1

Название атрибута	Описание
Address_Flat	Номер квартиры
Address_House	Номер дома
Address_Street	Улица
IsDeleted	Статус учетной записи

В таблице 3.2 представлены атрибуты таблицы User.

Таблица 3.2 – Атрибуты таблиц User

Название атрибута	Описание
UserID	Идентификатор администратора
Name	Имя
Password	Пароль
Phone	Номер телефона
Address_Flat	Номер квартиры
Address_House	Номер дома
Address_Street	Улица
IsDeleted	Статус учетной записи

В таблице 3.3 представлены атрибуты таблицы Product.

Таблица 3.3 – Атрибуты таблиц Product

Название атрибута	Описание
ProductID	Идентификатор товара
Name	Наименование товара
Price	Цена
IsDeleted	Статус товара

В таблице 3.4 представлены атрибуты таблицы Profile.

Таблица 3.4 – Атрибуты таблиц Profile

Название атрибута	Описание
UserID	Идентификатор покупателя
Name	Имя
Password	Пароль
Phone	Номер телефона
Address_Flat	Номер квартиры
Address_House	Номер дома
Address_Street	Улица
Deleted	Статус
Redacture	Редактирование

В таблице 3.5 представлены атрибуты таблицы Basket.

Таблица 3.5 – Атрибуты таблиц Basket

Название атрибута	Описание
BasketID	Идентификатор корзины
AdministrationID	Идентификатор администратора
UserID	Идентификатор покупателя
DataCreated	Дата заказа
Price	Итоговая сумма в чеке

В таблице 3.6 представлены атрибуты таблицы Sell.

Таблица 3.6 – Атрибуты таблиц Sell

Название атрибута	Описание
SellID	Идентификатор продажи
CheckID	Идентификатор чека
ProductID	Идентификатор товара

В таблице 3.7 представлены атрибуты таблицы Order.

Таблица 3.7 – Атрибуты таблиц Order

Название атрибута	Описание
OrderID	Идентификатор заказа
CheckID	Идентификатор чека
IsCompleted	Статус готовности заказа
IsDeliver	Статус исполнения заказа

3.3 Проектирование архитектуры программно-информационной системы

3.3.1 Компоненты программно-информационной системы

Программно-информационная система имеет сложную иерархическую структуру.

Во главе находится файл header-главная часть программы. От главной части программы идет administration-администратор и User-покупатель(клиент).

Администратор добавляет товар.

Клиент выбирает позицию, добавляет в корзину, делает заказ.

На рисунке 3.1 показана диаграмма компонентов программной системы.

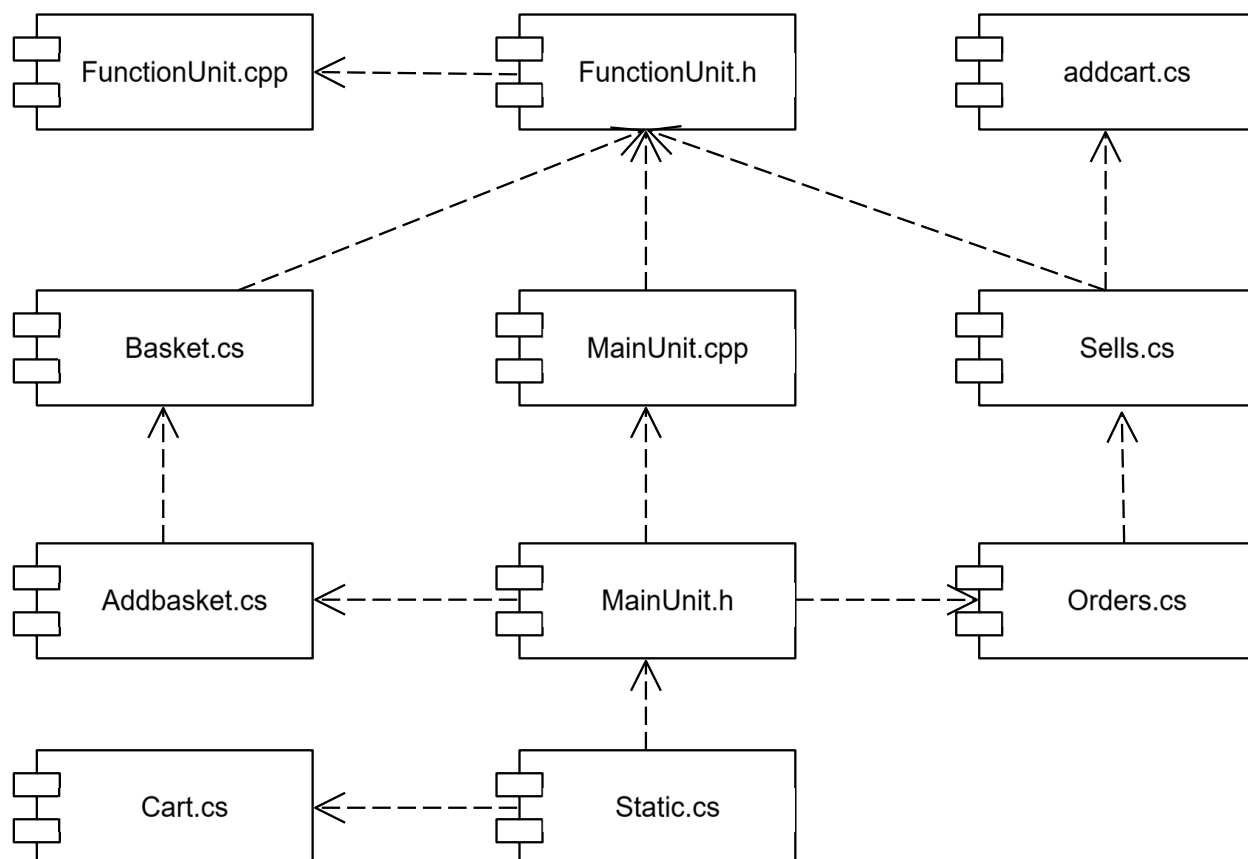


Рисунок 3.1 – Диаграмма компонентов программно-информационной системы

MainUnit.h

Компонент MainUnit.h отвечает за управление конфигурацией и административные функции. Администратор имеет разрешение управлять всеми аспектами информационной системы программного обеспечения, включая добавление, редактирование и удаление элементов из меню, управление пользователями (например, блокировку или создание учетных записей других), будет разрабатывать программу, просматривать отчеты о проверках и продажах, а также просмотр заказов. Эта функция позволяет поддерживать данные в актуальном состоянии и обеспечивать бесперебойную работу системы.

FunctionsUnit.cpp

Компонент FunctionsUnit представляет пользователей системы, которые взаимодействуют с ней для создания объектов. Эта функция включает в себя регистрацию и лицензирование клиентов, хранение личных данных,

историю заказов и предпочтения. Клиентам предоставляется интуитивно понятный интерфейс, позволяющий быстро найти понравившийся товар в меню, добавить его в корзину и оформить заказ.

Cart.cs

Компонент Cart отвечает за список предлагаемых товаров и услуг. Включает полное описание продукта, такое как название, цена, состав, калорийность, акции или скидки. Администратор может редактировать меню: добавлять новые позиции, изменять существующие или удалять устаревшие позиции. Клиенты используют эту функцию для выбора заказов.

Static.cs. Хранит и управляет личной информацией пользователя. Информация о клиенте может включать в себя такую информацию, как имя, контактную информацию (телефон, адрес электронной почты), адрес доставки, предпочтения и историю заказов. Эта функция оптимизирует работу системы, позволяя клиентам быстро размещать заказы на основе их прошлой активности или выявленных предпочтений

Basket.cs

Компонент Basket.cs позволяет пользователям накапливать запланированные заказы. В корзине перечислены выбранные товары с указанием количества и цены. Пользователи могут изменять содержимое корзины: добавлять новые товары, удалять ненужные или изменять количество уже добавленных товаров. После выбора товаров корзина отправляет информацию о заказе.

Orders.cs

Компонент "Статус заказов" отвечает за управление и отображение информации о текущих и прошлых заказах. Он позволяет покупателям отслеживать статус их заказа, например, "принят" "готовится" "в пути" или "доставлен". Администратор или сотрудники предприятия используют этот компонент для обновления статусов заказов, что помогает клиентам получать актуальную информацию.

Раздел Orders.cs «Статус заказа» отвечает за мониторинг и отображение текущей и прошлой информации. Это позволяет клиентам отслеживать

статус своего заказа, например «принято», «в ремонте», «в пути» или «доставлено». Менеджеры или сотрудники компании используют эту функцию для обновления статуса заказа, помогая клиентам оставаться в курсе событий. Sells.cs – связывает товар с чеком.

Sells.cs – связывает товар с чеком. Раздел «Продажи» выступает связующим звеном между позициями из меню и методом выдачи чеков. Отвечает за составление списка товаров, включенных в заказ, с учетом цены, количества и возможных скидок. Эта функция предоставляет квитанцию, которую можно отправить клиенту в цифровом или физическом виде. Он также используется для записи и анализа продаж.

Диаграмма развертывания программно-информационной системы показана на рисунке 3.2.

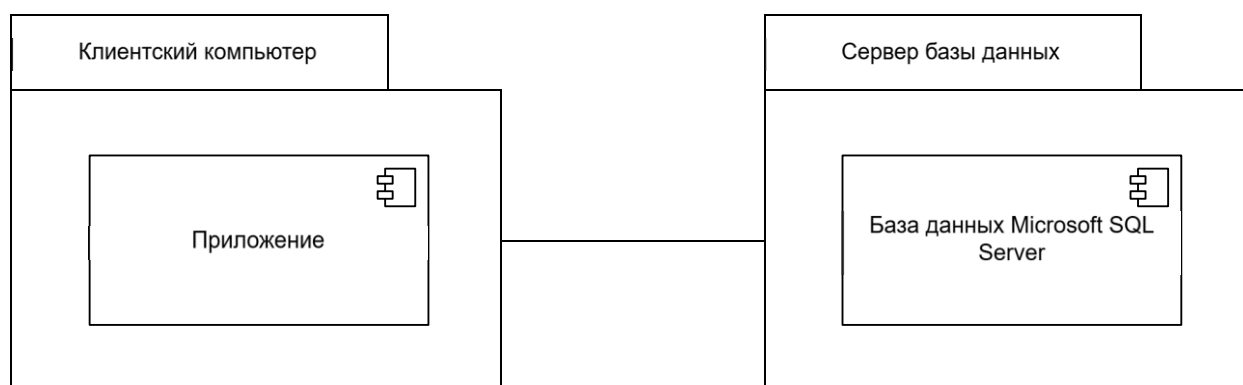


Рисунок 3.2 – Диаграмма развертывания

3.3.2 Архитектура программно-информационной системы

На рисунке 3.3 изображена общая диаграмма классов.

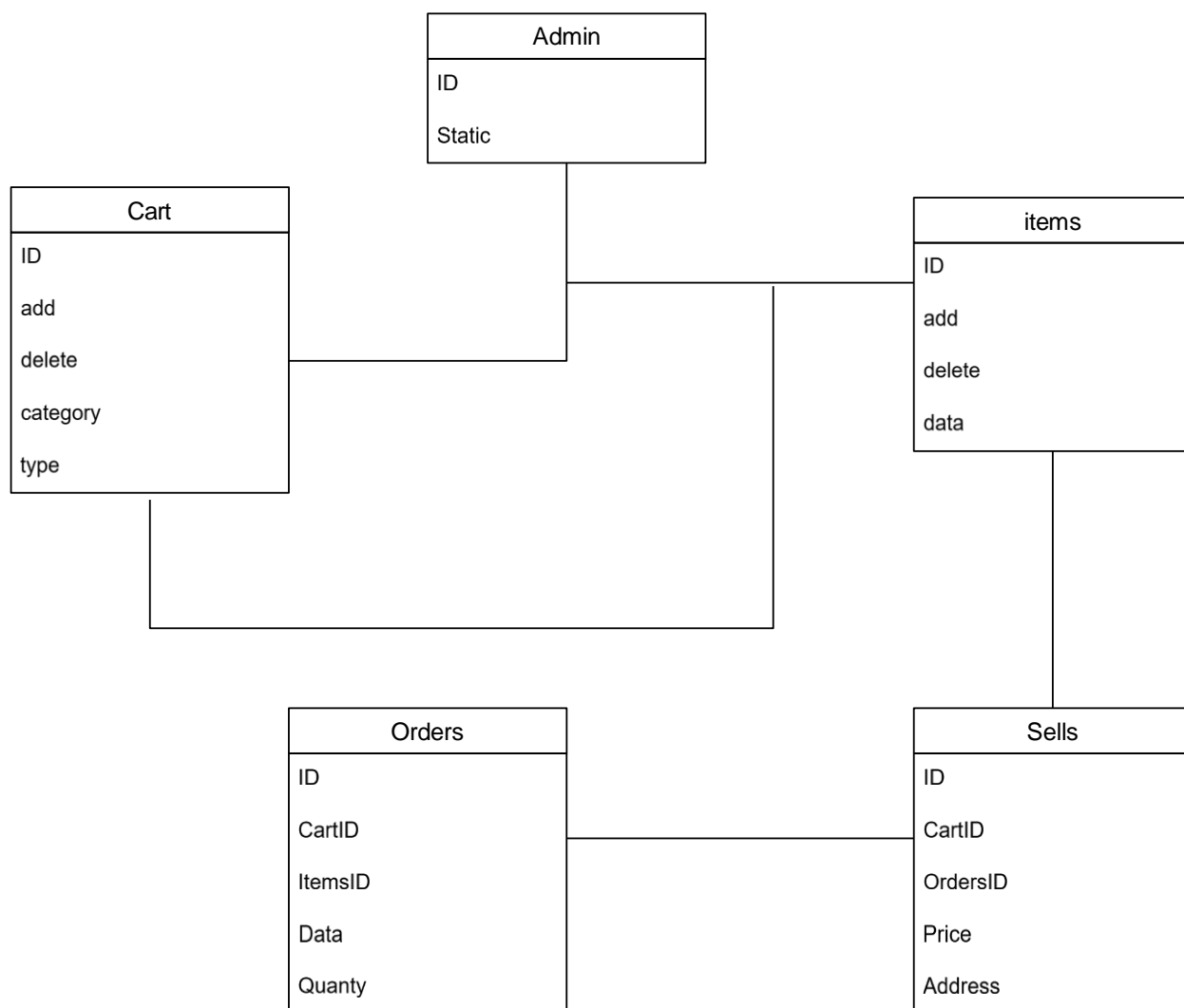


Рисунок 3.3 – Общая диаграмма классов

3.4 Проектирование пользовательского интерфейса программно-информационной системы

На основе требований к пользовательскому интерфейсу, представленных в пункте 2.3.3 технического задания, с помощью технологии Windows Forms был разработан пользовательский интерфейс.

В таблице 3.8 представлено описание объектов данного интерфейса.

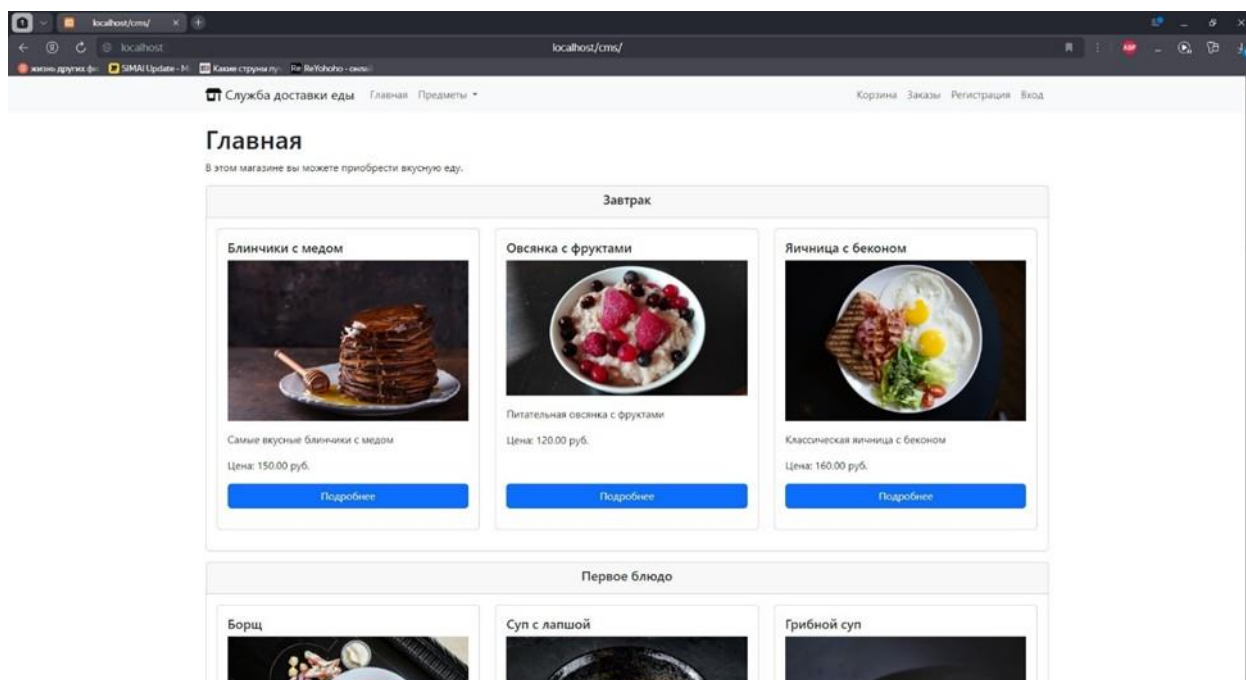


Рисунок 3.4 – Интерфейс главной формы

Таблица 3.8 – Описание объектов интерфейса главной формы

Номер объекта	Тип объекта	Имя объекта	Описание действия объекта
1	Кнопка	Главная	Отображает главную страницу интерфейса с обзором меню, акций и основных возможностей системы. Позволяет вернуться к стартовой странице из любого раздела.
2	Кнопка	Корзина	Открывает раздел, где пользователь видит список добавленных в корзину товаров, может редактировать количество, удалять позиции и оформлять заказ.
3	Кнопка	Заказы	Перенаправляет пользователя на страницу, где отображается информация о текущих и прошлых заказах, включая статус заказа и его детали.
4	Кнопка	Еда	Показывает полный каталог меню, включая категории товаров, их описание, изображения и возможность добавления в корзину.

Продолжение таблицы 3.8

5	Кнопка	Регистрация	Открывает форму регистрации нового пользователя, где вводятся данные, такие как имя, номер телефона, адрес и пароль. После регистрации создаётся профиль клиента.
6	Кнопка	Вход	Открывает форму авторизации, где пользователь вводит логин и пароль для входа в систему. При успешной авторизации открывается доступ к персонализированным функциям.

Код описания таблиц представлен ниже:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Описание объектов интерфейса</title>
7 <style>
8 table {
9 border-collapse : collapse ;
10 width: 100%;
11 }
12 th, td {
13 border: 1px solid #ddd;
14 padding: 8px;
15 text-align : left ;
16 }
17 th {
18 background-color : #f2f2f2 ;
19 }
20 </style>
21 </head>
22 <body>
23 <h1>Описание объектов интерфейса</h1>
24 <table>
25 <thead>
26 <tr>
27 <th>Номер объекта</th>
28 <th>Тип объекта</th>
29 <th>Имя объекта</th>
30 <th>Описание действия объекта</th>
31 </tr>
32 </thead>
33 <tbody>
34 <?php
35 // Массив объектов интерфейса
36 $objects = [

```

```

37 ['1', 'Кнопка', 'Главная', 'Отображает главную страницу интерфейса с обзором
    меню, акций и основных возможностей системы. Позволяет вернуться к
    стартовой странице из любого раздела.'],
38 ['2', 'Кнопка', 'Корзина', 'Открывает раздел, где пользователь видит список
    добавленных в корзину товаров, может редактировать количество, удалять
    позиции и оформлять заказ.'],
39 ['3', 'Кнопка', 'Заказы', 'Перенаправляет пользователя на страницу, где
    отображается информация о текущих и прошлых заказах, включая статус заказа
    и его детали.'],
40 ['4', 'Кнопка', 'Еда', 'Показывает полный каталог меню, включая категории
    товаров, их описание, изображения и возможность добавления в корзину.'],
41 ['5', 'Кнопка', 'Регистрация', 'Открывает форму регистрации нового
    пользователя, где вводятся данные, такие как имя, номер телефона, адрес и
    пароль. После регистрации создаётся профиль клиента.'],
42 ['6', 'Кнопка', 'Вход', 'Открывает форму авторизации, где пользователь вводит
    логин и пароль для входа в систему. При успешной авторизации открывается
    доступ к персонализированным функциям.'];
43 ];
44
45 // Вывод данных в таблицу
46 foreach ($objects as $object) {
47     echo "<tr>";
48     echo "<td>{$object[0]}</td>";
49     echo "<td>{$object[1]}</td>";
50     echo "<td>{$object[2]}</td>";
51     echo "<td>{$object[3]}</td>";
52     echo "</tr>";
53 }
54 ?>
55 </tbody>
56 </table>
57 </body>
58 </html>

```

ЗАКЛЮЧЕНИЕ

В результате выполнения работы были выполнены задачи, поставленные в техническом задании полном объёме.

Были решены следующие задачи:

1. Разработана модель данных для программно-информационной системы приема и обработки заказов предприятия быстрого питания.
2. Разработан алгоритм для осуществления продажи.
3. Разработан алгоритм для просмотра сформированного заказа и его редактирования.
4. Разработана программно-информационная система приема и обработки заказов предприятия быстрого питания.

Разработанный программный продукт можно использовать для организации деятельности предприятия быстрого питания. Эти факты являются безусловными достоинствами данной программно-информационной системы.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Шаблоны корпоративных приложений / М. Фаулер, Д. Райс, М. Фоммел, Э. Хайет [и др.] – Москва : Вильямс, 2018. – 544 с. – ISBN 978-5-6041394-0-0. – Текст : непосредственный.
2. Приемы объектно-ориентированного проектирования / Э. Гамма. Р. Хелм, Р. Джонсон, Дж. Влиссивус. – Санкт-Петербург : Питер, 2017. – 368 с. – ISBN 978-5-496-00389-6. – Текст : непосредственный.
3. Ларман, К. Применение UML 2.0 и шаблонов проектирования / К. Ларман. – Москва : Вильямс, 2020. – 736 с. – ISBN 978-5-907144-36-1. – Текст : непосредственный.
4. Фаулер, М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования / М. Фаулер. – Санкт-Петербург : Символ Плюс, 2013. – 192 с. – ISBN 978-5-93286-060-1. – Текст : непосредственный.
5. Гамма, Э. Паттерны объектно-ориентированного проектирования / Э. Гамма, Р. Джонсон, Р. Хелм. – Санкт-Петербург : Питер, 2021. – 448 с. – ISBN 978-5-4461-1595-2. – Текст : непосредственный.
6. Троелсен, Э. Язык программирования C# 5.0 и платформа .NET 4.5 / Э. Троелсен. – Москва : Вильямс, 2013. – 1312 с. – ISBN 978-5-8459-1814-7. – Текст : непосредственный.
7. MSDN – сеть разработчиков Microsoft : сайт. – URL: <https://msdn.microsoft.com> (дата обращения: 17.11.2021). – Текст : электронный.
8. Прайс, М. C# 7 и .NET Core. Кросс-платформенная разработка для профессионалов / М. Прайс. – Санкт-Петербург : Питер, 2017. – 640 с. – ISBN 978-5-4461-0516-8. – Текст : непосредственный.
9. Троелсен, Э. Язык программирования C# 7 и платформы .NET и .NET Core / Э. Троелсен, Ф. Джепикс. – Москва : Вильямс, 2018. – 1328 с. – ISBN 978-5-6040723-1-8. – Текст : непосредственный.

10. Албахари, Дж. С# 7.0. Справочник. Полное описание языка / Дж. Албахари, Б. Албахари. – Москва : Вильямс, 2018. – 1024 с. – ISBN 978-5-6040043-7-1. – Текст : непосредственный.
11. Джон, С. С# для профессионалов. Тонкости программирования / С. Джон. – Москва : Вильямс, 2019. – 608 с. – ISBN 978-5-907114-62-3. – Текст : непосредственный.
12. Head First. Паттерны проектирования. / Э. Фримен, Э. Робсон, К. Сиерра, Б. Бейтс. – Санкт-Петербург : Питер, 2018. – 656 с. – ISBN 978-5-4461-1034-6. – Текст : непосредственный.
13. Тепляков, С. Паттерны проектирования на платформе .NET / С. Тепляков. – Санкт-Петербург : Питер, 2018. – 320 с. – ISBN 978-5-4461-0843-5. – Текст : непосредственный.
14. Мартин, Р. С. Принципы, паттерны и методики гибкой разработки на языке С# / С. Р. Мартин, М. Мартин ; пер. с англ. А. Слинкин. – Санкт-Петербург : Символ-Плюс, 2011. – 757 с. – ISBN 978-5-93286-197-4. – Текст : непосредственный.
15. Криспин, Л. Гибкое тестирование : практическое руководство для тестировщиков ПО и гибких команд / Л. Криспин, Д. Грегори ; пер. с англ. Н. А. Мухина. – Москва : Вильямс, 2010. – 463 с. – ISBN 978-5-8459-1625-9. – Текст : непосредственный.
16. Блэк, Р. Ключевые процессы тестирования. Планирование, подготовка, проведение, совершенствование / Р. Блэк. – Москва : Лори, 2017. – 544 с. – ISBN 978-5-85582-392-9. – Текст : непосредственный.
17. Майерс, Г. Искусство тестирования программ / Г. Майерс, К. Сандлер, Т. Баджетт. – Москва : Диалектика, 2020. – 272 с. – ISBN 978-5-907203-66-2. – Текст : непосредственный.
18. Мюллер, Р. Проектирование баз данных и UML / Р. Мюллер; пер. с англ. Е. Молодцова. – Москва : Лори, 2018. – 420 с. – ISBN 978-5-85582-322-6. – Текст : непосредственный.

19. Пайлон, Д. UML 2 для программистов / Д. Пайлон, Н. Питмен. – Санкт-Петербург : Питер, 2012. – 240 с. – ISBN 978-5-459-01684-0. – Текст : непосредственный.
20. Ларман, К. Применение UML и шаблонов проектирования : практическое руководство / К. Ларман. – Москва : Вильямс, 2013. – 736 с. – ISBN 978-5-8459-1185-8. – Текст : непосредственный.
21. Буч, Г. Введение в UML от создателей языка / Г. Буч, Д. Рамбо, И. Якобсон. – Москва : ДМК Пресс, 2015. – 496 с. – ISBN 978-5-97060-157-0. – Текст : непосредственный.
22. Паттерны проектирования / Э. Фримен, Э. Фримен, К. Сьерра, Б. Бейтс. – Санкт-Петербург : Питер, 2011. – 656 с. – ISBN 978-5-459-00435-9. – Текст : непосредственный.
23. Босуэлл, Д. Читаемый код, или Программирование как искусство / Д. Босуэлл, Т. Фаучер. – Санкт-Петербург : Питер, 2012. – 208 с. – ISBN 978-5-459-01188-3. – Текст : непосредственный.
24. Назаров, С. В. Читаемый код, или Программирование как искусство / С. В. Назаров. – Москва : Инфра-М, 2016. – 376 с. – ISBN 978-5-16-011753-9. – Текст : непосредственный.
25. Фримен, А. LINQ. Язык интегрированных запросов в C# 2010 для профессионалов / А. Фримен, Д. Раттц-мл. – Москва : Вильямс, 2011. – 656 с. – ISBN 978-5-8459-1701-0. – Текст : непосредственный.
26. Абрамян, М. Э. Технология LINQ на примерах : практикум с использованием электронного задачника Programming Taskbook for LINQ / М. Э. Абрамян. – Москва : ДМК Пресс, 2014. – 328 с. – ISBN 978-5-94074-981-3. – Текст : непосредственный.
27. Ошероув, Р. Искусство автономного тестирования с примерами на C# / Р. Ошероув. – Москва : ДМК Пресс, 2016. – 360 с. – ISBN 978-5-97060-415-1. – Текст : непосредственный.
28. Павловская, Т. А. C#. Программирование на языке высокого уровня / Т. А. Павловская. – Москва : Питер, 2015. – 432 с. – ISBN 978-5-496-00861-7. – Текст : непосредственный.

29. Шилдт, Г. Полное руководство. С# 4.0 / Г. – Москва : Вильямс, 2019. – 1056 с. – ISBN 978-5-907114-49-4. – Текст : непосредственный.

30. Албахари, Дж. С# 7.0. Карманный справочник / Дж. Албахари, Б. Албахари ; пер. с англ. Ю. Н. Артеменко. – Москва : Альфа-книга, 2020. – 224 с. – ISBN 978-5-9909446-1-9. – Текст : непосредственный.

