Изображение выглядит как графическая вставка, Графика, мультфильм

Автоматически созданное описание

**КУРСОВОЙ ПРОЕКТ**

**по дисциплине: МДК 01.01 Разработка программных модулей**

**Тема: Разработка программного модуля**

**информационной системы «Система управления ресторанным меню»**

**Специальность 09.02.07 «Информационные системы и программирование»**

|  |  |  |
| --- | --- | --- |
| **Выполнил студент(ка) группы 312ИС-22** | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Е.М. Киселева** |
| **Руководитель** | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Н.И. Кручинкина** |

**Москва 2024**

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc185516726)

[1 Анализ предметной области 5](#_Toc185516727)

[1.1 Анализ автоматизации системы управления ресторанным меню 5](#_Toc185516728)

[1.2 Обзор и анализ существующих программных решений 5](#_Toc185516729)

[1.3 Постановка задачи. Структура входной и выходной информации 9](#_Toc185516730)

[2 Проектирование и разработка модуля 10](#_Toc185516731)

[2.1 Построение модели системы 10](#_Toc185516732)

[2.2 Проектирование базы данных и ее структура. 13](#_Toc185516733)

[2.3 Реализация интерфейса приложения 14](#_Toc185516734)

[3 Отладка и тестирование модуля 17](#_Toc185516735)

[3.1 Перехват исключительных ситуаций 17](#_Toc185516736)

[3.2 Технология работы с приложением 19](#_Toc185516737)

[ЗАКЛЮЧЕНИЕ 21](#_Toc185516738)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 22](#_Toc185516739)

[ПРИЛОЖЕНИЕ 1 23](#_Toc185516740)

[ПРИЛОЖЕНИЕ 2 26](#_Toc185516741)

# **ВВЕДЕНИЕ**

В современном мире ресторанный бизнес подвергается значительным изменениям под воздействием различных факторов, таких как рост конкуренции, изменение потребительских предпочтений и стремительное внедрение новых технологий. В условиях, когда качество обслуживания и скорость работы становятся критически важными для успешного функционирования заведений общественного питания, поэтому актуальность автоматизации процессов управления меню так важна.

Системный подход к управлению меню охватывает разработку концепции блюд, их оптимизацию под запросы целевой аудитории, контроль качества продукции и анализ предпочтений потребителей.

Цель курсового проекта заключается в разработке программного модуля, который обеспечит автоматизацию процессов управления ресторанным меню.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Проанализировать существующие решения в области управления меню в ресторане.
2. Разработать функциональные требования к системе управления ресторанным меню.
3. Осуществить проектирование архитектуры программного модуля.
4. Реализовать программный модуль, включая интерфейс и функции управления меню.
5. Провести тестирование разработанной системы на предмет корректности работы и выявить возможные недоработки.

Объектом исследования являются процессы управления ресторанным меню, включая разработку концепции, планирование ассортимента, учет запасов, контроль качества и анализ потребительского поведения.

Предметом выступает процесс управления меню, а именно: автоматизация учета и анализа данных, внедрение современных технологий, оптимизация бизнес-процессов и повышение уровня сервиса.

В рамках курсового проекта планируется проведение всестороннего исследования предметной области, охватывающей тему, заявленную в работе. Будут сформулированы основные требования к проекту и проведен анализ существующих решений.

Во второй части проекта будет представлен детальный обзор используемых инструментов и разработанных в ходе работы модулей. Будет описана структура программной части проекта, включая характеристики ключевых программных модулей. Каждый инструмент и модуль будут рассмотрены с точки зрения их функциональности и вклада в общую архитектуру проекта.

Третья часть проекта будет посвящена вопросам отладки и тестирования программы. Результаты тестирования будут проанализированы с целью выявления и устранения возможных проблем.

В заключительной части проекта будут сформулированы общие выводы о проделанной работе, подведены итоги исследования. Особое внимание будет уделено анализу ключевых моментов проекта и перспектив его дальнейшего развития.

# **1 Анализ предметной области**

## **1.1 Анализ автоматизации системы управления ресторанным меню**

Автоматизация системы управления ресторанным меню — это важный аспект современного ресторанного бизнеса, который может значительно повысить эффективность работы, улучшить качество обслуживания и оптимизировать процессы.

Цель автоматизации системы управления ресторанным меню заключается в повышении эффективности работы заведения и улучшении качества обслуживания клиентов за счет оптимизации процессов создания, обновления и контроля меню.

Преимущества автоматизации:

1. Снижение ошибок: меньше человеческого фактора приводит к более точным заказам.
2. Повышение качества обслуживания: персонал может больше внимания уделять клиентам, а не рутинным задачам.
3. Автоматизация системы управления ресторанным меню - это стратегический шаг, который может значительно улучшить работу заведения и повысить удовлетворенность клиентов. Однако важно тщательно продумать процесс внедрения и учитывать все возможные риски.

## **1.2 Обзор и анализ существующих программных решений**

На сегодняшний день существует множество программных продуктов, предназначенных для автоматизации управления ресторанными меню. Рассмотрим наиболее популярные из них, оценим их функциональные возможности, достоинства и недостатки.

1. R-Keeper



Рисунок 1 – R-Keeper

R-Keeper - еще одно популярное решение для автоматизации ресторанов, разработанное российской компанией UCS. Модуль управления меню интегрирован в общую систему управления заведением. Окно входа представлено на рисунке 1.

Функциональные возможности:

* быстрое создание и изменение меню;
* подключение к кассовому оборудованию;
* контроль за стоимостью блюд и запасами;
* отчеты по продажам и статистике;
* мобильные приложения для клиентов.

Достоинства:

* простота использования и интуитивно понятный интерфейс;
* быстрая настройка и запуск;
* доступная цена по сравнению с аналогичными продуктами.

Недостатки:

* ограниченный набор аналитических инструментов;
* менее гибкая система настроек.

1. Poster



Рисунок 2 - Poster

Poster - облачная платформа для автоматизации малого и среднего ресторанного бизнеса. Предлагает удобный интерфейс для создания и управления меню (рисунок 2).

Функциональные возможности:

* онлайн-доступ к управлению меню;
* интеграция с кассовыми аппаратами и POS-системами;
* мониторинг запасов и автоматическое уведомление о необходимости пополнения;
* простой интерфейс для внесения изменений в меню.

Достоинства:

* облачное решение, не требующее установки специального оборудования;
* низкие требования к технической инфраструктуре;
* удобство использования для небольших заведений.

Недостатки:

* отсутствие некоторых продвинутых функций, характерных для крупных систем;
* зависимость от Интернет-соединения.

1. QuickResto

Изображение выглядит как текст, программное обеспечение, Значок на компьютере, Операционная система

Автоматически созданное описание

Рисунок 3 – QuickResto

QuickResto - российское решение для автоматизации ресторанного меню, ориентированное на малый и средний бизнес. Отличается простотой использования и доступной ценой. Его интерфейс представлен на рисунке 3.

Функциональные возможности:

* легкое создание и редактирование меню;
* автоматический расчет стоимости блюд;
* инструменты для управления запасами и инвентаризацией;
* поддержка мобильных приложений для клиентов и сотрудников.

Достоинства:

* низкая стоимость владения;
* быстрое освоение и начало работы;
* оптимально подходит для небольших кафе и ресторанов.

Недостатки:

* относительно скромный функционал по сравнению с крупными системами;
* ограниченная поддержка сложных сценариев работы.

## **1.3 Постановка задачи. Структура входной и выходной информации**

Процесс управления ресторанным меню включает несколько ключевых этапов, которые охватывают создание, обновление, контроль и анализ меню. Он направлен на решение задач по управлению ассортимента блюд, контролю за качеством продукции, оптимизации закупок ингредиентов, а также обеспечению прозрачности и доступности информации для всех участников процесса.

Основные компоненты информационного обеспечения:

1. База данных рецептов: содержит информацию о каждом блюде ресторана - его состав, ингредиенты и другие характеристики. Это позволяет отслеживать соответствие каждого блюда стандартам качества и оперативно вносить изменения при необходимости.
2. Данные об ингредиентах: включает информацию о поставщиках, стоимости, сроках хранения, наличии на складе и других параметрах, что помогает оптимизировать закупки и минимизировать издержки.
3. Системы аналитики и отчетности: позволяют анализировать ключевые показатели деятельности ресторана, такие как объем продаж, прибыль, затраты, удовлетворенность клиентов, что способствует принятию обоснованных решений по развитию бизнеса.

Таким образом, для успешной реализации системы управления ресторанным меню, позволяя обеспечить оперативное управление всеми аспектами деятельности заведения и принимать взвешенные решения на основе актуальной и достоверной информации.

# **2 Проектирование и разработка модуля**

## **2.1 Построение модели системы**

Диаграмма сценариев (или диаграмма вариантов использования) для ресторанного меню помогает визуализировать взаимодействие различных пользователей (клиентов, официантов, поваров и администраторов) с меню ресторана. Каждый сценарий описывает, как пользователь может использовать меню и взаимодействовать с системой. Сценарий описанный ниже представлен на рисунке 4.

Участники:

1. Клиент - просматривает меню.
2. Повар - готовит блюда, сверяет поставки.
3. Администратор - управляет доступом к меню, актуализирует его.

Сценарии использования:

1. Выбор блюда

* клиент просматривает меню;
* клиент может фильтровать меню по категориям, ценам.

1. Готовка блюд

* повар проверяет наличие ингредиентов;
* повар начинает готовить блюда.

1. Коррекция меню

* администратор добавляет, удаляет или изменяет позиции в меню;
* администратор обновляет информацию о ценах и наличии блюд.

Изображение выглядит как диаграмма, зарисовка, рисунок, линия

Автоматически созданное описание

Рисунок 4 – Диаграмма сценариев

Для создания ER-диаграммы (рисунок 5) для ресторанного меню нужно определить основные сущности и их взаимосвязи. В данном случае можно выделить следующие ключевые сущности:

1. Меню
2. Блюдо
3. Ингредиент
4. Состав блюда

Связи между сущностями:

1. Меню - Блюдо: Меню может содержать несколько блюд (1:М).
2. Блюдо - Ингредиент (через Состав блюда): Блюдо может включать несколько ингредиентов, и каждый ингредиент может входить в несколько блюд (M:М).

Изображение выглядит как диаграмма, линия, оригами

Автоматически созданное описание

Рисунок 5 – ER-диаграмма

Далее разработаем диаграмму IDEF1X, которая предоставит информацию о их сущностях и взаимосвязях.

Изображение выглядит как текст, линия, диаграмма, Шрифт

Автоматически созданное описание

Рисунок 6 – IDEF1X

Проанализируем пару базовых алгоритмов в системе (рисунки 7 и 8):

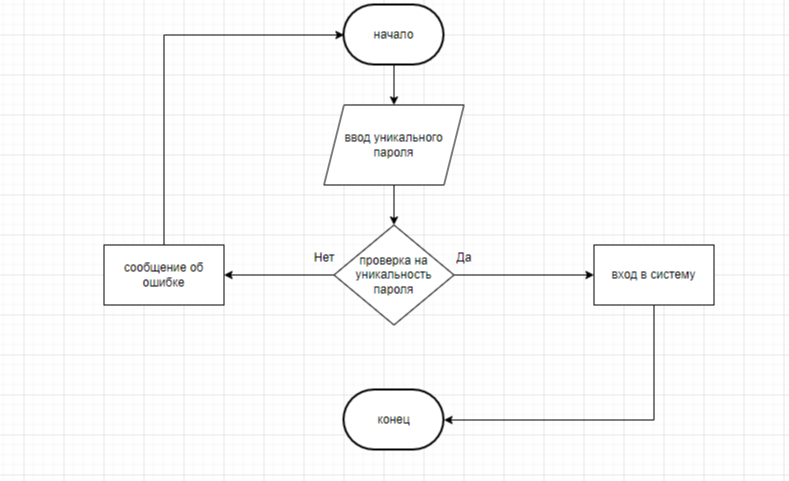


Рисунок 7 – Алгоритм входа администратора и повара.

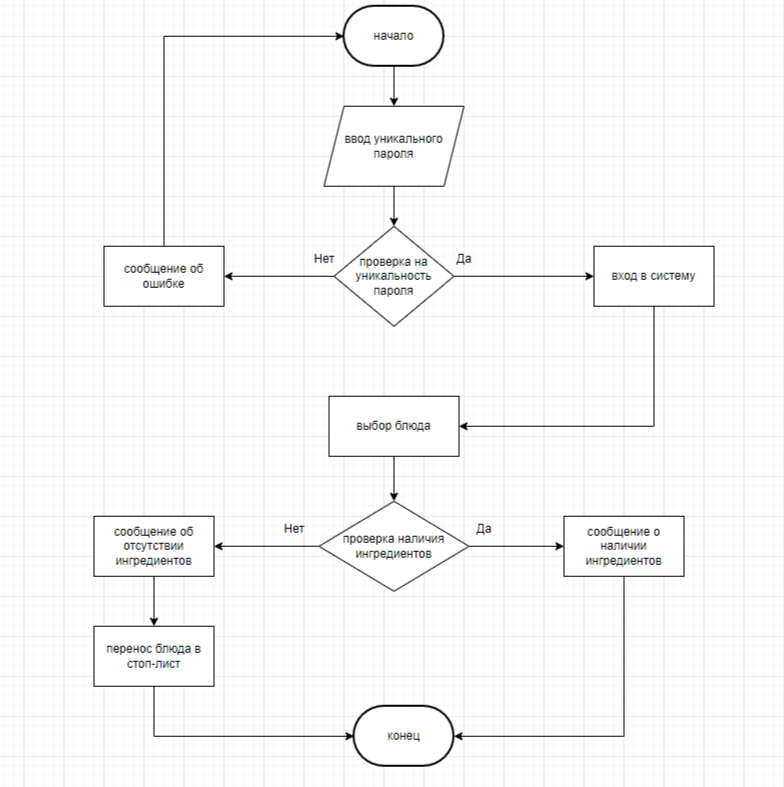


Рисунок 8 – Алгоритм проверки наличия ингредиентов для повара.

## **2.2 Проектирование базы данных и ее структура.**

Проектирование базы данных для системы управления ресторанным меню является ключевым этапом в разработке эффективного и функционального программного обеспечения. Основная цель данной базы данных – обеспечить хранение, управление и обработку информации о меню, блюдах, ингредиентах и ценах, что позволит ресторану оптимизировать свои операции и улучшить взаимодействие с клиентами.

Перед началом Проектирования базы данных необходимо определить основные требования к системе. К ним могут относиться:

* хранение информации о блюдах;
* управление ингредиентами;
* возможность обновления информации о меню в реальном времени;
* генерация отчетов по ингредиентам.

Для реализации базы данных рассмотрим следующие таблицы, их структуру.

Таблица «Menu» хранит всю информацию о меню. В ней есть такие атрибуты: menu\_id, menu\_name, creation\_date.

Таблица «Dish» хранит всю информацию о меню. В ней есть такие атрибуты: dish\_id, menu\_id, dish\_name, price. А атрибут menu\_id является вторичным ключом и помогает соединять таблицу с таблицей «Menu».

Таблица «Ingredient» хранит всю информацию о меню. В ней есть такие атрибуты: ingredient\_id, ingredient\_name, quantity.

Таблица «DishIngredient» хранит всю информацию о меню. В ней есть такие атрибуты: dish\_id, ingredient\_id. А атрибуты dish\_id и ingredient\_id являются вторичными ключами и помогают соединять таблицы «Dish» и «Ingredient».

Код создания базы данных и sql-запросы представлены в Приложении 1.

## **2.3 Реализация интерфейса приложения**

Интерфейс пользователя является одним из важнейших компонентов любого программного продукта, так как он определяет степень удобства взаимодействия пользователя с системой. Для успешного выполнения задач, поставленных перед приложением, интерфейс должен быть интуитивно понятным, функциональным и эстетически привлекательным. В данном разделе будет рассмотрено, как был спроектирован и реализован интерфейс нашего приложения.

Для построения кода импортируем данные библиотеки (рисунок 9):

Изображение выглядит как текст, Шрифт, линия, снимок экрана

Автоматически созданное описание

Рисунок 9 – Библиотеки

Всего есть 4 основных интерфейса приложения: главное окно, окно администратора, окно повара и окно клиента.

Интерфейс главного окна программы (MainApp) организован как стартовый экран, позволяющий пользователю выбрать свою роль для входа в систему. Этот интерфейс предоставляет три основных варианта: «Администратор», «Повар» и «Клиент», и каждая кнопка открывает соответствующее окно входа для роли. Эти кнопки одинакового размера, оформлены в современном стиле с удобным шрифтом и снабжены описательными надписями. Кнопка «Администратор» открывает окно авторизации администратора, где необходимо ввести пароль для подтверждения прав доступа. Кнопка «Повар» аналогично ведет к окну авторизации повара. Кнопка «Клиент», в отличие от остальных, сразу открывает интерфейс клиента, так как для него пароль не требуется.

Интерфейс повара (ChefWindow) предназначен для работы с блюдами, ингредиентами и стоп-листом, обеспечивая удобство и функциональность при выполнении задач повара. Окно повара открывается в увеличенном размере (режим "zoomed") и оформлено в светлых тонах, что создает спокойную рабочую атмосферу. Основное пространство окна разделено на две ключевые области: панель меню (слева) и панель стоп-листа (справа).

Интерфейс администратора (AdminWindow) предназначен для управления меню ресторана, стоп-листом блюд и получения отчетов об ингредиентах. Основное окно интерфейса открывается в увеличенном формате (режим "zoomed") и оформлено в светло-серой цветовой гамме. Окно разделено на три основные области: панель меню, панель стоп-листа и управляющие элементы.

Интерфейс клиента (ClientWindow) предназначен для просмотра меню ресторана в удобном и минималистичном формате. Он автоматически загружает данные из базы при открытии и обновляет отображение при изменении меню в базе данных. Основное окно интерфейса имеет светло-серую цветовую гамму с акцентами белого цвета, что делает визуальное восприятие легким и приятным. Окно открывается в размере 673x1200 пикселей и включает центрально расположенную рамку для отображения списка блюд.

Синтаксис интерфейсов представлен в Приложении 2.

# **Отладка и тестирование модуля**

## **3.1 Перехват исключительных ситуаций**

В разделе рассматривается перехват исключений на примере четырех представленных кодов для интерфейсов администратора и клиента, взаимодействующих с базой данных. Рассмотрены случаи, когда ошибки могут возникать в разных частях программы, и предложены способы их перехвата и обработки.

Ошибка неверного пароля возникает, когда пользователь вводит неправильные учетные данные при попытке авторизоваться в системе. Для предотвращения аварийного завершения программы следует перехватывать эту ошибку и предоставить пользователю соответствующее сообщение (рисунок 10).

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Рисунок 10 – Ошибка неверного пароля.

Ошибка при проверке ингредиентов может возникнуть, если в базе данных нет необходимого ингредиента или если ингредиент имеет неправильные значения (например, отрицательное количество). Программа должна перехватывать такие ошибки и показывать пользователю сообщение о наличии проблемы (рисунок 11).

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 11 – Ошибка при проверке ингредиентов

Ошибка неверного значения цены возникает, когда пользователь пытается ввести некорректную цену блюда, например, нечисловое значение. Чтобы избежать ошибок при сохранении данных, программа должна проверить корректность введенной цены (рисунок 12).

Изображение выглядит как текст, снимок экрана, Шрифт, документ

Автоматически созданное описание

Рисунок 12 – Ошибка неверного значения и сохранения данных

## **Технология работы с приложением**

Технология работы с данным приложением строится на клиент-серверной архитектуре, что позволяет эффективно разделять обработку данных и взаимодействие с пользователем. Важнейшими аспектами являются простота и удобство интерфейса, а также надежность работы с данными, обеспечиваемая использованием проверенных технологий и методов.

Приложение включает в себя несколько основных компонентов: пользовательский интерфейс, базу данных. В основе работы приложения лежит взаимодействие с базой данных для извлечения или сохранения информации. Пользовательский интерфейс представляет собой набор экранов и форм для ввода и отображения данных, обеспечивающих взаимодействие с пользователем.

Основные этапы работы с приложением

1. запуск приложения: после того как пользователь запускает приложение, оно загружает начальный экран, где предоставляется доступ к выбору роли. Следующий же экран может быть авторизации, если приложение требует пароль для входа, или главная страница с меню;
2. авторизация: пользователь вводит пароль (предоставляемый ему при трудоустройстве) в соответствующие поле на экране авторизации. Если данные верны, пользователь получает доступ к основным функциям приложения. В случае ошибки выводится сообщение о неправильном вводе.
3. навигация по интерфейсу: интерфейс интуитивно понятен, и переходы между экранами происходят с помощью кнопок или меню;
4. работа с данными: одной из ключевых функций приложения является работа с данными. Пользователю предоставляется возможность вносить новые данные, редактировать существующие. Эти операции отправляются на сервер, где они обрабатываются и сохраняются в базе данных;
5. обработка ошибок: в процессе работы с приложением могут возникать ошибки, такие как неверно введенные данные. В таком случае система выводит сообщения об ошибках, информируя пользователя о возможных причинах и путях решения проблемы;
6. выход из приложения: когда пользователь завершает работу с приложением, он может выйти из своей учетной записи или просто закрыть приложение. В случае выхода пользователь снова перенаправляется на экран авторизации.

# **ЗАКЛЮЧЕНИЕ**

В ходе выполнения курсового проекта была разработана система управления ресторанным меню, которая представляет собой программный модуль, предназначенный для автоматизации процессов редактирования и удаления позиций в меню, а также управления информацией о блюдах в ресторане. Проект был реализован с использованием современных технологий, что позволяет достичь высокой функциональности и удобства работы как для сотрудников ресторана, так и для администраторов системы.

Целью разработки было создание интуитивно понятного и удобного интерфейса для пользователей. В процессе работы был реализован функционал для внесения изменений в меню в режиме реального времени, что позволяет оперативно обновлять информацию о наличии блюд, их стоимости.

Система управления ресторанным меню предоставляет администратору возможность управления не только редактированием цен блюд, но и управление стоп-листом.

Также разработаны механизмы для предотвращения ошибок и удобной обработки ситуаций с некорректными данными.

Проект демонстрирует высокий уровень автоматизации и упрощения процессов в ресторанном бизнесе, позволяя существенно сократить время на обновление меню и снизить вероятность ошибок при работе с данными. В дальнейшем систему можно дополнительно развивать, добавляя новые функции, такие как интеграция с системой учета заказов, управление складскими запасами.

Таким образом, разработанный программный модуль "Система управления ресторанным меню" решает поставленные задачи и может быть успешно внедрен в реальных условиях ресторанного бизнеса, обеспечив более эффективное управление меню и улучшение качества обслуживания клиентов.

# **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

Монография

1. GitHub: <https://github.com/Ekaterina120/menu.git>

Нормативно-правовые источники:

1. ГОСТ 7.32-2017 Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе.
2. ГОСТ 7.1—2003 Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание. Общие требования и правила составления
3. ГОСТ 7.9—95 Система стандартов по информации, библиотечному и издательскому делу. Реферат и аннотация. Общие требования
4. ГОСТ 7.11—2004 (ИСО 832:1994) Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Сокращение слов и словосочетаний на иностранных европейских языках

Интернет-источники

1. Официальная документация SQLite: <https://sqlite.org/docs.html>
2. Официальная документация Python для Tkinter: <https://docs.python.org/3/library/tkinter.html>
3. DB Browser for SQLLite: <https://sqlitebrowser.org/>
4. Импорт стандартных модулей. Команды import и from. // <https://proproprogs.ru/python_base/python3-import-standartnyh-moduley-komandy-import-i-from>
5. Функции: первое знакомство, определение def и их вызов. // <https://proproprogs.ru/python_base/python3-funkcii-pervoe-znakomstvo-opredelenie-def-i-ih-vyzov>

# **ПРИЛОЖЕНИЕ 1**

Код создания базы данных:

CREATE TABLE Menu (

menu\_id INT PRIMARY KEY,

menu\_name VARCHAR(255),

creation\_date DATE

);

CREATE TABLE Dish (

dish\_id INT PRIMARY KEY,

menu\_id INT,

dish\_name VARCHAR(255),

price DECIMAL(10, 2),

FOREIGN KEY (menu\_id) REFERENCES Menu(menu\_id)

ON DELETE CASCADE

);

CREATE TABLE DishIngredient (

dish\_id INT,

ingredient\_id INT,

PRIMARY KEY (dish\_id, ingredient\_id),

FOREIGN KEY (dish\_id) REFERENCES Dish(dish\_id)

ON DELETE CASCADE,

FOREIGN KEY (ingredient\_id) REFERENCES Ingredient(ingredient\_id)

ON DELETE CASCADE

);

CREATE TABLE Ingredient (

ingredient\_id INT PRIMARY KEY,

ingredient\_name VARCHAR(255),

quantity DECIMAL(10, 2)

);

**Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание**

Рисунок 1 – Вывод ингредиентов для конкретного блюда и наличие ингредиентов на складе

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рисунок 2 – Вывод блюд с использованием конкретного ингредиента (например, яйца)

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рисунок 3 – Вывод общего количества ингредиентов

# **ПРИЛОЖЕНИЕ 2**

**Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание**

Рисунок 1 – Синтаксис интерфейса главного окна

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 2 – Синтаксис интерфейса окна повара

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 3 – Синтаксис интерфейса окна администратора

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 4 – Синтаксис интерфейса окна клиента