**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ 2**](#_Toc184482449)

[**1. Описание предметной области 4**](#_Toc184482450)

[**1.1. Введение в предметную область 4**](#_Toc184482451)

[**1.2. Анализ готовых решений 5**](#_Toc184482452)

[**2. Проектирование системы 7**](#_Toc184482453)

[**2.1. Общее назначение системы 7**](#_Toc184482454)

[**2.2. Требования к функциям, выполняемым системой 7**](#_Toc184482455)

[**2.3. План тестирования 11**](#_Toc184482456)

[**3. Реализация проекта системы 12**](#_Toc184482457)

[**3.1. Описание среды разработки 12**](#_Toc184482458)

[**3.1.1. Выбор и описание программных инструментов 12**](#_Toc184482459)

[**3.1.2. Обоснование выбора инструментария по разработке 13**](#_Toc184482460)

[**3.2. Разработка программного модуля 15**](#_Toc184482461)

[**3.2.1. Реализация пользовательского интерфейса программы 15**](#_Toc184482462)

[**3.2.2. Описание кодом функциональных узлов модуля 19**](#_Toc184482463)

[**3.2.3. Результат работы и тестирования 20**](#_Toc184482464)

[**ЗАКЛЮЧЕНИЕ 24**](#_Toc184482465)

[**СПИСОКИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 25**](#_Toc184482466)

[**ПРИЛОЖЕНИЕ 1 27**](#_Toc184482467)

[**ПРИЛОЖЕНИЕ 2 32**](#_Toc184482468)

[**ПРИЛОЖЕНИЕ 3 38**](#_Toc184482469)

# **ВВЕДЕНИЕ**

Система управления заказами в ресторане – это один из важнейших элементов для организации эффективной работы и правильного функционирования для заведений в области ресторанного бизнеса и общественного питания, в том числе ресторанов. Именно она направленна на увеличение показателей скорости оформления заказов, уменьшения возможных ошибок, связанных с человеческим фактором, таким как работа официантов и менеджеров заведения, улучшить взаимодействие между гостевым залом и кухней. Всё вышеперечисленное позволит облегчить процессы создания и предоставления блюд посетителям. В современном мире процессы автоматизации процессов позволяют предоставлять высокое качество обслуживания, повышать эффективность работы, экономии денежных ресурсов из бюджета заведения, которые можно эффективно направить на улучшение отдельных областей или показателей ресторана.

Цель курсового проекта заключается в разработке программного модуля системы управления заказами в ресторане, который способствует автоматизации учета, хранения и обработки заказов в заведении, позволяя усовершенствовать все вышеперечисленные пункты, а также сократить время ожидания клиентов. Создание данного модуля предполагает интеграцию с другими системами, такими как Microsoft Word или Excel, что позволит создать единую систему для учета товаров и финансовой отчетности, что обеспечит эффективное управление заведением.

Задачи курсового проекта, направленные на создание системы управления заказами в ресторане вытекают из поставленной цели и заключают в себе следующие пункты:

1. Проанализировать систему управления заказами в ресторане.
2. Проанализировать готовые решения.
3. Подготовить план тестирования.
4. Обосновать выбор инструментов и средств разработки.
5. Описать реализацию технического задания.
6. Выполнить тестирование.

Объектом исследования данного проекта является процесс управления заказами в ресторане, рассчитанный на оптимизацию и автоматизацию указанных процессов, а также повышение эффективности обслуживающего персонала.

Предметом исследования выступает прием, хранение и обработка заказов клиентов.

Актуальность работы вытекает из необходимости внедрения современных информационных технологий в заведениях общественного питания, ориентированных на ресторанный бизнес. Благодаря автоматизации, время ожидания заказа сокращается, а точность исполнения увеличивается. Это достигается за счёт электронных систем заказов и расчётов, которые минимизируют человеческие ошибки и ускоряют процесс обработки заказов. Клиенты оценят скорость и точность, а рестораторы – повышение уровня удовлетворённости гостей и, как следствие, их лояльности [11].

Таким образом, в условиях растущей конкуренции важно не только привлекать новых клиентов, но и удерживать существующих, что становится возможным благодаря эффективным системам управления. Поэтому разработка программного модуля «Система управления заказами в ресторане» представляет собой значимый шаг к оптимизации работы заведений общественного питания, что в конечном итоге способствует повышению их прибыли и конкурентоспособности на рынке.

# **Описание предметной области**

## **Введение в предметную область**

В современном мире информационные технологии внесли огромный вклад в развитие индустрии общественного питания, обеспечив более эффективное и быстрое обслуживание клиентов, а также оптимизацию внутренних процессов. Возрос спрос на создание специализированных программных систем для автоматизации обработки заказов, управления меню, учёта продуктов и взаимодействия с клиентами.

Теоретические основы предметной области включают в себя изучение процессов управления заказами, взаимодействия с клиентами и поставщиками, анализ внутренних процессов ресторана.

Методология информационных систем ориентирована на проектирование и внедрение программных решений для индустрии общественного питания, включая сбор, хранение и анализ данных, проектирование баз данных, разработку ПО с графическим интерфейсом, интеграцию с продуктами Microsoft и тестирование функционала. Связь с теоретическими основами заключается в учёте внутренних процессов и потребностей пользователей.

Система управления заказами в ресторанах эволюционировала от устных взаимодействий и записей вручную до современных технологий. С развитием бизнеса появились бумажные меню и записи, затем автоматизированные POS-системы, ускорившие обслуживание и упростившие учет. Современные онлайн-системы и мобильные приложения расширили клиентскую аудиторию, увеличили продажи и улучшили взаимодействие, обеспечивая интеграцию с доставкой, аналитикой и инструментами для оценки эффективности

Создание, внедрение и эксплуатация информационного продукта регулируются положениями настоящего Федерального закона, ЕСПД и Конституции Российской Федерации [1]-[2]. Основными нормативными актами, регулирующими данную сферу, являются: Федеральный закон "Об информации, информационных технологиях и о защите информации" № 149-ФЗ, Федеральный закон "О персональных данных" № 152-ФЗ, а также Федеральный закон "О безопасности критической информационной инфраструктуры Российской Федерации" № 187-ФЗ [3], [4], [6]. Пожалуй, ещё один немаловажный нормативный документ – это ГОСТ Р ИСО/МЭК 12207-2010 [5], который регулирует процессы жизненного цикла программных средств. Современные требования к информационным системам управления заказами в ресторанах включают в себя:

1. Безопасность
2. Автоматизация
3. Масштабируемость
4. Производительность
5. Интеграция с другими системами
6. Гибкость и адаптивность
7. Возможность аналитики и отчётности

Для разработки программного модуля выбрана каскадная модель «Waterfall», так как она подразумевает последовательное завершение стадий, что подходит для небольших проектов с чётко определёнными требованиями. К преимуществам модели относятся простота управления, предсказуемость, быстрота разработки и заранее определённые стоимость и сроки.

## **Анализ готовых решений**

На сегодняшний день системы управления заказами в ресторанах применяются в различных заведениях общественного питания, таких как рестораны, бары, кафе и другие. Анализ готовых решений включает автоматизацию работы, повышение эффективности и управление заказами и бизнес-процессами. Далее приведены примеры таких систем.

YUMA — облачная система управления ресторанами, разработанная для автоматизации процессов в заведениях общественного питания. Она подходит как для небольших кафе, так и для крупных ресторанов, улучшая управление и качество обслуживания.

Функции системы:

1. Приложение для официантов: упрощение принятия заказов.
2. Приложение для доставки: управление доставкой заказов.
3. Приложение для кухни: оптимизация работы кухни.

Таким образом, YUMA представляет собой полноценную экосистему автоматизации ресторанного бизнеса, способствующую повышению эффективности и качеству обслуживания.

Saby (СБИС) Presto – единая система автоматизации для всех бизнес-процессов ресторанного бизнеса. Подходит для различных предприятий, делая её универсальным инструментом по управлению заведением. Имеет широкий функционал:

1. Интеграция с бухгалтерией
2. Приложение для доставки
3. Складской учёт
4. Приложение для гостей

Restik – это сервис для небольших заведений общепита: кафе, кофейни, бары, фастфуды, пекарни, рестораны. Restik используют для приема заказов, учета продаж и остатков на складе, организации программ лояльности, отчетности. Сервис предлагает готовый шаблон сайта доставки еды и QR-меню. Пользователи отмечают простоту и удобство программы [15].

Имеет следующий функционал:

1. POS-система
2. Бронь столов в гостевом зале
3. Управление кухней
4. Складской учёт и аналитика

Таким образом анализ готовых решений на рынке выявил основные функции для разработки модуля управления заказами в ресторане, а также схожие черты этих систем. Конкуренция среди компаний сводится к цене и интуитивно понятному интерфейсу, так как заведения ориентированы на простоту и скорость внедрения продукта.

# **Проектирование системы**

## **Общее назначение системы**

В современных условиях скорость и качество бизнес-процессов влияют на организацию работы коллектива и прибыль. Система управления заказами в ресторане автоматизирует учет, хранение и обработку заказов, улучшая качество обслуживания и контроль процессов. В ресторане с этой системой гости бронируют столик, выбирают блюда, а официант вносит заказ. Администратор управляет меню и качеством обслуживания, системный администратор отвечает за базу данных и отчеты. Все участники взаимодействуют с системой для достижения общих целей.

Предполагаемые результаты от внедрения системы. В результате внедрения система повысит доходность заведения и скорость обслуживания, ускорит передачу заказов на кухню и уменьшит время ожидания для клиентов. Автоматизация снизит влияние человеческого фактора, уменьшив количество ошибок. Система позволит проводить аналитику данных и эффективно распределять ресурсы ресторана.

Эффект от внедрения. Система управления заказами в ресторане в совокупности позволит улучшить все показатели по всем параметрам начиная от увеличения финансового дохода заведения, популярности в общественности за счёт своего подхода к клиентам и сотрудникам ресторана, до стабильности в долгосрочной перспективе.

## **Требования к функциям, выполняемым системой**

Исходя из анализа предметной области программного модуля «Система управления заказами в ресторане» должен выполнять ряд ключевых функциональных задач, обеспечивающих эффективное и бесперебойное управление процессом приёма и обработки заказов. Основные функциональные требования:

1. Управление заказами – позволит отслеживать принятие заказов от клиентов, обработку заказов на кухне, заказов.
2. Управление клиентами – регистрация и аутентификация клиентов, хранение информации о клиентах.
3. Управление столами и резервациями – Бронирование столов.
4. Отчетность и аналитика – генерация отчетов о продажах, популярности блюд, анализ данных для оптимизации работы ресторана.
5. Управление сотрудниками – регистрация и управление данными о официантах и администраторах, назначение ролей и прав доступа в системе.
6. Контроль работ – позволит администратору проводить мониторинг обслуживания клиентов и координировать работой сотрудников
7. Безопасность – данные клиентов и заказы должны быть защищены от несанкционированного доступа
8. Удобство системы отчёта – позволит быстро и легко анализировать показатели эффективности работы сотрудников и популярность блюд.
9. Приятный и интуитивно понятный интерфейс – позволит пользователям легко разобраться с меню ресторана.

Исходя из организационная структуры системы управления заказами в ресторане можно выделить несколько ключевых актёров, каждый из которых выполняет определенные функции и взаимодействует с системой для достижения общих целей. Для наглядности актёры системы и их роли представлены в диаграмме прецедентности (ПРИЛОЖЕНИЕ 1, Рисунок 1).

1. Регистрация:

* Актёры: Клиент
* Описание: если Клиент не зарегистрирован в приложении, то он должен зарегистрироваться.

1. Авторизация:

* Актёры: Клиент, Официант, Администратор ресторана.
* Описание: Клиент или сотрудник вводит свои данные для входа.

1. Смена пользователя:

* Актёры: Клиент, Официант, Администратор ресторана.
* Описание: после авторизации актёры могут сменить пользователя или выйти в окно авторизации.

1. Выход:

* Актёры: Клиент, Официант, Администратор ресторана
* Описание: Клиент или сотрудник выходит из системы.

1. Управление заказами:

* Актёры: Клиент, Официант, Администратор ресторана.
* Описание: Клиент формирует заказ. Сотрудник может редактировать заказы.

1. Бронирование стола:

* Актёры: Клиент.
* Описание: Клиент может забронировать стол.

1. Создание отчёта:

* Актёры: Администратор ресторана.
* Описание: Актёр формирует отчёт о продажах и блюдах.

1. Управление меню:

* Актёры: Клиент, Официант, Администратор ресторана.
* Описание: Клиент добавляет блюда в корзину, официант просматривает меню, администратор редактирует.

Проанализировав предметную область, можно выделить основные объекты, которые представлены в (ПРИЛОЖЕНИЕ 1, Таблица 1).

Для эффективного функционирования системы управления заказами в ресторане необходимо определить ключевые сущности системы и их атрибуты, которые будут использоваться для хранения и обработки данных. Каждая сущность представлена набором атрибутов, отражающих важные характеристики, такие как уникальные идентификаторы, описание, количество и другие параметры, которые позволяют точно идентифицировать и взаимодействовать с данными в процессе работы системы.[14] (ПРИЛОЖЕНИЕ 1, Таблица2 - Таблица 10).

Связи между таблицами системы управления заказами показаны на «Рисунке 2», который визуализирует структуру базы данных и взаимодействие элементов. (ПРИЛОЖЕНИЕ 1, Рисунок 2).

Из «Рисунка 2» хорошо видны следующие связи:

1. Связь «Пользователи – Бронирования»

* Описание: Один пользователь может иметь множество резерваций. Каждая резервация связана с одним клиентом. Связь «один-ко-многим».

1. Связь «Столы – Бронирования»

* Описание: Один стол может быть зарезервирован в нескольких резервациях. Каждая резервация относится к одному столу. Связь «один-ко-многим».

1. Связь «Меню – Элементы меню и ингредиентов»

* Описание: Одно блюдо из меню может содержать множество ингредиентов. Один ингредиент относится к одному блюду. Связь «один-ко-многим».

1. Связь «Элементы меню и ингредиентов – Ингредиенты»

* Описание: один ингредиент может использоваться в нескольких блюдах меню. Связь «один-ко-многим».

1. Связь «Пользователь – Заказы»

* Описание: Один клиент может сделать множество заказов. Каждый заказ связан с одним клиентом. Связь «один-ко-многим».

1. Связь «Сотрудники – Заказы»

* Описание: Один официант может обрабатывать множество заказов. Связь «один-ко-многим».

1. Связь «Заказы – Элементы заказа»

* Описание: Один заказ может содержать множество элементов заказа. Каждый элемент заказа связан с одним заказом. Связь «один-ко-многим».

Для наглядного представления структуры базы данных и взаимодействия сущностей системы управления заказами создана ER-диаграмма, отображающая ключевые элементы, их атрибуты и связи. (ПРИЛОЖЕНИЕ 1, Рисунок 3)

## **План тестирования**

Тестирование программного модуля системы управления заказами в ресторане направлено на проверку функциональности, производительности, стабильности и удобства приложения. Цель тестирования — убедиться в корректности работы всех элементов системы, а также в соответствии функциональных и нефункциональных требований.

Процесс тестирования включает как позитивные, так и негативные тесты с использованием метода «чёрного ящика», при котором имитируются действия пользователя через интерфейс системы. Позитивное тестирование проверяет правильность выполнения функций, а негативное направлено на выявление ошибок при некорректных действиях пользователя. Это позволит выявить потенциальные сбои и улучшить стабильность системы.

Тестирование охватывает несколько ключевых аспектов: проверку функциональности, производительности, безопасности и юзабилити. Важным элементом является оценка корректности работы всех функций системы, включая проверку прав доступа пользователей, обеспечение безопасности данных и адаптивности интерфейса под различные устройства. Также особое внимание уделяется оценке производительности системы при различных нагрузках, а также проверке её стабильности при длительной эксплуатации.

Объекты тестирования: регистрация клиента, авторизация, бронирование стола, добавление блюда в корзину, просмотр заказа, принятие заказа, изменение статуса, выставление счёта, создание нового блюда и создание отчёта. Результаты тестирования в (ПРИЛОЖЕНИЕ 2, Таблица 11 – План тест).

# **Реализация проекта системы**

## **Описание среды разработки**

Для разработки модуля «Система управления заказами в ресторане» требуется популярная и мощная интегрированная среда, обеспечивающая удобство написания и тестирования кода. Она должна ускорять процесс разработки, иметь встроенные модули и библиотеки, поддерживать адаптацию под нужды программиста, а также включать интерактивный отладчик, инструменты тестирования и интеграцию с системами управления версиями.

### **Выбор и описание программных инструментов**

Для разработки программного модуля системы управления заказами в ресторане были рассмотрены и выбраны следующие программные инструменты:

Для реализации программного модуля рассматривались языки программирования, такие как Python, Java, Golang, C++ и другие. Python – высокоуровневый язык программирования, который был выбран в качестве основного языка разработки, благодаря своей простоте и гибкости. Python поддерживает структурное, объектно-ориентированное программирование и функциональное программирование, что делает его подходящим для реализации различных алгоритмических подходов в рамках проекта.

Для хранения и управления данными рассматривались варианты, такие как MySQL, PostgreSQL, Microsoft SQL Server и другие. Для хранения и управления данными выбрана встраиваемая реляционная база данных SQLite3, интегрированная в стандартную библиотеку Python. SQLite3 позволяет создавать, редактировать и извлекать данные из базы, хранящейся в виде одного локального файла, используя стандартный язык SQL.

Для создания графического интерфейса были рассмотрены различные библиотеки, такие как PyQt, Tkinter и Kivy. Выбор был сделан в пользу PyQt6, потому что она представляет собой набор связующих программных интерфейсов между языком Python и библиотекой Qt, разработанной компанией Digia. PyQt6 предоставляет мощные инструменты для создания полноценных графических пользовательских интерфейсов, обеспечивая разработчикам доступ к функциональным возможностям библиотеки Qt через язык Python.

Для генерации и экспорта отчетов рассматривались различные библиотеки, включая Openpyxl, Pandas и ReportLab. Было решено использовать Openpyxl, так как она обеспечивает удобное создание и редактирование файлов Excel (XLSX/XLSM), что позволяет пользователям экспортировать и анализировать данные вне приложения.

Таким образом, для реализации программного модуля были выбраны: интегрированная среда разработки PyCharm, высокоуровневый язык программирования Python, система управления базами данных SQLite3, Openpyxl для создания отчётов и библиотека для создания графического интерфейса пользователя PyQt6. Эти инструменты обеспечат функционал для разработки модуля управления заказами, сохраняя простоту использования и эффективность.

### **Обоснование выбора инструментария по разработке**

Выбор инструментов для разработки программного модуля был основан на требованиях к функциональности, производительности и интуитивно понятному интерфейсу.

PyCharm – это широко используемая среда разработки, предназначенная специально для языка программирования Python. Она обеспечивает богатый набор инструментов для автоматизации задач, отладки и улучшения производительности программистов.

Преимущества PyCharm заключаются в предоставлении мощных инструментов для управления проектом, такие как встроенные средства отладки, автодополнение кода, подсветка синтаксиса и интеграция с системами контроля версий. PyCharm поддерживает настройку виртуальных окружений и удобное управление зависимостями проекта, что особенно полезно для организации и поддержки кода и библиотек. Эти функции значительно ускоряют процесс разработки и делают его более удобным и структурированным, обеспечивая улучшение производительности и удобство в работе.

Python был выбран как основной язык программирования благодаря его гибкости и популярности. Его читаемый код ускоряет разработку и поддержку, а поддержка различных парадигм программирования и обширные библиотеки упрощают расширение функциональности. [7] – [9].

Преимущества Python – обеспечивает высокий уровень абстракции, что делает код более понятным и уменьшает время разработки. Богатая стандартная библиотека и активное сообщество программистов также являются сильными сторонами Python.

SQLite3 выбрана как встроенная реляционная база данных для хранения, легкости интеграции и управления данными на локальном уровне.

Преимуществами SQLite являются легкость интеграции с Python и отсутствие необходимости в серверной части, что позволяет хранить данные локально. Этот инструмент поддерживает стандартные SQL-запросы, которые обеспечивают удобное управление базой данных, а также занимает минимум места и ресурсов, что делает его эффективным решением для реализации задачи курсового проекта.

PyQt6 выбрана для создания графического пользовательского интерфейса благодаря своей эффективной интеграции с Python и богатыми возможностями для разработки интерфейса.

Преимущества PyQt6 – поддерживает кроссплатформенность, что позволяет запускать приложение на различных операционных системах. Также она легко интегрируется с Python-кодом, позволяя реализовать связь между пользовательским интерфейсом и функциональной частью программы. PyQt6 позволяет создавать современный интерфейс, обеспечивающий интуитивное взаимодействие с системой управления заказами.

Openpyxl отрывает возможности для создания отчётов и таблиц в формате Excel прямо из приложения, что обеспечивает пользователю удобство в анализе данных.

Использование данных инструментов позволило создать надежное и многофункциональное приложение с интуитивно понятным интерфейсом.

## **Разработка программного модуля**

В данной части курсового проекта представлен обзор разработки программного модуля «Система управления заказами в ресторане». Процесс включает реализацию функциональных требований из технического задания с использованием различных программных инструментов и технологий. Основное внимание уделяется ключевым аспектам создания продукта: формированию пользовательского интерфейса, кодированию логики работы со структурами данных и взаимодействием с базой данных, а также описанию реализации специализированных функций и модулей системы.

Структура программного модуля основывается на пяти основных окнах, каждое из которых выполняет определенный набор функций и взаимодействует с пользователем.

Акцент будет сделан на практических аспектах реализации решения, включая выбор библиотеки для работы с пользовательским интерфейсом, методы взаимодействия с операционной системой и подходы к обработке данных.

Таким образом, главная цель этого раздела – дать чёткое представление о логике и структуре разработанного программного модуля, подкрепленное примерами кода и скриншотами, демонстрирующими внешний вид и функционал системы.

### **Реализация пользовательского интерфейса программы**

Для обеспечения эффективного взаимодействия пользователя с программным модулем «Система управления заказами в ресторане» был спроектирован и реализован функциональный и интуитивно понятный пользовательский интерфейс. Сама система содержит пять основных окон, которые зависят от роли авторизованного пользователя, будь то администратор, сотрудник-официант или клиент. Примеры графического интерфейса приведены на изображениях от «Рисунок 5» до «Рисунок 21», а их подробное описание можно найти в «Приложении 3».

Спроектированных пользовательский интерфейс в среде PyCharm под управлением языка Python выглядит следующим образом:

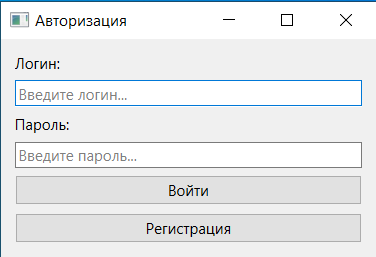


Рисунок 4 – Окно авторизации

1. Окно «Авторизация» содержит два поля «Логин», «Пароль» и две кнопки «Войти» и «Регистрация».

В рабочей области пользователя расположены два поля «Логин», «Пароль» и две кнопки «Войти» и «Регистрация».

1. Окно регистрации:

Ниже заголовка, в центральной части окна находится рабочая область, на ней видны текстовые элементы: «Имя», «Фамилия», «Телефон», «E-mail», «Логин», «Пароль» которые сопровождаются полями для ввода этих самых данных.

Под этими элементами пользовательского интерфейса расположены кнопки действий. Кнопка «Зарегистрироваться» и «Назад». Они предназначены для регистрации клиента в базе данных ресторана и возвращения в окно «Авторизация». (ПРИЛОЖЕНИЕ 3, Рисунок 5).

1. Окно клиента:

Ниже меню и панели инструментов расположена рабочая область, состоящая из текстового поля, в котором выводиться информация о корзине заказа клиента и кнопок «Забронировать столик», «Сделать заказ», «Моя корзина», «Редактировать корзину», «Счёт заказа».

Описание функционала кнопок клиентского окна:

* «Забронировать столик» – Окно с возможностью просмотра свободных столиков на выбранную дату, бронирования свободного стола.
* «Сделать заказ» – Окно с возможностью просмотра и добавления в заказ блюд, выбор категории блюда из меню, проверить текущую корзину и возврат на главное окно клиента.
* «Моя корзина» – При нажатии на кнопку выводится информация о корзине клиента, блюд, которые туда входят и их цены на главное клиентское окно.
* «Редактировать корзину» – Окно с возможностью удаления и сохранение изменений в корзине и возврат на главное окно клиента.
* «Счёт заказа» – В данном окне у клиента есть возможность просмотра информации о своём заказе, и возврат на главное окно клиента.

Подробнее посмотреть на интерфейс клиентского окна и его кнопок можно увидеть в (ПРИЛОЖЕНИЕ 3, Рисунок 6 –Рисунок 11).

1. Окно сотрудника:

Ниже меню и панели инструментов расположена рабочая область, состоящая из текстового поля, в котором выводиться информация о счёте заказа клиента и кнопок «Меню ресторана», «Принять заказ», «Изменить статус заказа», «Выставить счёт».

Описание функционала кнопок окна сотрудника:

* «Меню ресторана» – В данном окне у сотрудника есть возможность просмотра блюд из меню по категориям и возврат на главное окно клиента.
* «Принять заказ» – При нажатии этой копки сотруднику-официанту открывается окно «Принять заказ». В данном окне у сотрудника есть возможность просмотра и выбора заказов, которые может принять и возврат на главное окно клиента.
* «Изменить статус заказа» – В данном окне у сотрудника есть возможность просмотра и выбора заказов, которым можно изменить статус и возврат на главное окно клиента.
* «Выставить счёт» – В данном окне у сотрудника есть возможность просмотра и выбора заказов, которым выставить счёт, информация выводится на главное окно сотрудника и возврат на главное окно клиента.

Подробнее посмотреть на интерфейс окна сотрудника и его кнопок можно увидеть в (ПРИЛОЖЕНИЕ 3, Рисунок 12 –Рисунок 16).

1. Окно администратора:

Ниже меню и панели инструментов расположена рабочая область, состоящая из текстового поля, в котором выводиться информация о счёте заказа клиента и кнопок «Пользователи», «Меню», «Просмотр заказов», «Отчёты и аналитика».

Описание функционала кнопок окна администратора:

* «Пользователи» – Администратор может видеть список клиентов и сотрудников, которые зарегистрированы в системе. Также есть возможность добавлять и удалять их.
* «Меню» – Администратор может просматривать меню ресторана, добавлять, редактировать и удалять блюда.
* «Просмотр заказов» – Администратор может просматривать все заказы и подробную информацию о них.
* «Отчёты и аналитика» – При нажатии на кнопку система предлагает администратору создать отчет в формате .xlsx и выбрать место для его сохранения.

Подробнее посмотреть на интерфейс окна сотрудника и его кнопок можно увидеть в (ПРИЛОЖЕНИЕ 3, Рисунок 17 –Рисунок 21).

### **Описание кодом функциональных узлов модуля**

В первую очередь в программе происходит авторизация пользователя, работающего в системе. Метод авторизации реализован в классе PasswordWindow [13]. Этот метод принимает дынные, введённые пользователем в поля логина и пароля. Затем выполняется проверка пользователя по базе данных. А также метод открытия окна пользователя в зависимости от роли, будь то клиент, официант или администратор. Данные процессы изображены в (ПРИЛОЖЕНИЕ 3, Рисунок 22 – Рисунок 23).

Пользователь вместо авторизации может зарегистрироваться в системе, тем самым создаст новую учётную запись. Регистрация нового клиента осуществляется в классе RegisterWindow. Клиент вводит свои данные и подтверждает регистрацию. Этот процесс описан в (ПРИЛОЖЕНИЕ 3, Рисунок 24).

После авторизации в роли клиента пользователю доступны основные функции: добавления и просмотра блюд заказа, а также бронирование стола, которые реализованы в классе WindowForClient. Работа данных функций представлены в (ПРИЛОЖЕНИЕ 3, Рисунок 25–Рисунок 27).

Если пользователь авторизуется в роли сотрудника, ему доступны следующие функции: принятие заказов, изменение их статуса и выставление счетов. Сотрудник-официант выбирает из списка отрытых заказов и может закрепить его за собой. Этот процесс изображён в (ПРИЛОЖЕНИЕ 3, Рисунок 28).

После принятия заказа официант может изменять его статус, сопровождая клиента на каждом этапе — от принятия заказа до его полного выполнения. Рассмотрим работу функции в (ПРИЛОЖЕНИЕ 3, Рисунок 29).

После выдачи заказа клиенту официант оформляет счет за весь заказ, включая все позиции, указанные в заказе с помощью функции, показанной в (ПРИЛОЖЕНИЕ 3, Рисунок 30).

Если пользователь авторизуется в роли администратора, ему доступны следующие функции: добавление новых блюд в меню ресторана, проведение отчетов и аналитики по блюдам и заказам, которые реализованы в классе WindowForAdmin. При добавлении блюда администратор указывает его название, описание, цену и категорию, после чего сохраняет данные в меню. Этот процесс представлен в (ПРИЛОЖЕНИЕ 3, Рисунок 31).

Администратор может формировать отчеты, включая данные о выручке за день, популярности блюд и среднем чеке. Отчет о выручке за день показывает общую сумму заказов за выбранный период. Отчет о популярных блюдах содержит список самых востребованных позиций меню на основе количества заказов. Отчет о среднем чеке отображает среднюю сумму заказа за указанный период. Этот функционал представлен на «Рисунок 32» и «Рисунок 33». (ПРИЛОЖЕНИЕ 3, Рисунок 32 – Рисунок 33).

Таким образом, были рассмотрены ключевые компоненты модуля, каждый из которых выполняет определенный набор функций. Эти узлы являются основой для работы системы, обеспечивают её функциональность и взаимодействие с пользователем. Детальное описание кода даёт чёткое представление о логике работы модуля и формирует основу для дальнейшей реализации и тестирования разработанной системы. Подробнее о реализации процессов и модулей можно ознакомится на GitHub [10].

### **Результат работы и тестирования**

1. Кейс «Регистрация клиента»

В данном кейсе клиент проходит регистрацию в системе. Клиент должен заполнить все поля необходимые поля, указанные в окне. Затем пользователь должен нажать кнопку «Зарегистрироваться», после чего клиент увидит уведомление об успешной регистрации. Результат показан в (ПРИЛОЖЕНИЕ 3, Рисунок 34).

1. Кейс «Авторизация пользователя»

В данном кейсе пользователю необходимо ввести свой логин и пароль для успешной авторизации в системе. Подготовка присутствует на «Рисунок 35».

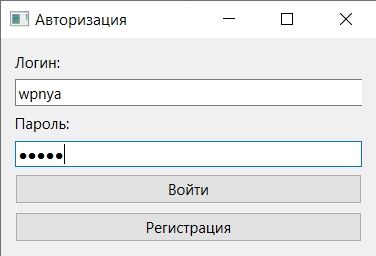


Рисунок 35 – Авторизация пользователя в системе

После нажатия на кнопку «Войти» пользователь авторизуется в системе, это и будет нашим результатом выполнения. После авторизации пользователя встречает уведомление об успешной авторизации, после скрытия уведомления перед нами откроется окно для пользователя в зависимости от его роли.

1. Кейс «Бронирование стола»

В данном кейсе стоит задача бронирования стола, для этого потребуется нажать на кнопку «Забронировать столик» и в появившемся окне клиент при нажатии на кнопку «Показать свободные столики», где может увидеть свободные столы. Клиент нажимает на свободный и после нажимает на кнопку «Забронировать столик». После успешного бронирования стола клиент увидит информационное сообщение об успешной операции. Процесс показан в (ПРИЛОЖЕНИЕ 3, Рисунок 36).

1. Кейс «Добавление блюда в корзину»

В данном кейсе стоит задача добавить блюдо в корзину, для этого потребуется нажать на кнопку «Выбрать категорию», нажать на блюдо, а после нажать на кнопку «Добавить в заказ». После успешного добавления блюда в корзину клиент увидит информационное сообщение об успешной операции. Процесс показан в (ПРИЛОЖЕНИЕ 3, Рисунок 37).

1. Кейс «Просмотр заказа»

В данном кейсе клиент имеет возможность просматривать информацию о текущем заказе. Для этого пользователь нажимает на кнопку «Моя корзина». В главном окне отображаются все блюда, добавленные в заказ, их количество и стоимость. Если заказ пуст, клиент увидит соответствующее сообщение о том, что корзина пуста. Результат отображен в (ПРИЛОЖЕНИЕ 3, Рисунок 38).

1. Кейс «Принятие заказа»

В данном кейсе официант ресторана должен принять заказ от клиента. Для этого нужно выбрать заказ из списка активных заказов и нажать на кнопку «Принять выбранный заказ». Официант увидит уведомление об успешном принятии заказа. Процесс показан в (ПРИЛОЖЕНИЕ 3, Рисунок 39).

1. Кейс «Изменение статуса заказ»

В данном кейсе официант ресторана изменяет статус заказа, например, на «В процессе», «Закрыт» или «Отменён». Для этого необходимо выбрать заказ в списке заказов, затем нажать на кнопку «Изменить статус» и выбрать новый статус из выпадающего списка. После изменения статуса система должна обновить информацию о заказе, и сотрудник увидит уведомление о успешном изменении статуса. Процесс показан в (ПРИЛОЖЕНИЕ 3, Рисунок 40).

1. Кейс «Выставление счёта»

В данном кейсе официант выставляет счёт для заказа из списка, который он увидит при нажатии на кнопку «Выставить счёт». Для этого нужно выбрать активный заказ и нажать кнопку «Показать счёт». После этого система должна отобразить информацию о заказе, включая список блюд и их стоимость, а также общую сумму. Процесс показан в (ПРИЛОЖЕНИЕ 3, Рисунок 41).

1. Кейс «Создание нового блюда»

В данном кейсе администратор создаёт новое блюдо для меню. Для этого необходимо перейти в раздел «Меню», нажать на кнопку «Создать новое блюдо» и заполнить все обязательные поля: название блюда, описание, цена и категория. Процесс показан в (ПРИЛОЖЕНИЕ 3, Рисунок 42).

1. Кейс «Создание отчёта»

В данном кейсе администратор должен создать отчёт по заказам или продажам за день. Для этого нужно нажать на кнопку «Отчёты и аналитика», выбрать место, где будет сохранён файл и дать ему название, затем нажать кнопку «Сохранить». После этого система генерирует отчёт, который можно просмотреть на рабочем столе устройства. Результат отображен в (ПРИЛОЖЕНИЕ 3, Рисунок 43).

Таким образом, в данной главе был представлен процесс разработки модуля «Система управления заказами в ресторане», с акцентом на выбор технологий и инструментов, таких как Python, PyQt6, SQLite3 и Openpyxl. Описание графического интерфейса пользователя, управление заказами, базой данных и функционалом приложения позволяет создать эффективную и удобную систему для различных ролей пользователей. Для более детального ознакомления с проектом и исходным кодом можно обратиться к репозиторию на GitHub [10].

# **ЗАКЛЮЧЕНИЕ**

В ходе курсового проекта была реализована задача по разработке программного модуля системы управления заказами в ресторане.

Были выполнены все поставленные задачи: проведён анализ предметной области, определены ключевые требования к системе, выбраны инструменты для реализации программного продукта, разработан интерфейс пользователя, запрограммирован функционал модуля, а также выполнено тестирование созданной системы.

Разработанный модуль автоматизирует процесс учёта, хранения и обработки заказов, включая управление меню, ингредиентами, резервирование столов и формирование отчётной документации. Это способствует увеличению скорости обслуживания клиентов, повышению эффективности работы сотрудников и снижению временных затрат.

Программный продукт имеет экономическое обоснование, так как его внедрение позволяет оптимизировать ключевые процессы ресторана, увеличить прибыль за счёт повышения производительности и качества обслуживания, а также улучшить управление за счёт возможности дальнейшей интеграции с аналитическими инструментами.

Результаты тестирования подтвердили соответствие модуля требованиям и возможность его интеграции. Выявлена ошибка: при изменении статуса заказа официантом и отмене действия приложение закрывается. Устранение запланировано на этапе доработки.

Система управления заказами требует регулярного обновления для соответствия новым требованиям и стандартам. В будущем планируется расширение функциональности, включая интеграцию с внешними сервисами и аналитическими инструментами.

Разработанный модуль доказал свою ценность для автоматизации ключевых процессов ресторана, повышения производительности, улучшения качества обслуживания и снижения временных затрат.

# **СПИСОКИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

Стандарты и законодательные материалы:

1. ГОСТ 19.106-78. Единая система программной документации (ЕСПД). Требования к программным документам, выполненным печатным способом (с Изменениями №1). – М.: РСТ, 2010. (с Изменением №1, утвержденным в сентябре 1981 г.).
2. Конституция Российской Федерации (принята всенародным голосованием 12.12.1993) (с учётом поправок, внесённых Законами РФ о поправках к Конституции РФ, одобренными в ходе общественного голосования от 01.07.2020).
3. Федеральный закон «Об информации, информационных технологиях и о защите информации» от 27.07.2006 № 149-ФЗ. – М.: Государственная Дума Российской Федерации, 8 июля 2006 года.
4. Федеральный закон «О персональных данных» от 27.07.2006 № 152-ФЗ. – М.: Государственная Дума Российской Федерации, 8 июля 2006 года.
5. ГОСТ Р ИСО/МЭК 12207-2010. Национальный стандарт Российской Федерации. Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств» (утв. и введен в действие Приказом Росстандарта от 30.11.2010 N 631-ст) – М.: Стандартинформ, 2011.
6. Федеральный закон «О безопасности критической информационной структуры Российской Федерации» от 26.07.2017 № 187-ФЗ. – М.: Государственная Дума Российской Федерации, 12 июля 2017 года.

Учебники и учебные пособия:

1. Изучаем Python, том 1 5-е издание: Учебное пособие / Марк Лутц – Пер. с англ. – СПб.: ООО «Диалектика», 2017.
2. Изучаем Python, том 2 5-е издание: Учебное пособие / Марк Лутц – Пер. с англ. – СПб.: ООО «Диалектика», 2018.
3. Python 3 и PyQt 6. Разработка приложений: Учебное пособие / Н. А. Прохоренок, В. А. Дронов. – СПб.: БХВ-Петербург, 2018.

Монографии

1. GitHub: Курсовой проект Климина А. Д. по теме: «Система управления заказами в ресторане» //<https://github.com/Klimin0Andrey/Restoran.git> (28.11.2024)

Интернет источники

1. Козлов Р. А. «Программы управления рестораном и кафе» //<https://a2is.ru/catalog/programmy-dlya-kafe-i-restoranov> (28.11.2024)
2. Павлов М. И. «Алгоритмы и структуры данных» //https://education.vk.company/program/kurs-algoritmy-structury-dannyh
3. Python Software Foundation « Python 3.12.7 documentation» //https://docs.python.org/3/ (28.11.2024)
4. Селектовна И. Д. «Введение в SQL» //https://stepik.org/course/191774/syllabus (28.11.2024)
5. Сужаев О. И. «Как автоматизировать заведение общепита» //https://restik.com/blog/kak-avtomatizirovat-zavedenie-obshchepita/ (28.11.2024)

# **ПРИЛОЖЕНИЕ 1**

Таблица 1 – Объекты предметной области

|  |  |  |
| --- | --- | --- |
| № п/п | Наименование объекта | Описание |
| 1 | Пользователь | Человек имеющий определенные права и роли в системе. |
| 2 | Действие | Система хранения информации о зафиксированном действии сделанным пользователем. |
| 3 | Заказ | Система хранения информации о заказах, сделанных клиентами в ресторане. |

Таблица 2. Атрибут сущности – «Пользователь»(Customer)

| Атрибут | Тип данных | Описание |
| --- | --- | --- |
| Customer\_ID | INT | Уникальный идентификатор пользователя |
| First\_name | VARCHAR(50) | Имя пользователя |
| Last\_name | VARCHAR(50) | Фамилия Пользователя |
| Phone\_num | VARCHAR(20) | Номер телефона |
| Email | VARCHAR(100) | Адрес электронной почты |
| Login | VARCHAR(50) | Уникальное имя пользователя в системе |
| Password | VARCHAR(100) | Пароль пользователя |
| Role | VARCHAR(20) | Роль пользователя |

Таблица 3. Атрибут сущности – «Сотрудники» (Employees)

| Атрибут | Тип данных | Описание |
| --- | --- | --- |
| Employee\_ID | INT | Уникальный идентификатор сотрудника |
| First\_name | VARCHAR(50) | Имя пользователя |
| Last\_name | VARCHAR(50) | Фамилия Пользователя |
| Phone\_num | VARCHAR(20) | Номер телефона |
| Email | VARCHAR(100) | Адрес электронной почты |
| Login | VARCHAR(50) | Уникальное имя сотрудника в системе |
| Password | VARCHAR(100) | Пароль пользователя |
| Role | VARCHAR(20) | Роль сотрудника |

Таблица 4. Атрибут сущности – «Столы» (Tables)

|  |  |  |
| --- | --- | --- |
| Атрибут | Тип данных | Описание |
| Table\_ID | INT | Уникальный идентификатор стола |
| Table\_Number | INT | Номер стола |
| Capacity | INT | Вместимость стола |

Таблица 5. Атрибут сущности – «Бронирования» (Reservations)

| Атрибут | Тип данных | Описание |
| --- | --- | --- |
| Reservation\_ID | INT | Уникальный идентификатор брони |
| Customer\_ID | INT | Уникальный идентификатор пользователя |
| Table\_ID | INT | Уникальный идентификатор стола |
| Date | DATETIME | Дата |
| Amount\_people | INT | Количество людей |

Таблица 6. Атрибут сущности – «Меню» (MenuItems)

| Атрибут | Тип данных | Описание |
| --- | --- | --- |
| MenuItem\_ID | INT | Уникальный идентификатор меню |
| Name | VARCHAR(100) | Название блюда |

Продолжение Таблицы 6

|  |  |  |
| --- | --- | --- |
| Атрибут | Тип данных | Описание |
| Description | TEXT | Описание блюда |
| Price | DECIMAL(10, 2) | Цена |
| Category | VARCHAR(50) | Категория блюда |

Таблица 7. Атрибут сущности – «Ингредиенты» (Ingredients)

|  |  |  |
| --- | --- | --- |
| Атрибут | Тип данных | Описание |
| Ingredient\_ID | INT | Уникальный идентификатор ингредиента |
| Name | VARCHAR(100) | Название продукта |
| Quantity\_in | DECIMAL(10, 2) | Количество ингредиентов |
| Unit | VARCHAR(20) | Единица измерения |

Таблица 8. Атрибут сущности – «Элементы меню и ингредиентов» (MenuItemIngredients)

|  |  |  |
| --- | --- | --- |
| Атрибут | Тип данных | Описание |
| MenuItem\_ID | INT | Уникальный идентификатор меню |
| Ingredient\_ID | INT | Уникальный идентификатор ингредиента |
| Quantity | DECIMAL(10, 2) | Количество |
| Unit | VARCHAR(20) | Единица измерения |

Таблица 9. Атрибут сущности – «Заказы» (Orders)

| Атрибут | Тип данных | Описание |
| --- | --- | --- |
| Order\_ID | INT | Уникальный идентификатор заказа |
| Customer\_ID | INT | Уникальный идентификатор пользователя |
| Employee\_ID | INT | Уникальный идентификатор официанта |
| Order\_Date | DATETIME | Дата заказа |
| Total\_Amount | DECIMAL | Общая сумма |
| Order\_Status | VARCHAR(20) | Статус заказа |

Таблица 10. Атрибут сущности – «Элементы заказа» (OrderItems)

| Атрибут | Тип данных | Описание |
| --- | --- | --- |
| OrderItem\_ID | INT | Уникальный идентификатор элемента заказа |
| Order\_ID | INT | Уникальный идентификатор заказа |
| MenuItem\_ID | INT | Уникальный идентификатор меню |
| Quantity | INT | Количество |
| Price | DECIMAL(10,2) | Цена |

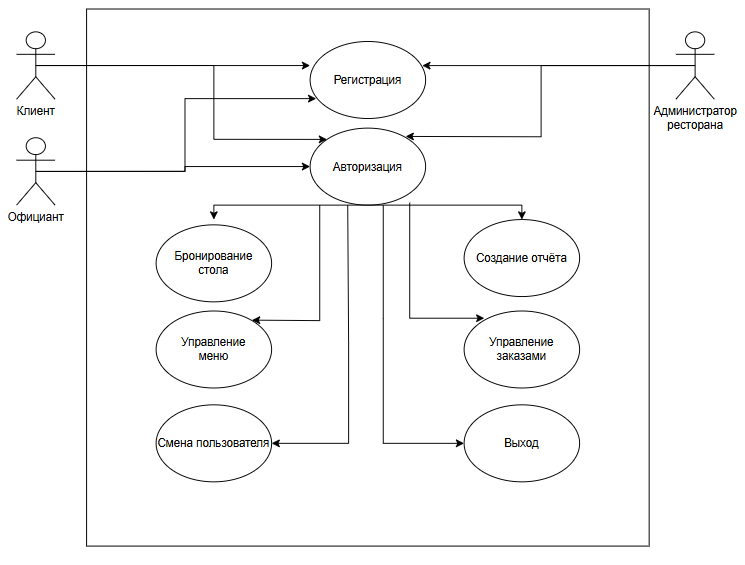


Рисунок 1 – Диаграмма прецедентности

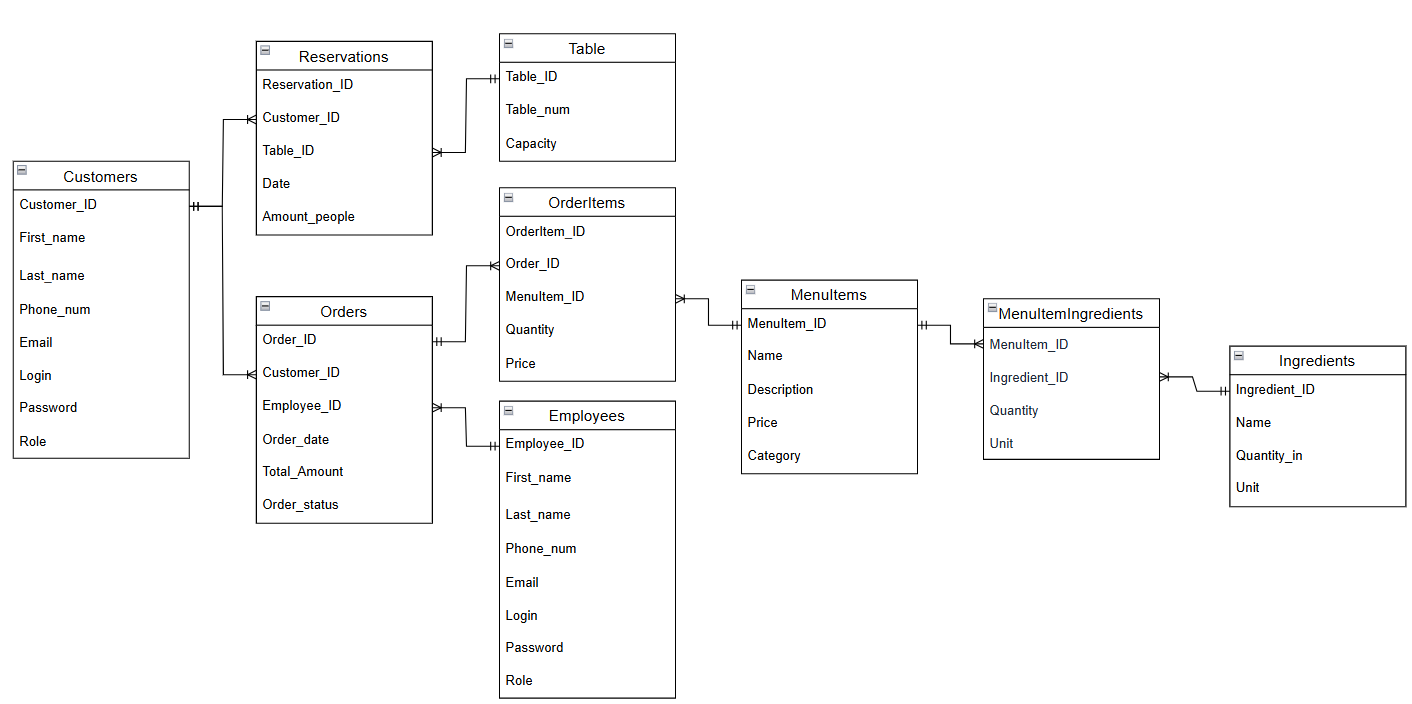


Рисунок 2 – Диаграмма IDEF1X

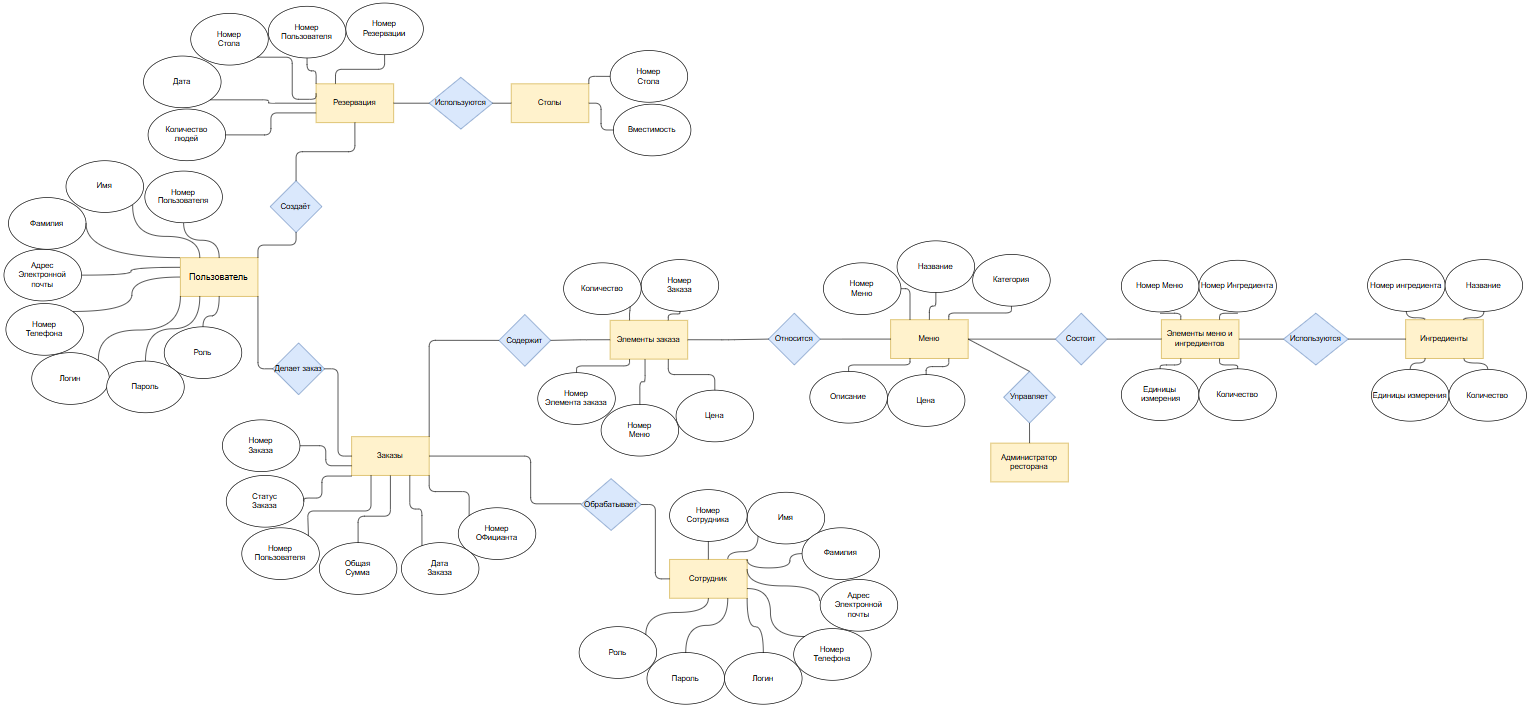


Рисунок 3 – ER диаграмма

# **ПРИЛОЖЕНИЕ 2**

Таблица 11 – План тест

| № | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат |
| --- | --- | --- | --- | --- |
| 1 | Регистрация клиента | Имя | Иван | Успешная регистрация с заполненными полями. |
|  |  | Фамилия | Полетаев |
|  |  | Телефон | +7 (985) 738-11-12 |
|  |  | E-mail | wpnya@mail.com |
|  |  | Логин | wpnya |
|  |  | Пароль | 123456 |
|  |  | Кнопка «Зарегистрироваться» | Нажать на кнопку |
|  |  | Кнопка «Назад» | Нажать на кнопку | Возвращение в окно авторизации |
| Фактический результат | | | | |
| 2 | Авторизация пользователя | Логин | wpnya | Успешный вход в окно пользователя с приветственным сообщением. |
|  |  | Пароль | 123456 |
|  |  | Кнопка «Войти» | Нажать на кнопку |
| Фактический результат | | | | |

Продолжение таблицы 11

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат |
| 3 | Бронирование стола | Кнопка «Забронировать столик» | Нажать на кнопку | Клиент успешно забронировал стол на определенную дату. |
|  |  | Кнопка «Показать свободные столики» | Нажать на кнопку |
|  |  | Кнопка «Забронировать столик» | Нажать на кнопку |
| Фактический результат | | | | |
| 4 | Добавление блюда в корзину | Кнопка «Сделать заказ» | Нажать на кнопку | Клиент успешно добавил блюдо в корзину. |
|  |  | Кнопка «Выбрать категорию» | Нажать на кнопку |
|  |  | Кнопка «Завершить заказ» | Нажать на кнопку |

Продолжение таблицы 11

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат |
| Фактический результат | | | | |
| 5 | Просмотр заказа | Кнопка «Моя корзина» | Нажать на кнопку | Информация о корзине клиента выведена в «Окно клиента» |
| Фактический результат | | | | |
| 6 | Принятие заказ | Кнопка «Принять заказ» | Нажать на кнопку | Официант успешно принял заказ. |
|  |  | Кнопка «Принять выбранный заказ» | Нажать на кнопку |

Продолжение таблицы 11

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат |
| Фактический результат | | | | |
| 7 | Изменение статуса заказа | Кнопка «Изменить статус заказа» | Нажать на кнопку | Официант успешно изменил статус заказа. |
|  |  | Кнопка «Изменить статус» | Нажать на кнопку |
| Фактический результат | | | | |
| 8 | Выставление счёта | Кнопка «Выставить счёт» | Нажать на кнопку | Официант успешно выставил счёт заказа. Информация о счёте выведена в окне сотрудника. |
|  |  | Кнопка «Показать счёт» | Нажать на кнопку |

Продолжение таблицы 11

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Наименование функциональности | | Наименование поля | | Тестовый набор | Ожидаемый результат |
| Фактический результат | | | | | | |
| 9 | | Создание нового блюда | Кнопка «Меню» | Нажать на кнопку | | Новое блюдо успешно создано и добавлено в меню ресторана. |
|  | |  | Кнопка «Добавить блюдо» | Нажать на кнопку | |
|  | |  | Название блюда | Эклер | |
|  | |  | Описание | Заварное тесто с нежным кремом и глазурью. | |
|  | |  | Цена | 220.00 | |
|  | |  | Категория | Десерты | |
|  | |  | Кнопка «Сохранить» | Нажать на кнопку | |
| Фактический результат | | | | | | |

Продолжение таблицы 11

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат |
| 10 | Создание отчёта | Кнопка «Отчёты и аналитика» | Нажать на кнопку | Файл отчёта успешно создан. |
|  |  | Имя файла | Отчёты за декабрь. |
|  |  | Кнопка «Сохранить» | Нажать на кнопку |
| Фактический результат | | | | |

# **ПРИЛОЖЕНИЕ 3**

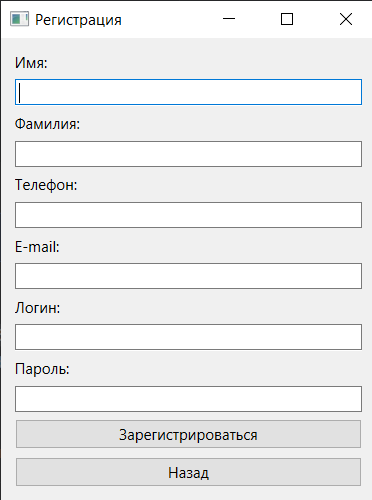


Рисунок 5 – Окно регистрации

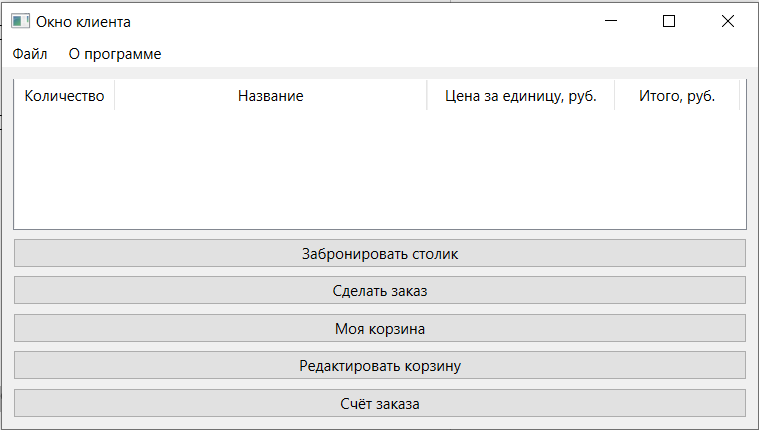


Рисунок 6 – Окно клиента

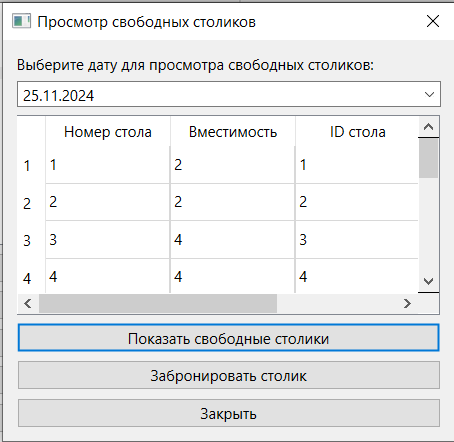


Рисунок 7 – Окно кнопки «Забронировать столик»

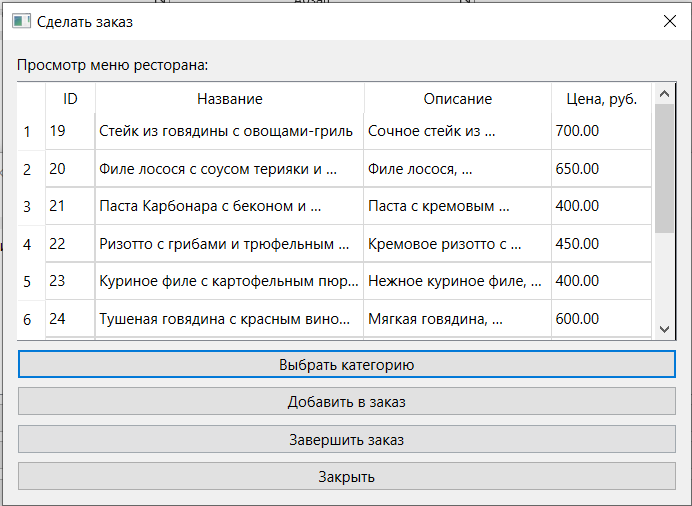


Рисунок 8 – Окно кнопки «Сделать заказ»

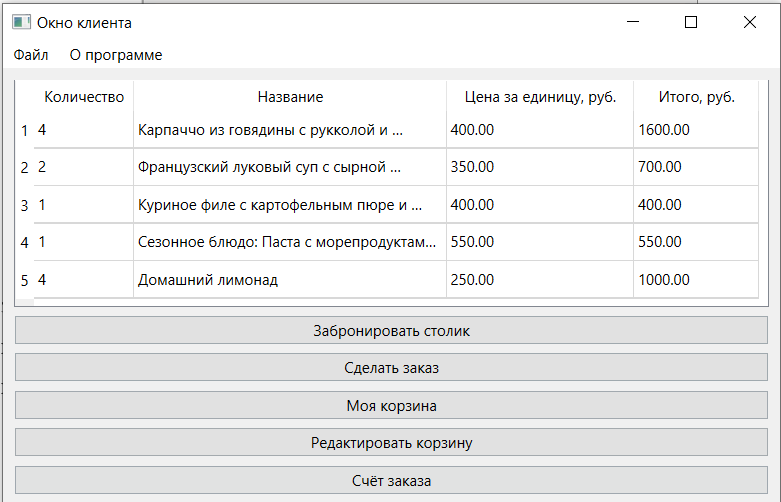


Рисунок 9 – Вывод информации о заказе клиента

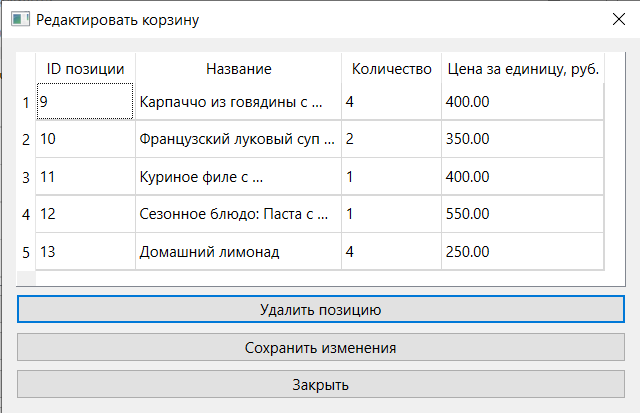


Рисунок 10 – Окно кнопки «Редактировать корзину»

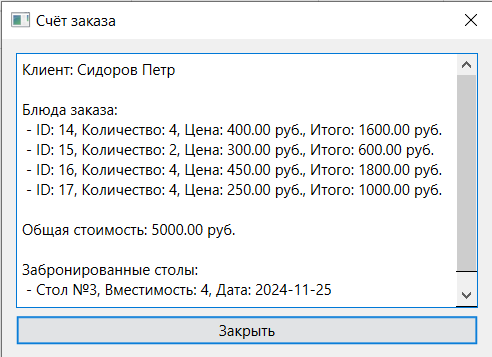


Рисунок 11 – Окно кнопки «Счёт заказа»

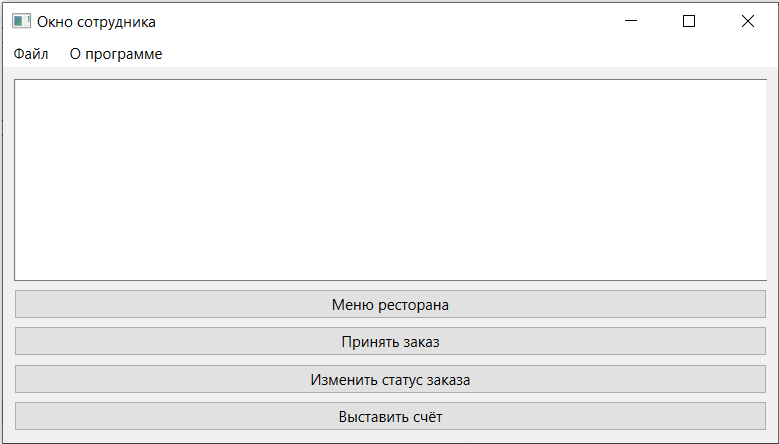


Рисунок 12 – Окно сотрудника

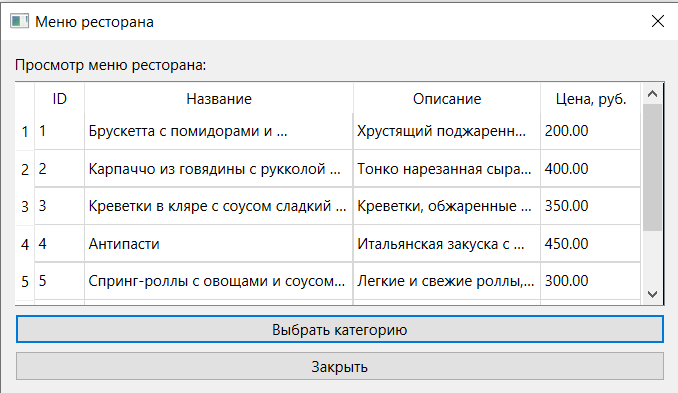


Рисунок 13 – Функционал кнопки «Меню ресторана»

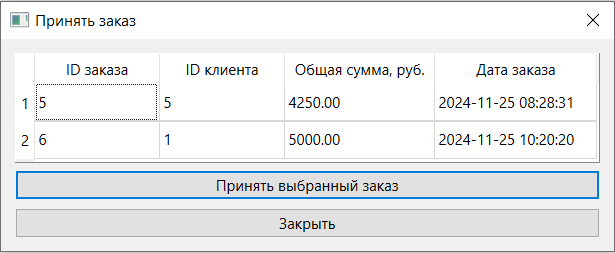


Рисунок 14 – Функционал кнопки «Принять заказ»

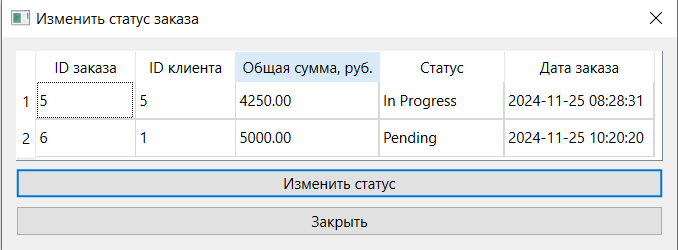


Рисунок 15 – Функционал кнопки «Изменить статус заказа»

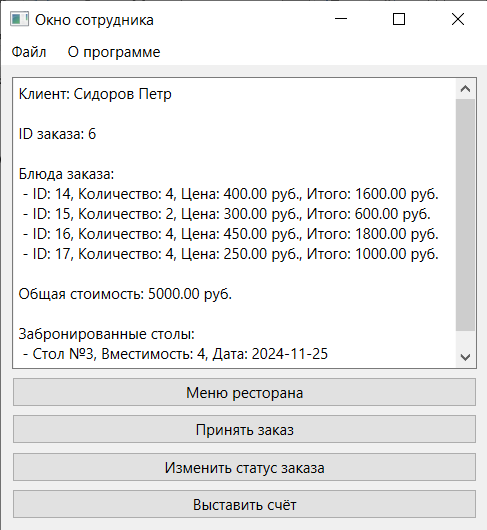


Рисунок 16 – Функционал кнопки «Выставить счёт»

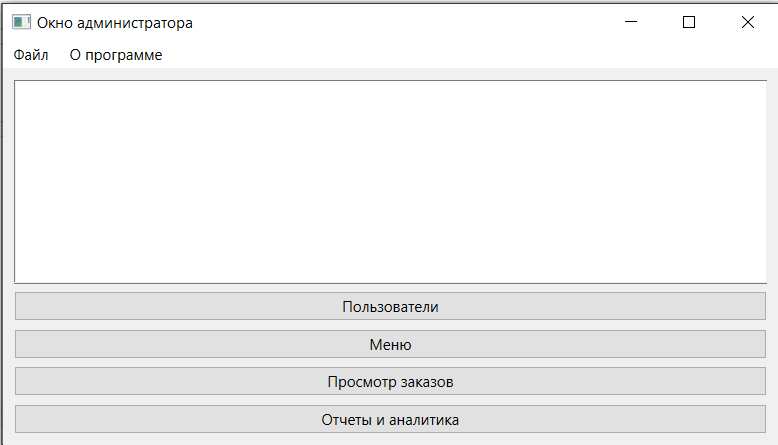


Рисунок 17 – Окно администратора

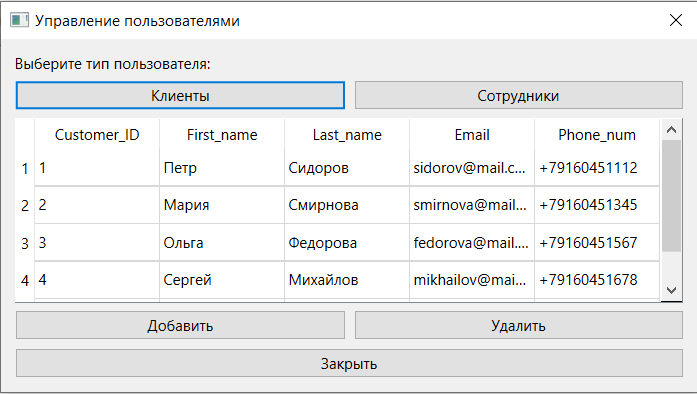


Рисунок 18 – Функционал кнопки «Пользователи»

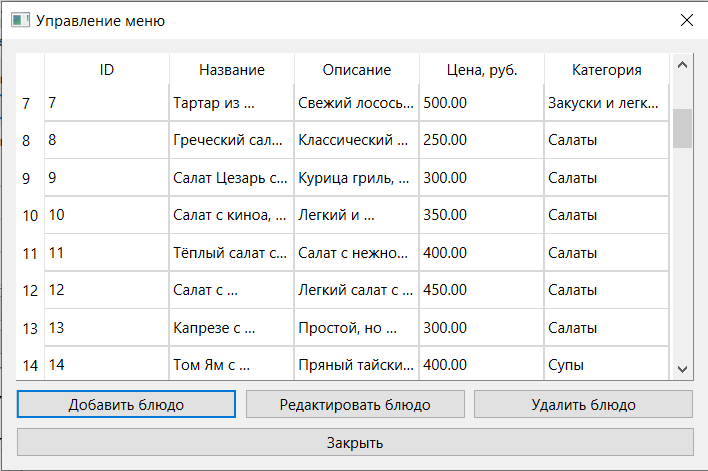


Рисунок 19 – Функционал кнопки «Меню»

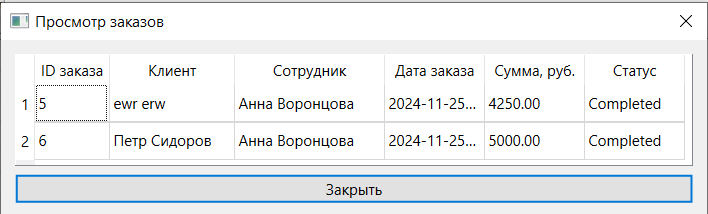


Рисунок 20 – Функционал кнопки «Просмотр заказов»

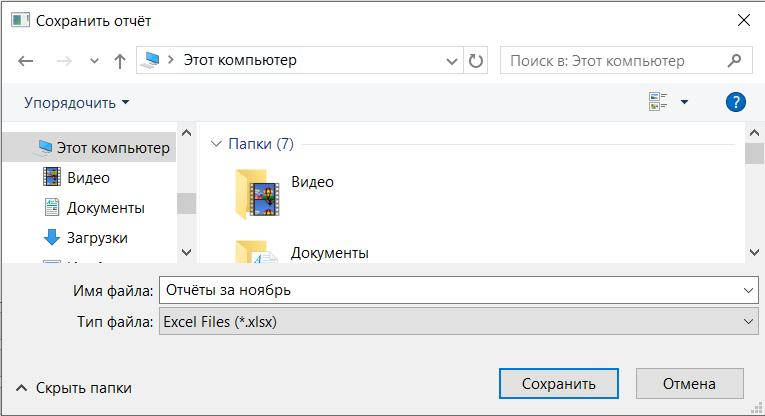


Рисунок 21 – Функционал кнопки «Отчёты и аналитика»

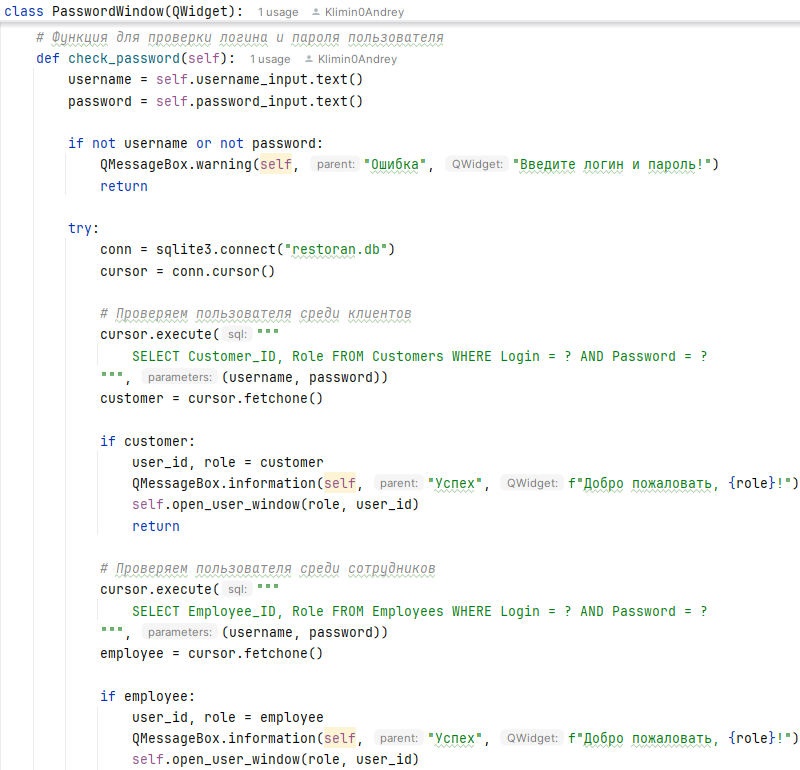


Рисунок 22 – Метод авторизации пользователей

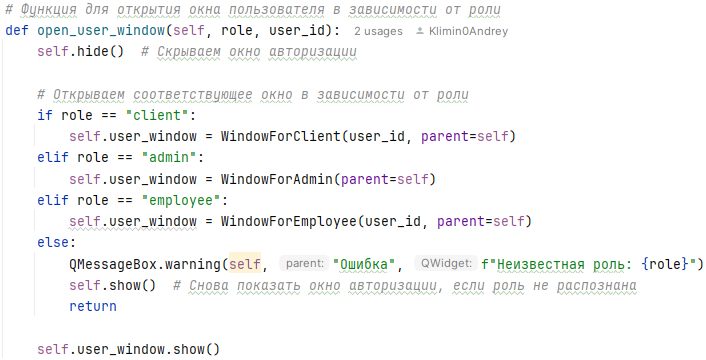


Рисунок 23 – Метод открытия окна пользователя в зависимости от роли

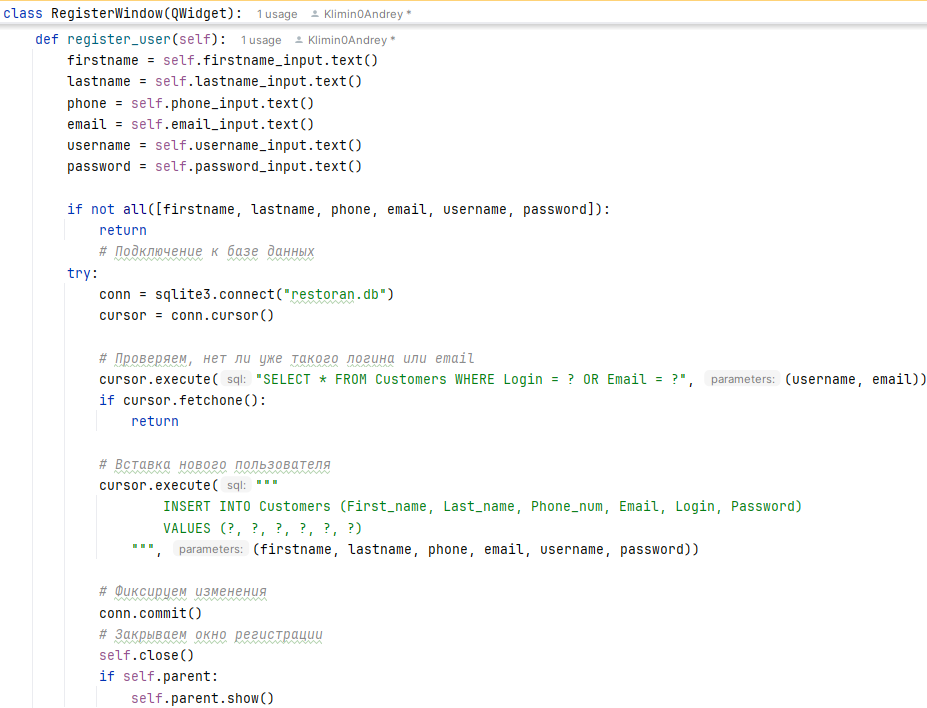


Рисунок 24 – Регистрация нового клиента



Рисунок 25 – Функция добавления блюд в заказ

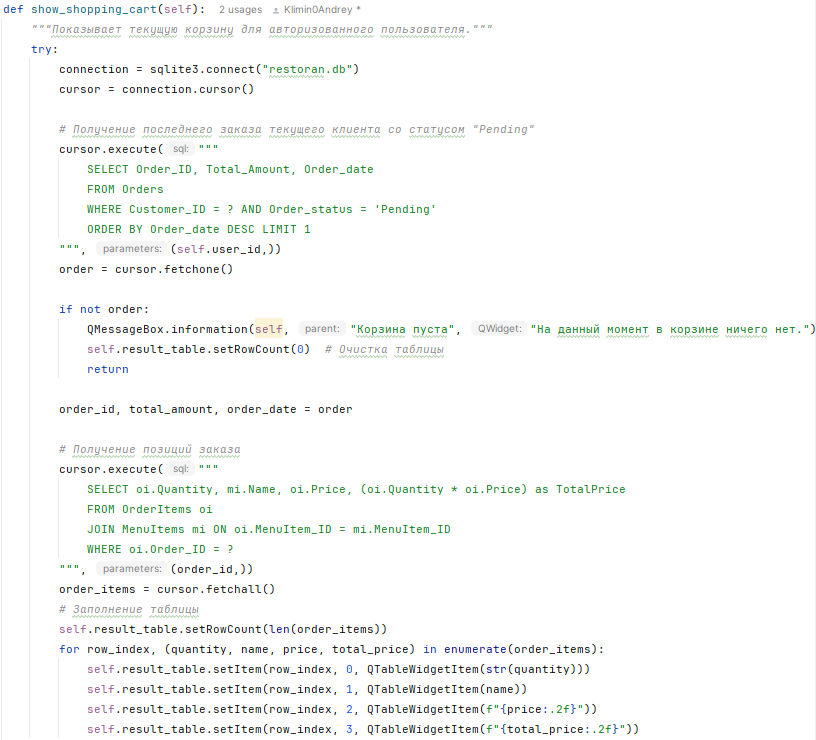


Рисунок 26 – Функция просмотра блюд заказа

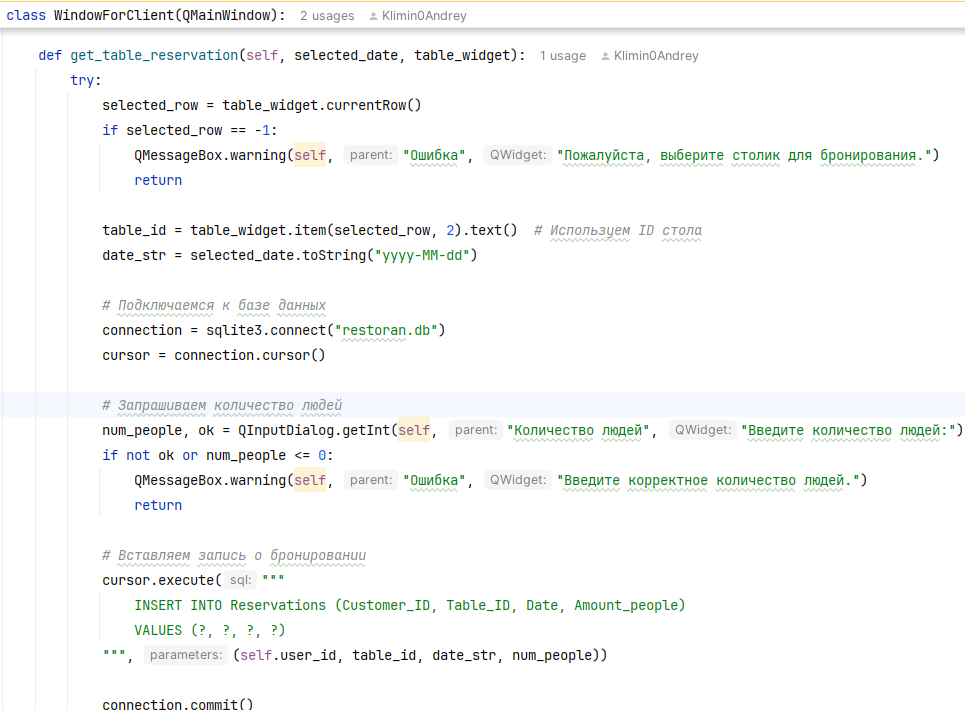


Рисунок 27 – Функция бронирования стола

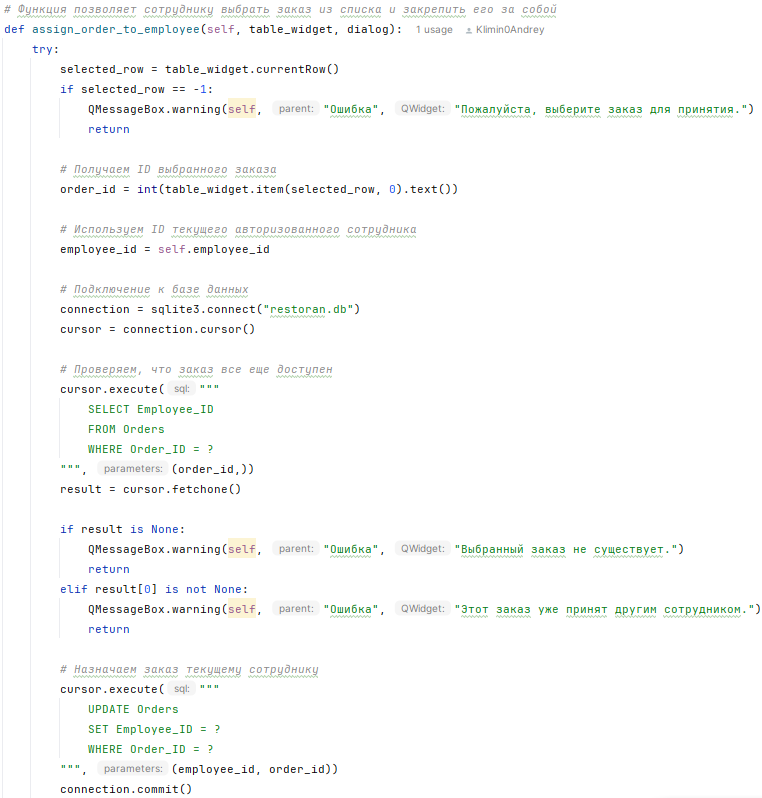


Рисунок 28 – Функция принятия заказа официантом

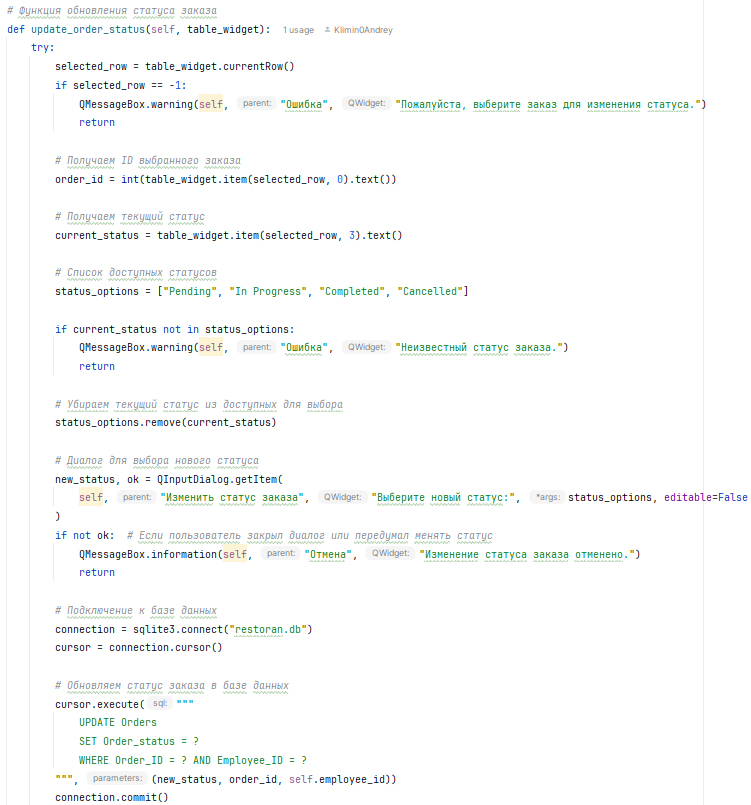


Рисунок 29 – Функция изменения статуса заказа

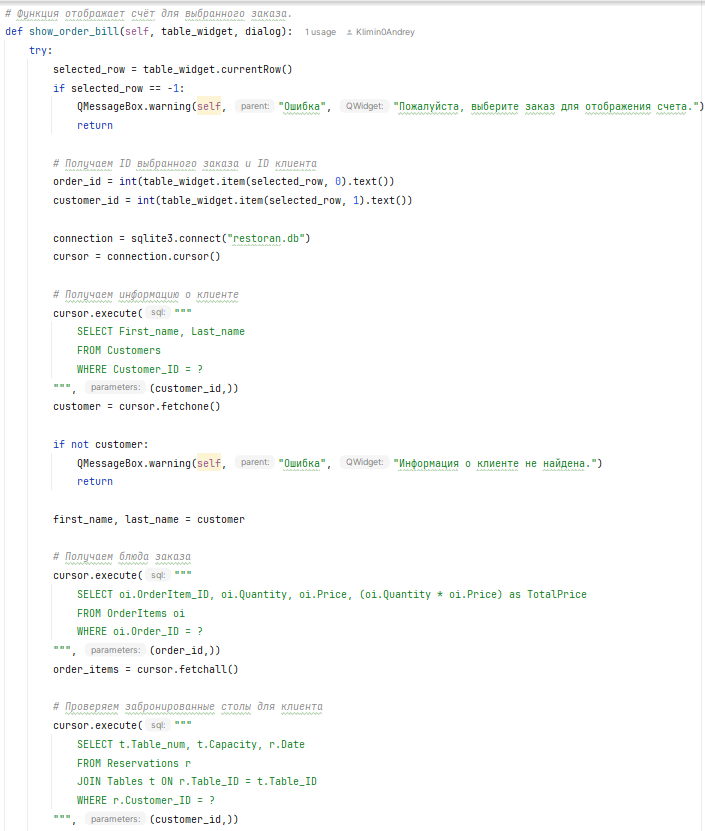


Рисунок 30 – Функция отображения счёта для выбранного заказа



Рисунок 31 – Функция добавления нового блюда в меню



Рисунок 32 – Функция генерации отчётов и аналитики Excel



Рисунок 33 – Продолжение функции отчётов и аналитики

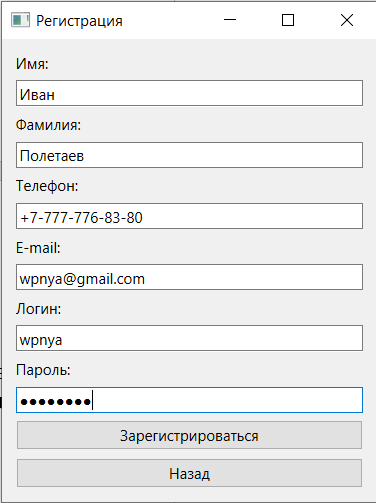


Рисунок 34 – Регистрация клиента в системе

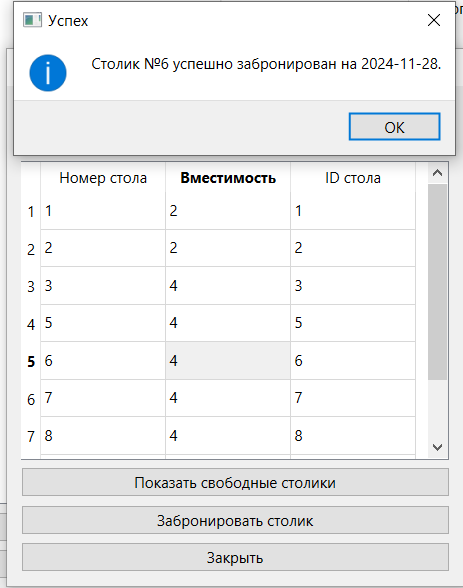


Рисунок 36 – Бронирование стола

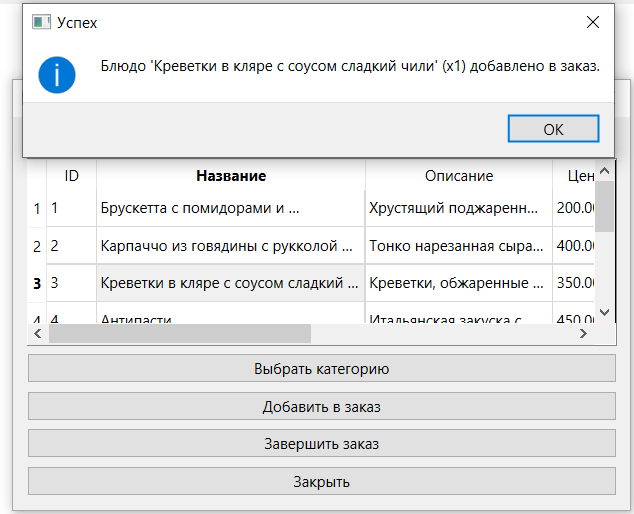


Рисунок 37 – Добавление блюда в корзину

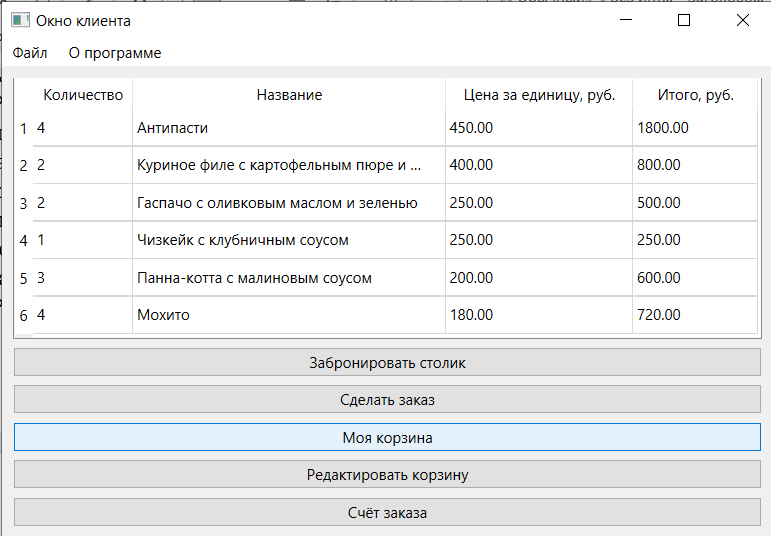


Рисунок 38 – Просмотр заказа

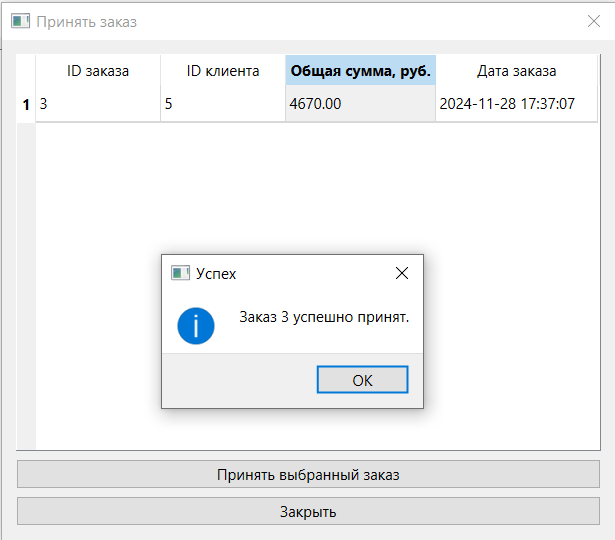


Рисунок 39 – Принятие заказа

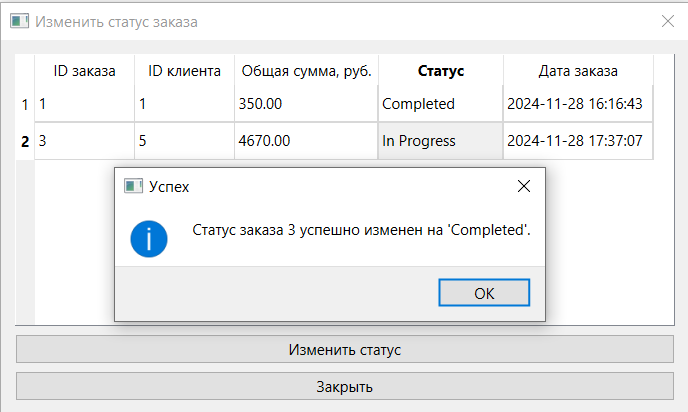
****

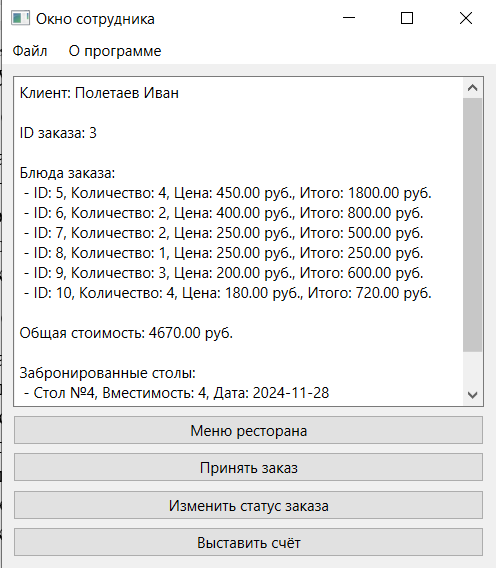
Рисунок 40 – Изменение статуса заказа 

Рисунок 41 – Выставление счёта

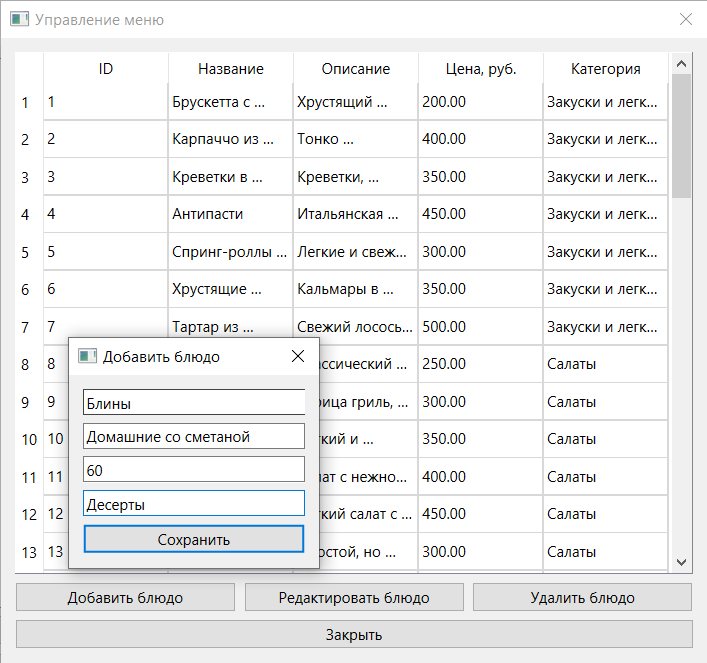


Рисунок 42 – Создание нового блюда

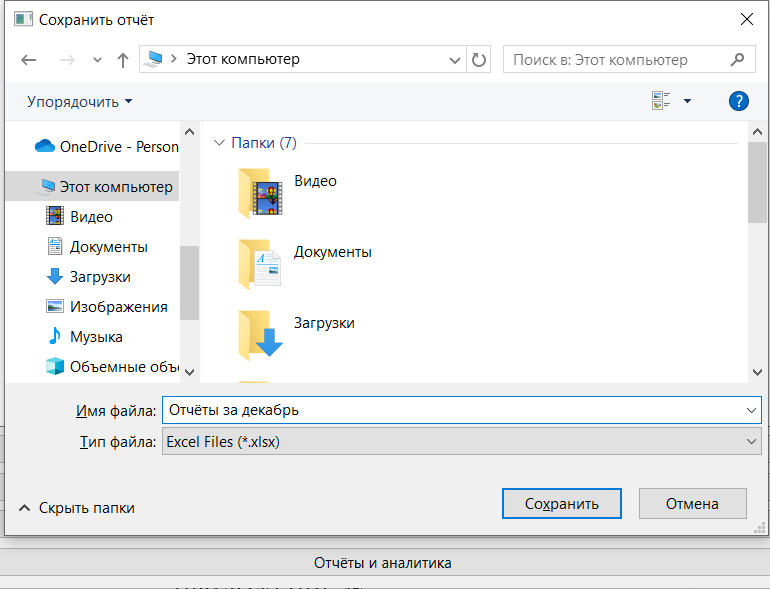


Рисунок 43 – Создание отчёта