

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ПЕДАГОГИКИ, ПСИХОЛОГИИ И СОЦИОЛОГИИ
КАФЕДРА СОВРЕМЕННЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ

УТВЕРЖДАЮ

Заведующий кафедрой

_____ И. А. Ковалевич

подпись инициалы, фамилия

« _____ » _____ 2018 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.03.19 – Прикладная информатика в социальных коммуникациях

код – наименование направления

Создание базы данных для ресторана

Руководитель

подпись, дата

должность, ученая степень

М.М. Манушкина

инициалы, фамилия

Выпускник

подпись, дата

Егоян Д.З.

инициалы, фамилия

Красноярск 2018

РЕФЕРАТ

Выпускная квалификационная работа содержит 51 страниц, 41 рисунка, 2 таблиц, 20 источников, 1 приложения.

БАЗА ДАННЫХ, РЕСТОРАН, УПРАВЛЕНИЕ БАЗАМИ ДАННЫХ.

Объектом исследования является автоматизация на основе баз данных.

Предметом исследования являются методы проектирования и использования баз данных для ресторана.

Цель работы – создание базы данных для ресторана.

В ходе выполнения бакалаврской работы была рассмотрена структура организации ООО «Инвест Мастер», проанализирована методология проектирования баз данных, проанализирована предметная область организации ООО «Инвест Мастер» и существующие информационные системы с базой данных для ее автоматизации, выполнено проектирование базы данных для автоматизации ресторана организации ООО «Инвест Мастер».

Результатом бакалаврской работы является разработанная база данных для ресторана организации ООО «Инвест Мастер».

СОДЕРЖАНИЕ

Введение.....	4
1 Теоретические основы разработки баз данных.....	6
1.1 Анализ методологии проектирования баз данных	6
1.2 Анализ организации ООО «Инвест Мастер»	13
1.3 Обоснование выбора СУБД	15
1.4 Выбор технологии.....	18
2 Создание базы данных для ресторана организации ООО «Инвест Мастер» ..	25
2.1 Проектирование базы данных.....	25
2.2 Описание разработанного продукта.....	38
Заключение	45
Список сокращений	47
Список использованных источников	48
Приложение А.	50

ВВЕДЕНИЕ

Целью выпускной квалификационной работы по теме: «Создание базы данных для ресторана» является анализ технологий проектирования баз данных, программных средств и методов их практического использования при организации базы данных для ресторана.

Степень разработанности проблемы. Изучение и анализ литературных источников [2,3,4,5,8,11,16,17] позволили сделать вывод, что в последнее время становится все острее проблема автоматизации предприятий питания для повышения его конкурентоспособности, следовательно, создания и использования соответствующих информационных систем. Интенсивное развитие информационных систем, баз данных и сетевых технологий требуют постоянного детального изучения применительно к организации системы хранения данных ресторана.

Актуальность темы обусловлена тем, что в последнее время информационные системы и базы данных стали неотъемлемой частью деятельности предприятия питания и одной из основных целей развития современных информационных технологий является повышение эффективности таких организаций. Современный подход к управлению предполагает высокий уровень капиталовложений в информационные технологии. При этом возникает проблема эффективного построения базы данных для ресторана. Поэтому усовершенствуются существующие методы проектирования баз данных и методы организации недорогих и эффективных решений для сетевых систем.

При создании данной выпускной квалификационной работы в качестве источников были взяты труды таких авторов как: Балдин К.В., Божко В.П., Кузин, А.В., Новиков Б.А., Домбровская Г.Р., Гуцин А.Н., Карпова Т. Важным источником информации являлись и статьи, размещенные на интернет сайтах: www.bizhit.ru, www.intuit.ru.

Объектом исследования является автоматизация на основе баз данных.

Предметом исследования являются методы проектирования и использования баз данных для ресторана.

Задачи выпускной квалификационной работы:

- Проанализировать методологии проектирования баз данных;
- Исследовать применение информационных систем и баз данных для автоматизации организации ООО «Инвест Мастер»;
- Проанализировать предметную область организации ООО «Инвест Мастер» и существующие информационные системы с базой данных для ее автоматизации;
- Выполнить проектирование базы данных для автоматизации ресторана организации ООО «Инвест Мастер».

Методы исследования, использованные в процессе выполнения работы и послужившие инструментом в получении необходимого фактического материала: изучения и анализа научной литературы и практики, моделирования, сравнения и синтеза.

Практическая значимость. Практическая значимость работы заключается в выявленных методах применения информационных систем и баз данных для автоматизации ресторана, а также в разработанной базе данных для ресторана, что позволит предприятию повысить общую эффективность работы и конкурентоспособность. А также в возможности использования результатов исследования, выработанных рекомендаций в практической деятельности предприятий.

Данная выпускная бакалаврская работа состоит из введения, двух глав, заключения, списка использованных источников и приложений.

1 Теоретические основы разработки баз данных

1.1 Анализ методологии проектирования баз данных

Существует много разновидностей методологии рассмотрения баз данных в классическом подходе, однако чаще всего придерживаются методологии ANSI/SPARC. На рисунке 1.1 показана совокупность процедур проектирования централизованной базы данных, которые можно объединить в четыре этапа [6,8,11].



Рисунок 1.1 – Схема этапов проектирования БД

На первом этапе формулируются и анализируются требования: задается основная цель предприятия, определяется набор требований к базе данных. В этот набор входят общие и специфические требования. Чтобы сформировать специфические требования применяют методику интервьюирования персонала на различных уровнях управления. В результате создается документ с требованиями в форме, понятной конечным пользователям и проектировщикам баз данных.

На втором этапе производится концептуальное проектирование – описываются и обобщаются информационные требования пользователей в первоначальном проекте базы данных. В качестве исходных данных при классическом подходе используют совокупность документов пользователя, а для современного подхода используют алгоритмы приложений (алгоритмы бизнес-процессов). В результате получают в виде системы таблиц базы данных высокоуровневое представление информационных требований пользователей на основе этих подходов. При этом должна быть выбрана модель базы данных.

Определяется в каком режиме будет осуществляться управление базой данных: однопользовательском или многопользовательском. Процедуры управления многопользовательского режима и распределенных баз данных сильно усложняются. В этих режимах работы возможен одновременный доступ нескольких пользователей и необходимы специальные меры обеспечения целостности баз данных. Для обеспечения целостности на этапе физического проектирования необходимо обеспечить систему обработки транзакций и режим блокировки таблиц или отдельных записей.

На третьем этапе проводят логическое проектирование – для полученного высокоуровневого представления данных выполняется преобразование в структуру используемой системы управления базами данных (СУБД). Основная цель этапа заключается в устранении избыточности данных на основе правил нормализации.

Целью нормализации является минимизация повторения данных и возможных структурных изменений базы данных в процессе обновлений. Этого

можно достигнуть за счет разделения (декомпозиции) одной таблицы на две или более с последующим применением в запросе операций навигации. В результате, полученную логическую структуру базы данных можно будет количественно оценить на основе разных характеристик (числа обращений к логическим записям, объема данных в каждом приложении, общего объема данных). С помощью таких оценок можно затем усовершенствовать логическую структуру для получения большей эффективности.

На четвертом этапе производится физическое проектирование, решение вопросов, связанных с производительностью СУБД, определяется структура хранения данных и методы доступа.

Осуществление взаимодействия между этапами проектирования и словарной системой требует отдельного рассмотрения. Операции проектирования можно выполнять независимо при отсутствии словарной системы. Саму словарную систему можно считать элементом автоматизации проектирования.

В ходе всех этапов разработки необходимо использовать средства проектирования и оценочные критерии. Трудность описания и идентификации большого количества альтернативных решений обуславливает неопределенность при выборе таких критериев, что вызывает наибольшие трудности при проектировании баз данных [9,11].

Гораздо проще использовать с количественные критерии, такие как время ответа на запрос, стоимость модификации, стоимость памяти, время на создание, стоимость реорганизации. Затруднения могут вызвать лишь противоречия между критериями.

Но, кроме того, существует множество критериев оптимальности, отражающих неизмеримые свойства и трудно выразимые в количественном представлении или в виде целевой функции.

Качественными критериями являются критерии гибкости, адаптивности, доступности для новых пользователей, совместимости с другими системами,

возможности конвертирования в другую вычислительную среду, возможности восстановления, возможности распределения и расширения.

В процессе создания базы данных необходима интеграция данных, которые должны обеспечить решение группы прикладных задач различных пользователей. Следовательно, в ходе интеграции данных необходимо учитывать требования к данным каждого из пользователей, которые основываются на представлениях конкретного пользователя о данных и связях между ними. Затем надо обобщить собранные требования в единое представление, которое и станет основой для проектирования единой базы данных [11].

Полученное обобщенное представление всех пользователей о данных является концептуальной моделью базы данных. В концептуальной модели содержится информационное описание предметной области с учетом выявленных логических взаимосвязей, поэтому она получила еще одно название – инфологическая (информационно-логическая) модель. Модель не включает какие-либо понятий, связанных с компьютерами, памятью компьютера, способами хранения данных в памяти компьютера, и, фактически является моделью только предметной области.

Далее требуется выполнить преобразование представлений концептуальной модели с учетом модели данных конкретной системы управления базами данных (СУБД) в логическую модель базы данных. Таким образом, логическая модель является концептуальной схемой, специфицированной в языке описания данных конкретной СУБД. В логической модели представление данных и элементов данных отражено вне зависимости от их содержания и среды хранения.

Затем разработчику необходимо с помощью средств конкретной СУБД выполнить отображение полученной логической модели базы данных в память компьютера и определить методы доступа к данным. В результате будет получено представление данных в памяти компьютера, называемое внутренним представлением или структурой хранения.

Прикладные приложения взаимодействуют с логической моделью, при этом конкретным пользователям предоставлено подмножество этой логической модели, которое отражает представления пользователя о предметной области. Каждому прикладному приложению "видны" и доступны для обработки только те данные, которые необходимы именно ему.

Соответствующая интерпретация данных прикладными приложениями (пользователями) представляет собой внешние представления. Взаимосвязь таких трех уровней представлений отображена на рисунке 1.2 [10,11].

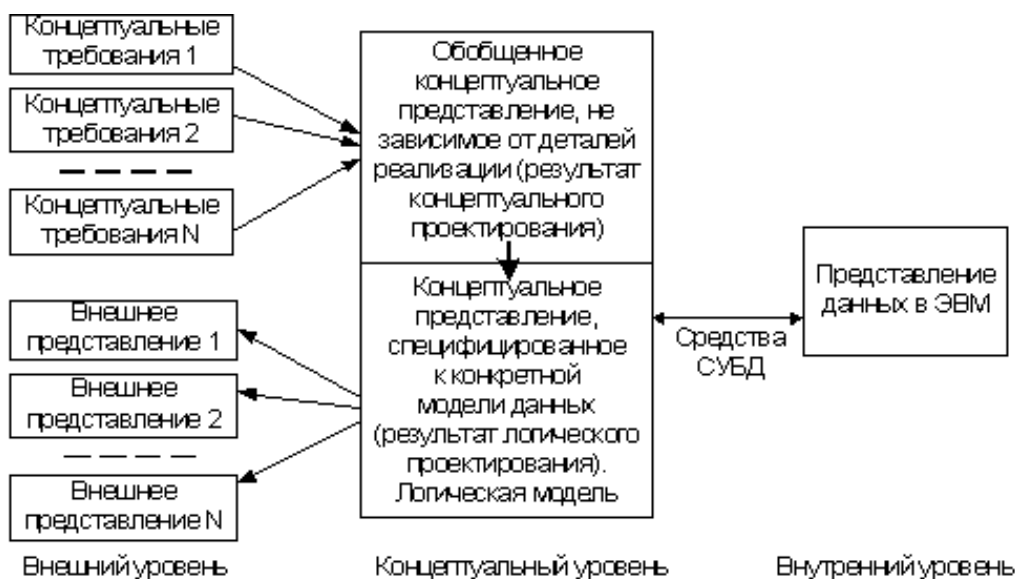


Рисунок 1.2 – Уровни представлений о данных в базе данных

В соответствии с этим в трехуровневой архитектуре ANSI/SPARC определено выделение трех различных уровней описания данных (внешнего, концептуального, внутреннего) в модели системы управления базами данных (Рисунок 1.3).

Уровень внешних моделей — на этом самом верхнем уровне определяется свое «видение» данных для каждой модели. На этом уровне определена точка зрения на базу данных для каждого отдельного приложения. Для каждого приложения определена видимость и возможность обработки только тех данные, которые необходимы именно этому приложению. Например, в системе распределения рабочих заданий используется информация о квалификации и должностных обязанностях сотрудников, но совершенно не

нужны данные о количестве детей, адресе проживания и телефоне сотрудников, и наоборот, именно эта информация находит применение в подсистеме отдела кадров [1,11,25].



Рисунок 1.3 – Схема трехуровневой модели системы управления базами данных, предложенной ANSI

Концептуальный уровень является центральным управляющим звеном, в котором отражено в наиболее общем виде представление базы данных, объединяющее набор данных, используемых всеми приложениями, которые работают с этой базой данных. Фактически на концептуальном уровне отражена обобщенная модель предметной области (объекты реального мира), для которой выполнялось проектирование базы данных. Подобно любой модели, в концептуальной модели отражены только существенные, с точки зрения обработки, свойства объектов реального мира.

Физический уровень является совокупностью данных, которые хранятся в файлах или в табличных структурах, находящихся на внешних накопителях данных.

В этой архитектуре обеспечивается логическая независимость между 1-м и 2-м уровнями и физическую независимость между 2-м и 3-м уровнями при работе с данными. Логической независимостью определяются возможности

изменений одного приложения без корректировки остальных приложений, которые работают с этой же базой данных. Физической независимостью определяются возможности переноса хранящихся данных с одних носителей на другие с сохранением работоспособности всех приложений, которые работают с данной базой данных. Именно эта особенность отсутствовала при организации обработки данных с использованием файловых систем. Выделение концептуального уровня дало возможность разработки аппарата централизованного управления базой данных [9,13,16].

Основной целью рассматриваемой архитектуры является отделение пользовательских представлений о данных в базе данных от их физического представления. Оперирование такими представлениями о данных способствует обеспечению выполнения основного требования к базе данных – независимости программ и данных. В процессе изменения прикладных приложений могут изменяться и соответствующие внешние представления, но логическую модель данных изменять не нужно и, соответственно, не требуется изменение других прикладных приложений. С другой стороны, при изменении внутренних представлений (структур хранения) логическая модель также не требует изменений, соответственно, не изменяются прикладные приложения.

Применение соответствующих представлений дает возможность четкого разграничения полномочий разных лиц, которые работают с базой данных [11,14,15]:

- внешнее представление – совокупность представлений специалистов предметной области (пользователей);
- внешнее представление и логическая модель – совокупность представлений прикладных программистов, разрабатывающих конкретное приложение для пользователей;
- логическая модель и внутреннее представление – совокупность представлений системных программистов, администрирующих базу данных.

При проектировании реляционных моделей данных разрабатываются ER-диаграммы, реляционная схема и сопроводительная документация. Для проверки корректности построенной логической модели данных используются правила нормализации. Нормализация является эффективным средством, которое позволяет удостовериться структурную согласованность, логическую целостность и минимальную избыточность выбранной модели данных [27,30].

А также выполняется дополнительная проверка модели данных, чтобы выявить возможность осуществления транзакций, которые будут производиться пользователями проектируемого приложения. Такой комплекс проверок позволит обрести необходимую уверенность в приемлемости принятой модели данных. А каждую локальную логическую модель данных при необходимости можно применить для подготовки прототипов реализаций базы данных, которые предназначены для отдельных групп пользователей приложения.

Таким образом, проектирование баз данных является длительным и трудоемким процессом, занимающим сотни человеко-часов и большое число циклов улучшений и доработок. При этом, в качестве основных ресурсов проектировщика баз данных выступают его собственная интуиция и опыт, что достаточно часто может негативно сказаться на качестве проектируемых решений.

Можно выделить основные причины низкой эффективности проектируемых баз данных:

- недостаточно глубокая степень анализа требований на начальных этапах проектирования, включая их семантический аспект и взаимосвязи данных;
- длительный и утомительный процесс структурирования данных, трудно выполняемый при ручной обработке информации.

Это обуславливает необходимость автоматизации разработки баз данных на разных этапах проектирования.

1.2 Анализ организации «Инвест Мастер»

ООО «Инвест Мастер» – частное предприятие, созданное для получения прибыли от посреднической деятельности производства, реализация и организация потребления продукции питания. Основным видом деятельности компании является деятельность ресторанов и услуги по доставке продуктов питания.

Основными услугами являются:

- Сочетание вышеназванных функций с организацией отдыха, развлечений и досуга потребителей;
- Предоставление продукции и услуг высокого качества по ценам, соответствующие классу ресторанов;
- Предоставление дополнительных сопутствующих услуг: услуги по доставке еды по адресу, сервисные услуги на дому, в офисе и т. д.

Первые русские заведения стали появляться в первое двадцатилетие XIX века исключительно при отелях и рассчитаны были на иностранных гостей и местных аристократов. И только в 1873 году в Москве открылся легендарный «Славянский базар» - первый публичный ресторан, удививший публику не только роскошным интерьером, но и работающими во фраках и белых перчатках официантами. На сегодняшний день данный сегмент рынка является достаточно перспективным. Поэтому в ресторанный бизнес инвестируют средства представители известных компаний, специализирующихся на добыче энергоресурсов, а также пивовары и производители продуктов питания. Однако, как и любая предпринимательская деятельность, открытие ресторана или другой точки общественного питания сопряжено с некоторыми трудностями, основными из которых являются:

- высокая конкуренция;
- сложности при открытии и развитии заведения;
- увеличение показателя доходности.

Однако, несмотря на все вышеперечисленные факторы, перспективы для развития точки общественного питания в России достаточно неплохие, поэтому организация точки общественного питания – это перспективное направление бизнеса.

1.3 Обоснование выбора СУБД

При реализации проекта в качестве СУБД был выбран – MS Access 2013, включающий необходимый функциональный набор средств для проектирования базы данных ресторана.

Приложение Microsoft Access является мощной и высокопроизводительной 32-разрядной системой управления реляционной базой данных (далее СУБД).

База данных – это совокупность структурированных и взаимосвязанных данных и методов, обеспечивающих добавление выборку и отображение данных.

Реляционная база данных. Практически все СУБД позволяют добавлять новые данные в таблицы. С этой точки зрения СУБД не отличаются от программ электронных таблиц (Microsoft Excel), которые могут эмулировать некоторые функции баз данных. Существует три принципиальных отличия между СУБД и программами электронных таблиц:

СУБД разрабатываются с целью обеспечения эффективной обработки больших объёмов информации, намного больших, чем те, с которыми справляются электронные таблицы.

СУБД может легко связывать две таблицы так, что для пользователя они будут представляться одной таблицей. Реализовать такую возможность в электронных таблицах практически невозможно.

СУБД минимизируют общий объём базы данных. Для этого таблицы, содержащие повторяющиеся данные, разбиваются на несколько связанных таблиц.

Access – мощное приложение Windows. При этом производительность

СУБД органично сочетаются со всеми удобствами и преимуществами Windows.

Как реляционная СУБД Access обеспечивает доступ ко всем типам данных и позволяет одновременно использовать несколько таблиц базы данных. Можно использовать таблицы, созданные в среде Paradox или dBase. Работая в среде Microsoft Office, пользователь получает в своё распоряжение полностью совместимые с Access текстовые документы (Word), электронные таблицы (Excel), презентации (PowerPoint). С помощью новых расширений для Internet можно напрямую взаимодействовать с данными из World Wide Web и транслировать представление данных на языке HTML, обеспечивая работу с такими приложениями как Internet Explorer и Netscape Navigator.

Access специально спроектирован для создания многопользовательских приложений, где файлы базы данных являются разделяемыми ресурсами в сети. В Access реализована надёжная система защиты от несанкционированного доступа к файлам.

Созданное в Microsoft Access приложение может выступать как в роли OLE- контролера, так и OLE сервера. То есть, возможность осуществить контроль работы приложений MS Access из любого приложения, с условием, что это приложение сможет выступить в роли OLE контролера или OLE сервера. Наличие встроенного языка структурированных запросов SQL обеспечивает максимально гибкую работу с данными и значительное ускорение доступа к внешним данным.

Ресурсов Microsoft Access достаточно для обслуживания всего делопроизводства малого предприятия. Все сотрудники предприятия будут работать по сети с одной базой данных, которая может быть установлена на рабочей станции или на выделенном сервере.

Для исключения проблем целостности и доступа к данным надо использовать предоставляемые MS Access средства защиты. Возможно с помощью специального мастера правильно установить права и ограничения для пользователей базы данных. В Microsoft Access имеется лучшая встроенная система защиты среди всех рассмотренных настольных приложений СУБД. Эта

система защиты обеспечит создание групп, отдельных пользователей, присвоение прав доступа ко всем объектам, в том числе и к программным модулям. Этим решаются вопросы скрытия от пользователей содержимого модулей процедур и функций [16, 18].

Элементы базы данных.

Таблицы. В базе данных информация хранится в виде двумерных таблиц. Можно так же импортировать и связывать таблицы из других СУБД или систем управления электронными таблицами. Одновременно могут быть открыты 1024 таблицы.

Запросы. При помощи запросов можно произвести выборку данных по какому-нибудь критерию из разных таблиц. В запрос можно включать до 255 полей.

Формы. Формы позволяют отображать данные из таблиц и запросов в более удобном для восприятия виде. С помощью форм можно добавлять и изменять данные, содержащиеся в таблицах. В формы позволяют включать модули.

Отчёты. Отчёты предназначены для печати данных, содержащихся в таблицах и запросах, в красиво оформленном виде. Отчёты так же позволяют включать модули.

Макросы, модули. Модули содержат VBA-код, используемый для написания процедур обработки событий, таких как например нажатия кнопки в форме или отчёте, для создания функций настройки, для автоматического выполнения операций над объектами базы данных и программного управления операциями, т.е. добавление VBA-кода позволяет создать полную базу данных с настраиваемыми меню, панелями инструментов и другими возможностями. Модули снимают с пользователя приложения необходимость помнить последовательность выбора объектов базы данных для выполнения того или иного действия и повышают эффективность работы.

База данных может содержать до 32768 объектов.

В состав Access входит множество мастеров, строителей и надстроек,

которые позволяют упростить процесс создания объектов базы данных.

1.4 Выбор технологии

Рассмотрим для сравнения несколько технологий, использование которых при разработке приложений обеспечит клиентам доступ к базе данных и возможности манипулирования данными.

ADO (ActiveX Data Objects – объекты данных ActiveX) – интерфейс программирования приложений для доступа к данным, разработанный компанией Microsoft и основанный на технологии компонентов ActiveX. ADO позволяет представлять данные из разнообразных источников (реляционных баз данных, текстовых файлов) в объектно-ориентированном виде.

Источниками данных технологии ADO могут быть (рисунок 1.5):

- серверные БД;
- файловые БД;
- текстовые , XML , двоичные файлы.
- таблицы Excel;
- серверы электронной почты;
- базы данных журналов Windows и других ОС.



Рисунок 1.4 – Источники данных ADO

В общем, все объекты, которые могут использовать OLE DB провайдер.

Объекты ADO. В основу интерфейса ADO положен набор объектов, гораздо более простых в использовании, чем объекты OLE DB. Хотя структура объектов ADO аналогична OLE DB, объекты ADO не являются столь же зависимыми от их иерархии. В большинстве случаев можно просто создать и использовать лишь объекты, необходимые для работы, и не заботиться о создании многих других лишних "родительских" объектов.

Рассмотрим основные объекты интерфейса ADO.

ADO использует объекты Connection для представления отдельных соединений с источником данных. Если используется ADODB, то эти объекты представляют соединения с источником данных OLE DB. Если же используется MSDASQL, то источник данных OLE DB может соответствовать источнику данных ODBC. Соединение может представлять собой реальное сетевое соединение с сервером базы данных или с локальным файлом базы данных Microsoft Access.

Для выполнения любых операций над источником данных требуется объект Connection, хотя не обязательно создавать его самостоятельно. Зачастую можно просто позволить ADO создать соединение, используемое объектами Command и Recordset. Тем не менее многие операции можно выполнить лишь с использованием объекта Connection. Эти объекты также применяются для создания единого соединения, используемого несколькими другими объектами, что может пригодиться для оптимизации приложений. Объекты Connection служат также для управления транзакциями в ADO.

Объекты Command используются для представления конкретных команд, которые выполняются над источниками данных.

Объекты Recordset обеспечивают взаимодействие с данными, используются для хранения набора записей, возвращаемого из источника данных.

Объекты Field и коллекции Fields. Класс Recordset содержит коллекцию объектов Field, используемых для работы с отдельными столбцами группы строк. С каждым столбцом в группе строк связан объект Field этой коллекции,

который обеспечивает доступ к метаданным столбца, таким как имя столбца и тип данных, а также к реальному значению данного столбца в текущей строке.

Объекты `Parameter` и коллекция `Parameters`. Объекты `Command` содержат коллекцию `Parameters`, включающую все связанные с командой параметры. Каждый отдельный объект `Parameter` служит для хранения информации о параметре, который передается команде во время выполнения или возвращается процедурой команды. Новые объекты `Parameter` можно создать с использованием метода `CreateParameter()` объекта `Command`. Объекты `Property` и коллекция `Properties`. Объекты `Connection`, `Command`, `Recordset` и `Field` содержат коллекцию `Properties`, которая используется для получения отдельных объектов `Property`, связанных с данным объектом. Объекты `Property` служат для представления отдельных дополнительных параметров или других характеристик объекта ADO, которые не могут управляться встроенными свойствами объекта. Хотя каждый объект ADO может поддерживать различные типы свойств, все они управляются с использованием стандартного объекта `Property`.

Объекты `Error` коллекция `Errors`. Объекты ADO `Connection` содержат коллекцию `Errors`, включающую объекты `Error` с конкретной информацией об ошибках, которые могут происходить во время соединения в рамках одной операции. В большинстве случаев объекты `Error` генерируются только при возврате ошибки системой базы данных, а не при ошибках ADO во время выполнения процедур.

Можно сделать вывод, что использование технологии ADO позволяет упростить проектирование приложения за счет концептуальной модели представления данных. Дает возможность создать независимость модели данных приложения от модели базы данных и средства для решения проблем, возникающих из-за изменений в структуре базы данных.

Использование технологии ODBC. Стандарт ODBC является программным интерфейсом, обеспечивающим прикладным программам возможность обращения к SQL базе данных и обработки их независимым от

СУБД способом. То есть, клиентские приложения, использующие интерфейс ODBC, способны подключиться к базе данных ORACLE, SYBASE, INFOMIX и любой другой базе данных, совместимой с ODBC, и выполнять операции обработки данных изменения в программном коде.

Рассмотрим применение компонентов стандарта ODBC для приложений, которые используют интернет-технологии (рисунок 1.9). Компьютер с Web-сервером содержит также прикладную программу, диспетчер драйверов и драйверы СУБД. Диспетчер драйверов посредством драйверов конкретной СУБД передает запросы приложения источнику данных, находящемуся на сервере баз данных. Приложение запрашивает создание соединения с источником данных, а затем запросы выполнения SQL-операторов и получения результатов, обработки ошибок и операций начала, фиксации и отката транзакций. Технологии ODBC обеспечивают стандартные способы выполнения запросов и определяют стандартный набор кодов ошибок и сообщений.



Рисунок 1.5 – Организация доступа к базам данных разных СУБД

Диспетчер драйверов является посредником между приложением и драйверами СУБД. При запросе приложением соединения, диспетчером определяется тип СУБД, обрабатывающий указанный источник данных ODBC, и загружает соответствующий драйвер в память (если это не произошло ранее). Диспетчером драйверов также производится обработка

определенных запросов инициализации, проверка допустимости форматов и порядка запросов к ODBC, получаемых от приложений.

Требуемым драйвером обрабатываются запросы к ODBC и обеспечивается передача конкретных SQL- операторов на выполнение операций заданным источником данных. Драйверы ODBC входят в поставку конкретной СУБД и разработаны ее производителем.

В MS Access также возможна организация связей с внешними данными на основе интерфейса ODBC. Отображение и функционирование связанных таблиц происходит аналогично собственным таблицам. Посредством связанных таблиц можно обеспечить организацию взаимодействия с большинством распространенных СУБД клиент - серверной архитектуры, включая MS SQL Server, Oracle, Informix, IBM DB2, Sybase.

В MS Access предоставляются разные способы взаимодействия приложений на базе интерфейса ODBC:

- при помощи запросов на языке SQL к базе данных SQL Server обеспечивается выполнение операций выборки данных из таблиц, добавления, обновления данных и структур таблиц сервера, выполняемых непосредственно на сервере;

- при помощи связи с таблицами базы данных SQL Server, с отображением таблиц сервера в окне программы MS Access, чтобы можно было просмотреть и изменить их средствами MS Access, использовать в запросах, формах, отчетах, а также выполнить объединение внешних данных с данными из объектов MS Access;

- с помощью импорта объектов SQL Server в базу данных MS Access, с созданием копий объектов и неизменным состоянием исходных объектов источника;

- с помощью экспорта локальных таблиц базы данных MS Access в базу данных в общую базу данных SQL Server.

MS Access 2013 является также платформой для создания веб-приложений, с последующим их совместным использованием с помощью браузера в пределах интрасети предприятия или в Интернет.

Проектирование веб-страниц приложения выполняется аналогично созданию форм MS Access с функциями ввода и редактирования информации, отчетов MS Access с отображением иерархически сгруппированной информации. Для взаимодействия с созданным веб-приложением на компьютерах пользователей требуется только браузер, установка MS Access не нужна.

Разработанное веб-приложение может быть опубликовано как на сайте собственного сервера SharePoint 2013 в интрасети, так и на сайтах SharePoint, которые предоставляет Office 365 в "облаке". Базы данных веб-приложений должны быть сохранены на SQL Server 2012, который определен в SharePoint. Административные функции обеспечения безопасности доступа и управления данными осуществляются средствами SharePoint и SQL Server.

Можно создать корпоративную сеть, также базирующуюся на рассмотренных клиент-серверных и интернет-технологиях. Такая сеть может быть подключена или не подключена к Интернет (рисунок 1.10).

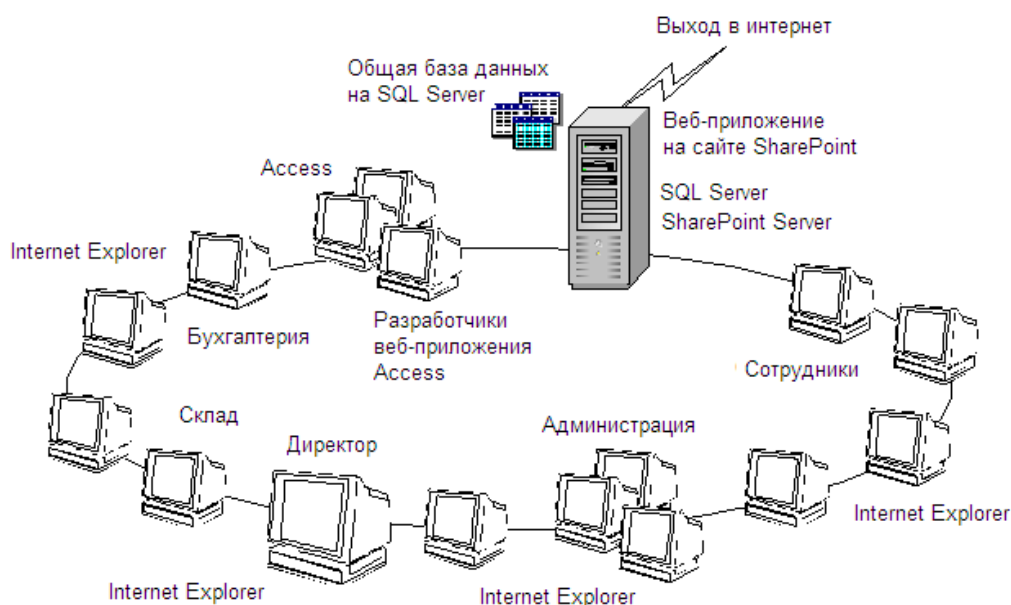


Рисунок 1.6 – Корпоративная сеть с общей базой данных веб-приложения

Применение Интернет технологий обеспечит сотрудников доступом к данным всего предприятия со своих рабочих мест, не думая о совместимости аппаратных и программных платформ, используя обычный веб-браузер.

Средствами SharePoint производится устранение всех проблем, связанных с параллельным доступом к данным.

Работа с приложением базы данных через браузер выполняется на основе разрешений SharePoint, определяющих возможность доступа к тем или иным объектам.

Кроме того, средствами SharePoint можно воспользоваться для выполнения таких функций, как запись версий данных, подписка на получение предупреждений при внесении изменений. А также обеспечивается эффективный общий доступ к базе данных, а также предоставляются новые возможности для совместной работы через Интернет.

Таким образом, выбрана технология реализации базы данных на MS Access 2013 для регистрации заказов.

А в перспективе предлагается публикация веб-приложения на сайте SharePoint, которые предоставляет Office 365 в "облаке".

2 Создание базы данных для ресторана организации ООО «Инвест Мастер»

2.1 Проектирование базы данных

Проводя инфологическое проектирование, требуется выявить общие информационные объекты и связи между ними, произвести анализ общих информационных требований к системе и выявить информационные потоки, отображающие процессы создания, обработки и взаимодействия данных.

В результате инфологического проектирования создают концептуальную модель, представляющую собой, не зависящую от любой физической реализации структуру данных.

На основе анализа предметной области ООО «Инвест Мастер» выделим информационные объекты и их атрибуты. Выделяем объекты-сущности «Товары», «Типы товара», «Заказы», «Детали заказа», «Адреса», «Цвета», «Марки», «Страны». Рассмотрим атрибуты перечисленных объектов в таблице 2.1.

Таблица 2.1 – Атрибуты объектов

Объект	Атрибуты объектов
«Блюда»	Код блюда Название Количество Категория Состав Цена
«Блюда в заказе»	Код блюда Название Количество Категория Состав Цена

Продолжение таблицы 2.1

«Блюда в предзаказе»	Код блюда Название Количество Категория Состав Цена
«Заказы»	Код заказа Дата Время Место Выполнен Оплачено Примечание
«Заказы в оплате»	Код заказа Код оплаты
«Залы»	№ зала Наименование
«Категории блюд»	Код категории Название категории
«Места»	Код места № места Зал
«Места в предзаказе»	№ пп Код предзаказа Место
«Оплата заказов»	Код оплаты Сумма заказов Оплачено Сдача

Окончание таблицы 2.1

«Предварительные заказы»	Код заказа Дата Время ФИО Контактный телефон Выполнен Сумма заказов Сумма предоплаты Сумма доплаты Сдача Оплачено Примечание
--------------------------	---

Необходимо выполнить анализ каждого атрибута на наличие взаимосвязей с другими реквизитами объекта. Реквизит приобретает смысл, только в том случае, когда связывается с другими атрибутами, которые обладают смысловым единством.

Определим отношения (взаимосвязи) и мощности отношений между объектами.

Отношение «Блюда» -> «Категории блюд». «Категория блюда» является главным объектом, а «Категории блюд» подчинённым объектом. Определен тип связи «один ко многим» (Рисунок 2.1). В одной категории может быть выбрано несколько блюд. Указанные объекты связываются между собой по атрибуту «Код категории».

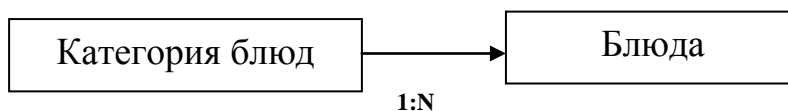


Рисунок 2.1 – Связь между объектами «Категория блюд» и «Блюда»

Отношение «Блюда» -> «Блюда в заказа». «Блюда» является главным объектом, а «Блюда в заказа» подчинённым объектом. Определен тип связи «один ко многим» (рисунок 2.2). Для одного блюда может быть несколько детализированных описаний. Указанные объекты связываются между собой по атрибуту «Код блюда».

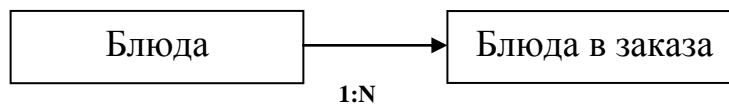


Рисунок 2.2 – Связь между объектами «Блюда» и «Блюда в заказа»

Отношение «Блюда» -> «Блюда в предзаказе». «Блюда» является главным объектом, а «Блюда в предзаказе» подчинённым объектом. Определен тип связи «один ко многим» (рисунок 2.3). Для одного блюда может быть несколько детализированных описаний. Указанные объекты связываются между собой по атрибуту «Код блюда».



Рисунок 2.3 – Связь между объектами «Блюда» и «Блюда в предзаказе»

Отношение «Блюда в предзаказе» -> «Предварительные заказы». «Предварительные заказы» является главным объектом, а «Блюда в предзаказе» подчинённым объектом. Определен тип связи «один ко многим» (рисунок 2.4). Одно блюдо может быть в нескольких предзаказах. Указанные объекты связываются между собой по атрибуту «Код заказа».

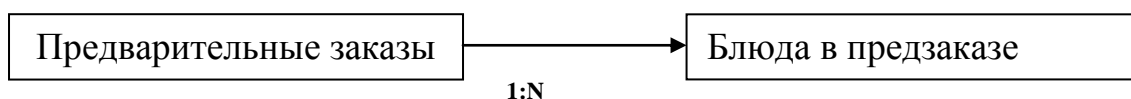


Рисунок 2.4 – Связь между объектами «Предварительные заказы» и «Блюда в предзаказе»

Отношение «Предварительные заказы» -> «Места в предзаказе». «Предварительные заказы» является главным объектом, а «Места в предзаказе» подчинённым объектом. Определен тип связи «один ко многим» (рисунок 2.5). Один заказ может занимать несколько мест. Указанные объекты связываются между собой по атрибуту «Код заказа».

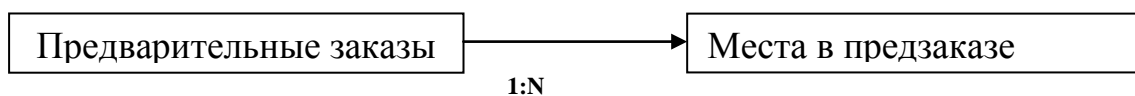


Рисунок 2.5 – Связь между объектами «Предварительные заказы» и «Места в предзаказе»

Отношение «Места в предзаказе» -> «Места». «Места» является главным объектом, а «Места в предзаказе» подчинённым объектом. Определен тип связи «один ко многим» (рисунок 2.6). Одно место может использоваться в нескольких предзаказах. Указанные объекты связываются между собой по атрибуту «Код места».

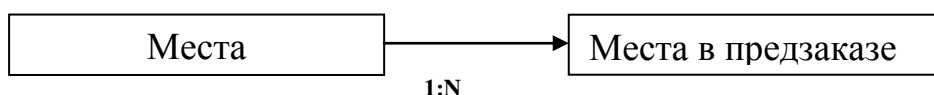


Рисунок 2.6 – Связь между объектами «Места» и «Места в презаказе»

Отношение «Места» -> «Залы». «Залы» является главным объектом, а «Места» подчинённым объектом. Определен тип связи «один ко многим» (рисунок 2.7). К одному залу может относиться несколько мест. Указанные объекты связываются между собой по атрибуту «№ Зала».

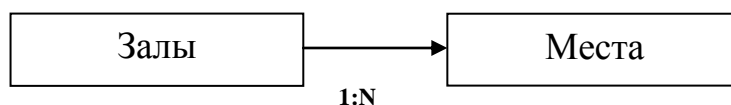


Рисунок 2.7 – Связь между объектами «Залы» и «Места»

Отношение «Блюда в заказе» -> «Заказы». «Заказы» является главным объектом, а «Блюда в заказе» подчинённым объектом. Определен тип связи «один ко многим» (рисунок 2.8). К одному заказу может относиться несколько блюд. Указанные объекты связываются между собой по атрибуту «Код заказа».

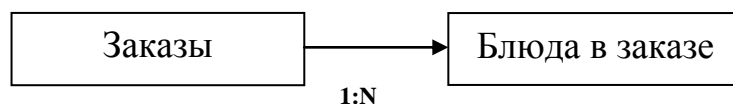


Рисунок 2.8 – Связь между объектами «Заказы» и «Блюда в заказе»

Отношение «Заказы» -> «Заказы к оплате». Определен тип связи «один к одному» (рисунок 2.9). Один заказ можно оплатить только раз. Указанные объекты связываются между собой по атрибуту «Код заказа».

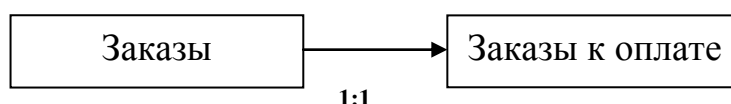


Рисунок 2.9 – Связь между объектами «Заказы» и «Заказы к оплате»

Отношение «Заказы» -> «Места». «Места» является главным объектом, а «Заказы» подчинённым объектом. Определен тип связи «один ко многим» (рисунок 2.10). Одно место может быть в нескольких заказах. Указанные объекты связываются между собой по атрибуту «Код места».

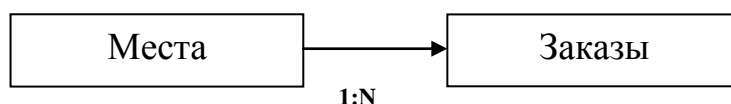


Рисунок 2.10 – Связь между объектами «Места» и «Заказы»

Отношение «Оплата заказов» -> «Заказы в оплате». «Оплата заказов» является главным объектом, а «Заказы в оплате» подчинённым объектом. Определен тип связи «один ко многим» (рисунок 2.11). К одному заказу может относиться несколько оплат. Указанные объекты связываются между собой по атрибуту «Код оплаты».

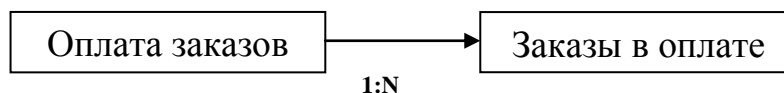


Рисунок 2.11 – Связь между объектами «Оплата заказов» и «Заказы в оплате»

В проектируемой базе данных применяются связи типа один-ко-многим (1:M).

В ходе этапа физического проектирования базы данных разработчик принимает окончательное решение о способах реализации создаваемой базы. Поэтому физическое проектирование необходимо производить, учитывая все особенности выбранной СУБД MS Access. Дatalogическая структура реляционной базы данных определяется совокупностью логически связанных реляционных таблиц. Логические связи соответствуют структурным связям между объектами в концептуальной модели, каждый объект в логической модели отображается соответствующей реляционной таблицей. Связи между таблицами осуществляются посредством общих атрибутов.

Разработанная схема данных MS Access показана на рисунке 2.12.

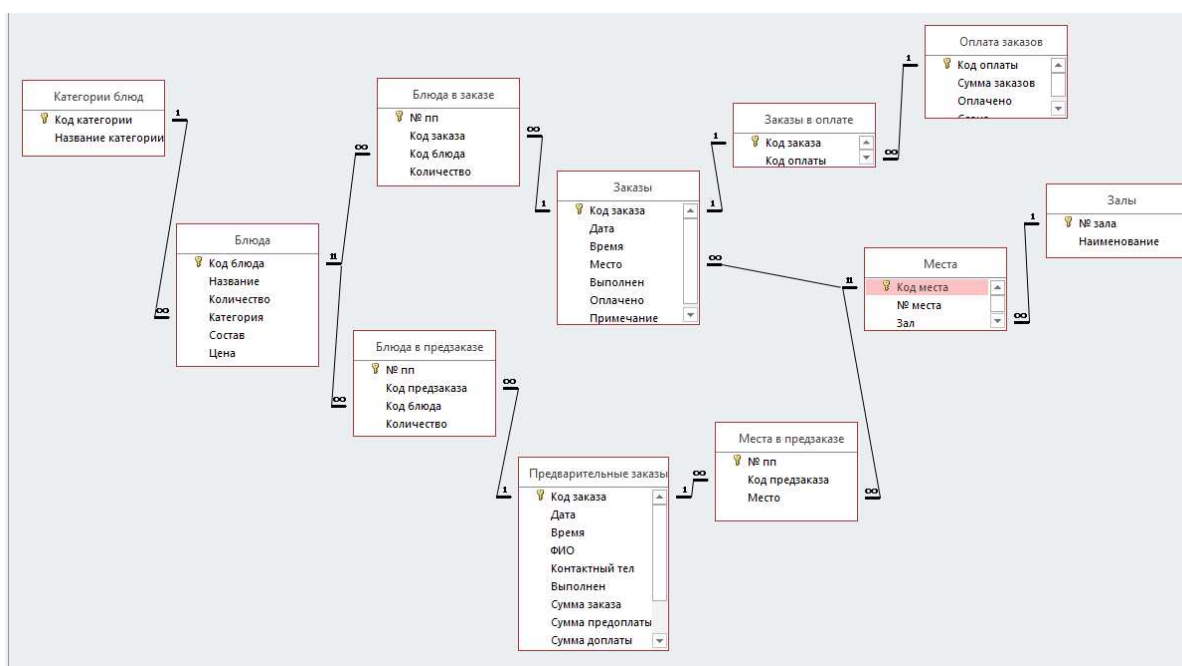


Рисунок 2.12 – Схема данных MS Access

При создании БД в СУБД ACCESS были созданы таблицы, которые заполнялись данными. В таблице «Блюда» созданы следующие поля и заданы типы данных (рисунок 2.13):

Поле «Код блюда» имеет тип данных Счётчик.

Поле «Название» имеет тип данных Короткий текст.

Поле «Количество» имеет тип данных Числовой.

Поле «Категория» имеет тип данных Числовой.

Поле «Состав» имеет тип данных Короткий текст.

Поле «Цена» имеет тип данных Числовой.

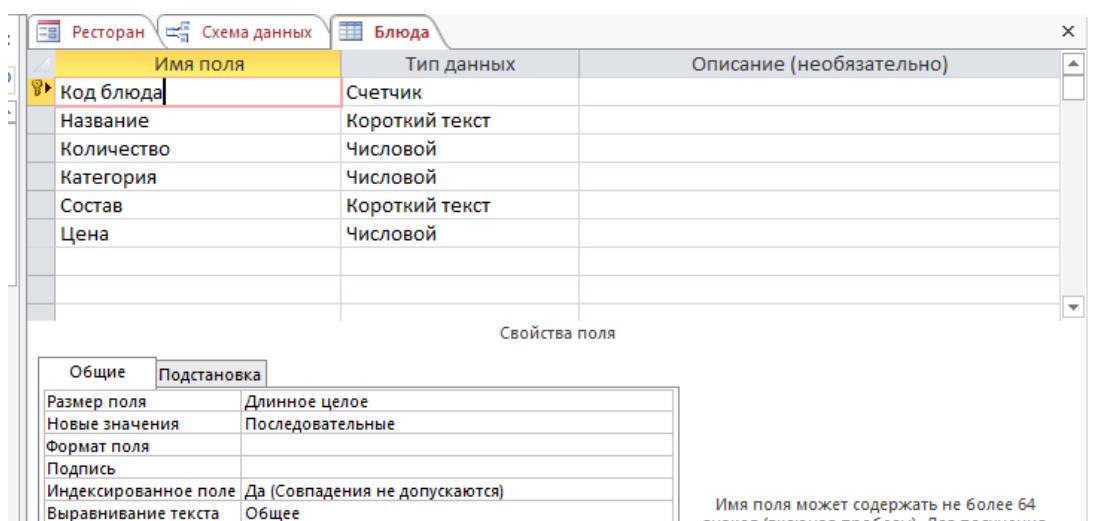


Рисунок 2.13 – Таблица «Блюда» в режиме конструктора

В таблице «Блюда в заказе» созданы следующие поля и заданы типы данных (рисунок 2.14):

Поле «№ пп» имеет тип данных Счётчик.

Поле «Код заказа» имеет тип данных Числовой.

Поле «Код блюда» имеет тип данных Числовой.

Поле «Количество» имеет тип данных Числовой.

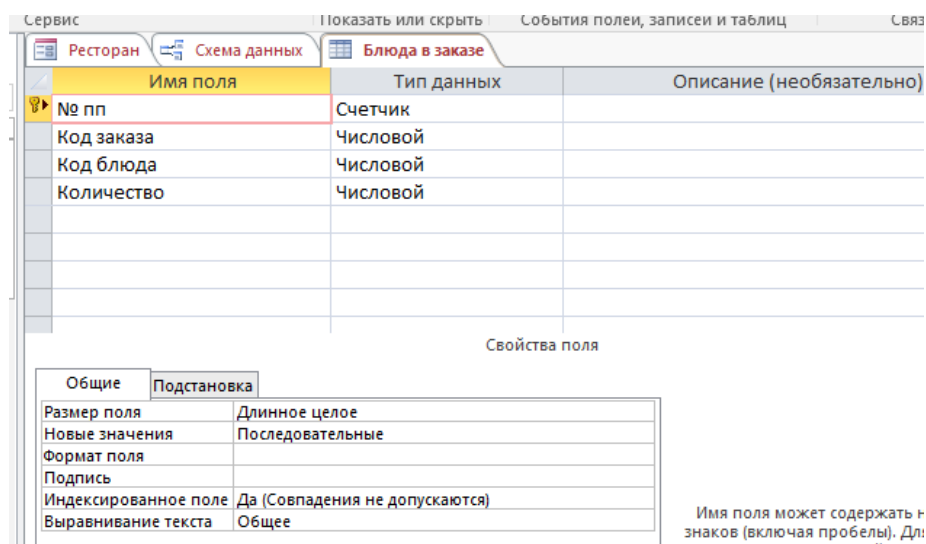


Рисунок 2.14 – Таблица «Блюда в заказе» в режиме конструктора

В таблице «Блюда в предзаказе» созданы следующие поля и заданы типы данных (рисунок 2.15):

Поле «№ пп» имеет тип данных Счётчик.

Поле «Код предзаказа» имеет тип данных Числовой.

Поле «Код блюда» имеет тип данных Числовой.

Поле «Количество» имеет тип данных Числовой.

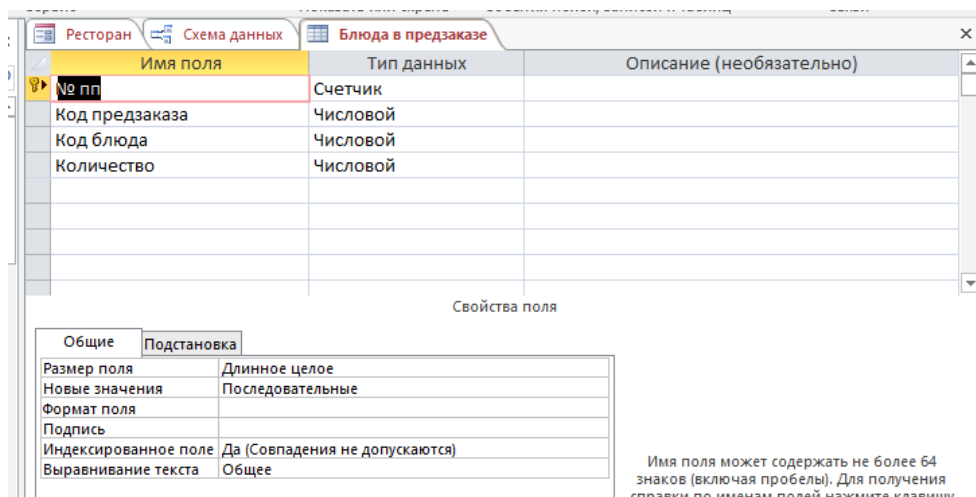


Рисунок 2.15 – Таблица «Блюда в предзаказе» в режиме конструктора

В таблице «Заказы» созданы следующие поля и заданы типы данных (рисунок 2.16):

Поле «Код заказа» имеет тип данных Счётчик.

Поле «Дата» имеет тип данных Дата и время.

Поле «Время» имеет тип данных Дата и время.

Поле «Место» имеет тип данных Числовой.

Поле «Выполнен» имеет тип данных Логический.

Поле «Оплачено» имеет тип данных Логический.

Поле «Примечание» имеет тип данных Длинный текст.

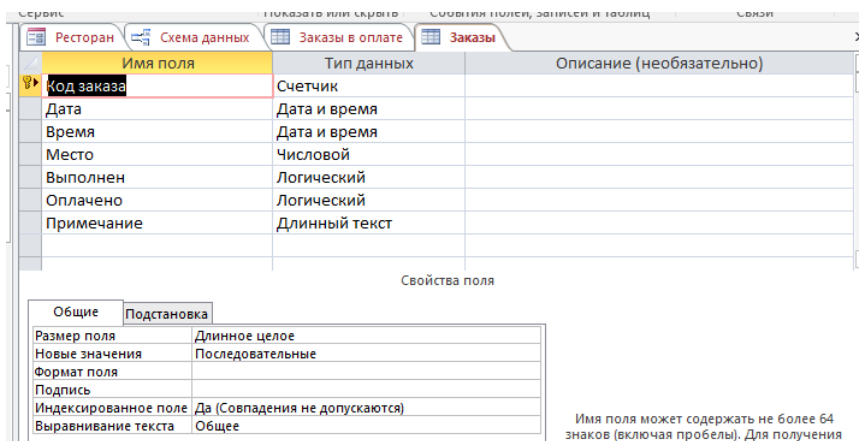


Рисунок 2.16 – Таблица «Заказы» в режиме конструктора

В таблице «Заказы в оплате» созданы следующие поля и заданы типы данных (рисунок 2.17):

Поле «Код заказа» имеет тип данных Числовой.

Поле «Код оплаты» имеет тип данных Числовой.

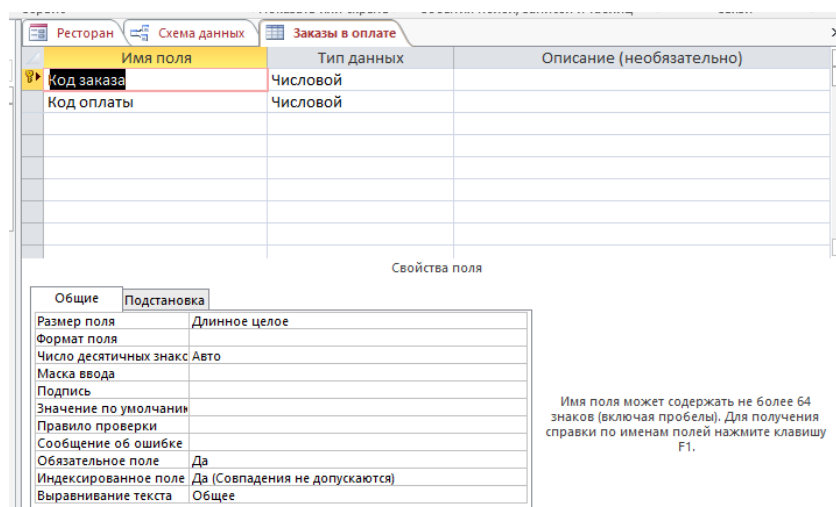


Рисунок 2.17 – Таблица «Заказы в оплате» в режиме конструктора

В таблице «Залы» созданы следующие поля и заданы типы данных (рисунок 2.18):

Поле «№ зала» имеет тип данных Числовой.

Поле «Наименование» имеет тип данных Короткий текст.

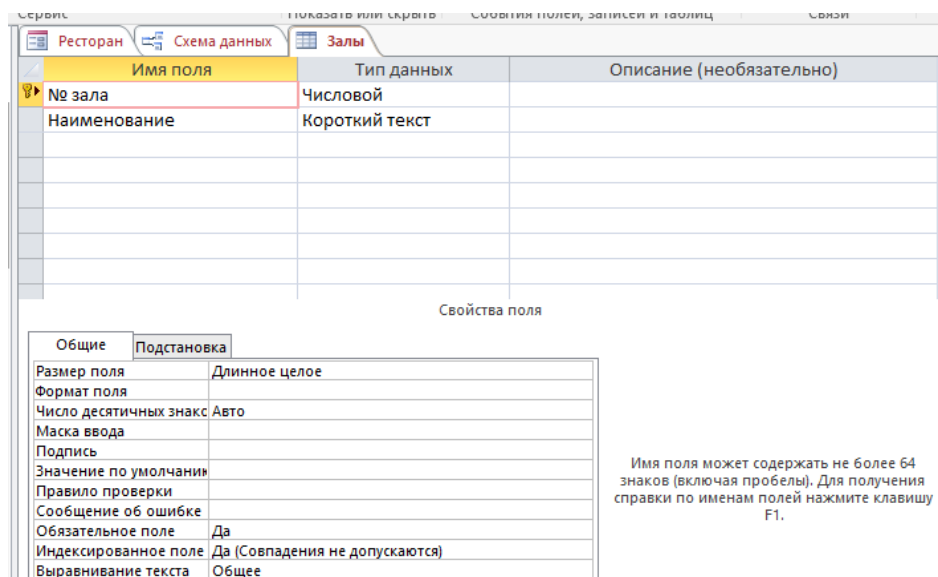


Рисунок 2.18 – Таблица «Залы» в режиме конструктора

В таблице «Категории блюд» созданы следующие поля и заданы типы данных (рисунок 2.19):

Поле «Код категории» имеет тип данных Счётчик.

Поле «Название категории» имеет тип данных Короткий текст.

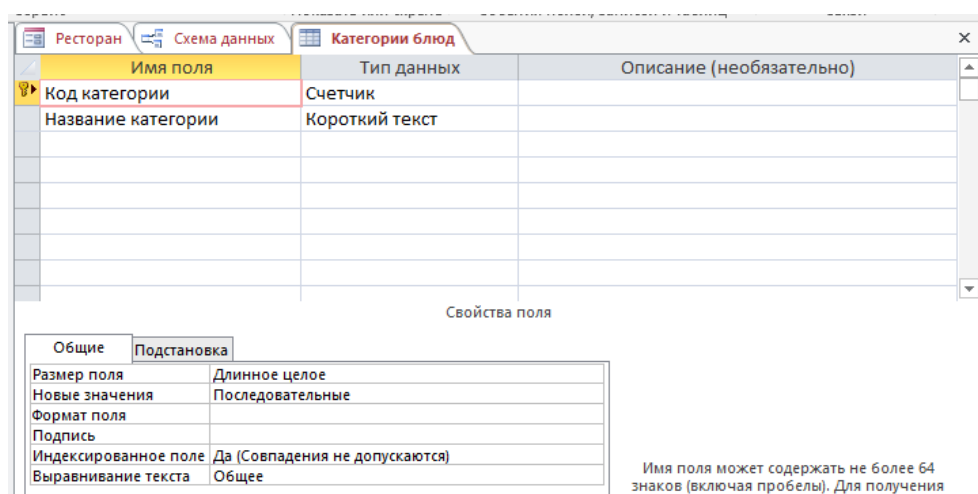


Рисунок 2.19 – Таблица «Категории блюд» в режиме конструктора

В таблице «Места» созданы следующие поля и заданы типы данных (рисунок 2.20):

Поле «Код места» имеет тип данных Счетчик.

Поле «№ места» имеет тип данных Числовой.

Поле «Зал» имеет тип данных Числовой.

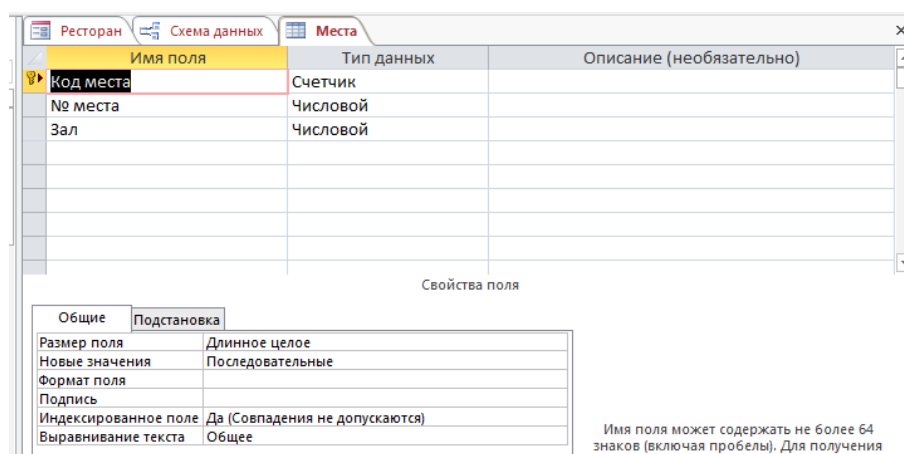


Рисунок 2.20 – Таблица «Места» в режиме конструктора

В таблице «Места в предзаказе» созданы следующие поля и заданы типы данных (рисунок 2.21):

Поле «№ пп» имеет тип данных Счетчик.

Поле «Код предзаказа» имеет тип данных Числовой.

Поле «Место» имеет тип данных Числовой.

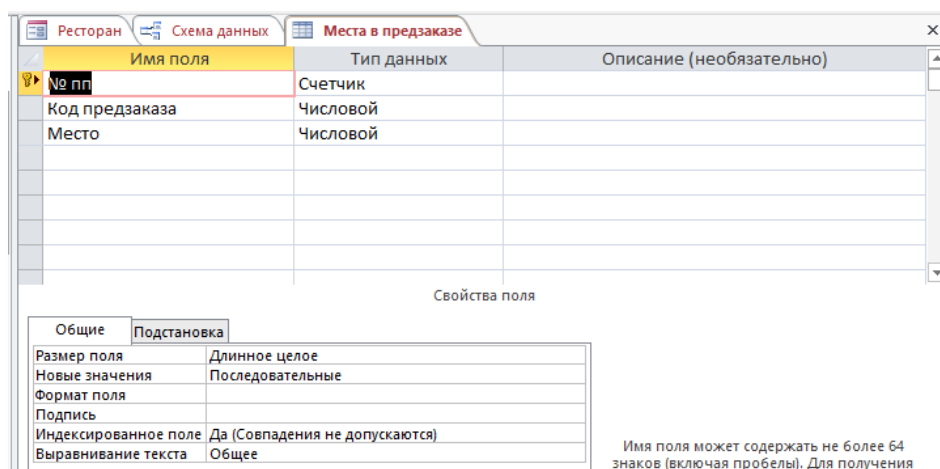


Рисунок 2.21 – Таблица «Места в предзаказе» в режиме конструктора

В таблице «Оплата заказов» созданы следующие поля и заданы типы данных (рисунок 2.22):

Поле «Код оплаты» имеет тип данных Счётчик.

Поле «Сумма заказов» имеет тип данных Числовой.

Поле «Оплачено» имеет тип данных Числовой.

Поле «Сдача» имеет тип данных Числовой.

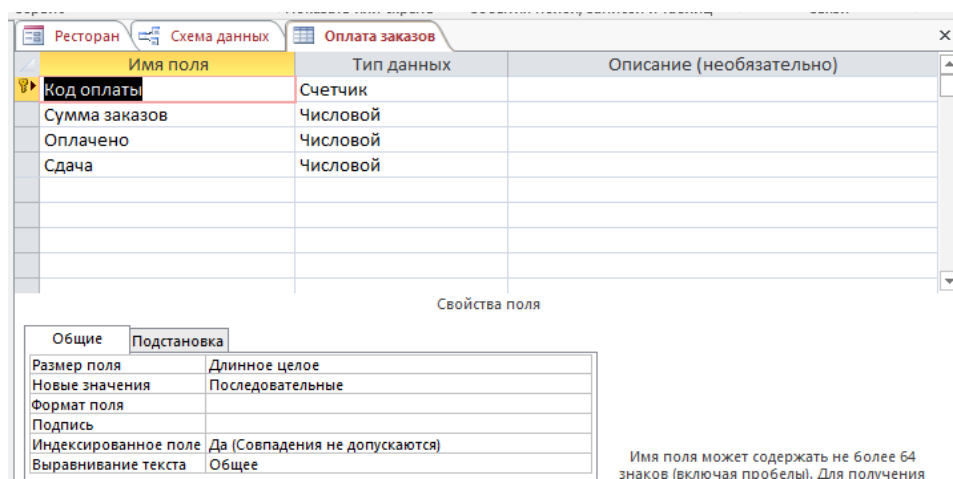


Рисунок 2.22 – Таблица «Оплата заказов» в режиме конструктора

В таблице «Предварительные заказы» созданы следующие поля и заданы типы данных (рисунок 2.23):

Поле «Код заказа» имеет тип данных Счётчик.

Поле «Дата» имеет тип данных Дата и время.

Поле «Время» имеет тип данных Дата и время.

Поле «ФИО» имеет тип данных Короткий текст.

Поле «Контактный тел» имеет тип данных Короткий текст.

Поле «Выполнен» имеет тип данных Логический.

Поле «Сумма заказов» имеет тип данных Числовой.

Поле «Сумма предоплаты» имеет тип данных Числовой.

Поле «Сумма доплаты» имеет тип данных Числовой.

Поле «Сдача» имеет тип данных Числовой.

Поле «Оплачено» имеет тип данных Логический.

Поле «Примечание» имеет тип данных Длинный текст.

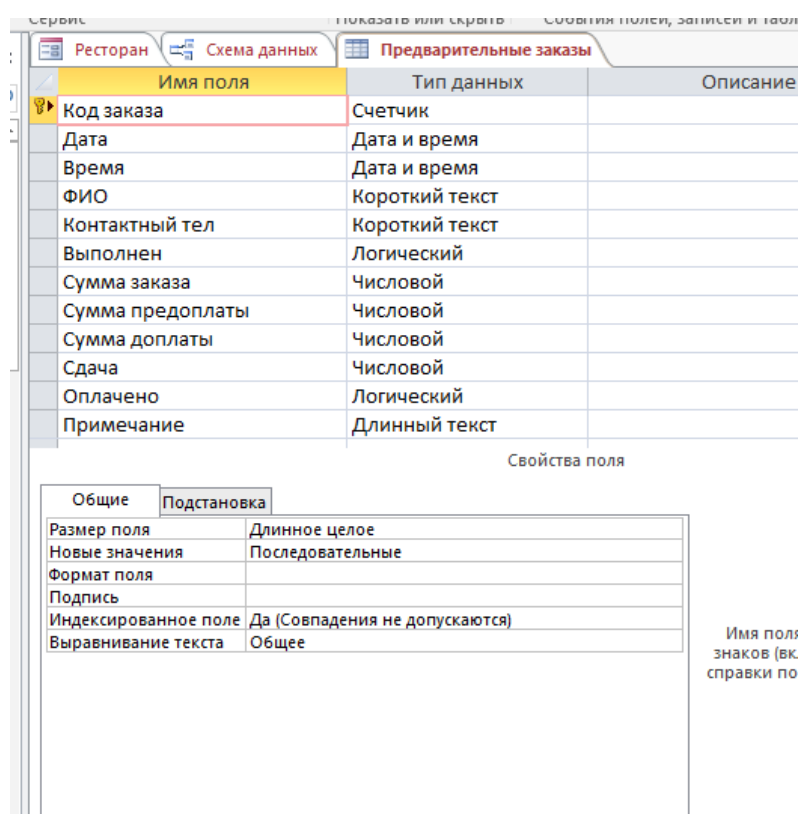


Рисунок 2.23 – Таблица «Предварительные заказы» в режиме конструктора

2.2 Описание разработанного продукта

Разработанная база данных обладает достаточно высокой функциональностью, а ее интерфейс удобен и прост в использовании.

На рисунке 2.24 отображена главная форма, обеспечивающая возможность доступа к любым необходимым объектам базы данных.

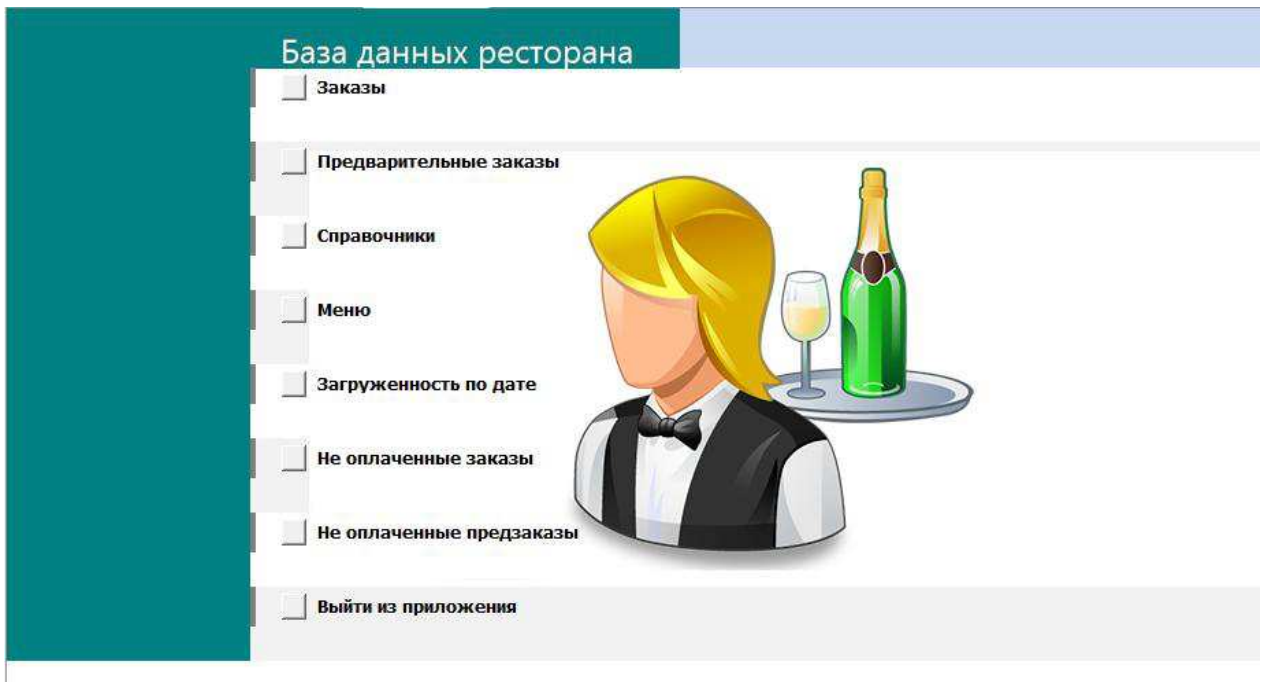


Рисунок 2.24 – Главная форма

Пользователь сможет щелчком «мыши» по кнопке, например, «Заказы» открывает окно формы для просмотра информации о заказе или редактировать данные уже введенных заказов (рисунок 2.25).

Код заказа: 9 Выполнен:

Дата: 27.05.2018 Оплачено:

Время: 10:27 Примечание: День рождения

Место: 20

Код заказа	Дата	Время	Место	Выполнен	Оплачено	Примечани
1	27.05.2018	12:17	2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
2	27.05.2018	14:00	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	27.05.2018	14:20	3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
4	27.05.2018	14:20	4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Свадьба
5	27.05.2018	15:22	4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
6	27.05.2018	12:24	5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	27.05.2018	12:24	16	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	27.05.2018	12:24	15	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
9	27.05.2018	10:27	20	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	День рождения
10	27.05.2018	11:20	12	<input type="checkbox"/>	<input type="checkbox"/>	
11	27.05.2018	11:20	10	<input type="checkbox"/>	<input type="checkbox"/>	

Рисунок 2.25 – Форма «Заказы»

Аналогичным образом пользователь может просмотреть форму «Предварительные заказы», где отображается вся информация о заказчике, так на рисунке 2.26 отображена форма в режиме просмотра имеющихся заказов.

Код заказа: 1

Дата: 27.05.2018

Время: 13:25

ФИО: Кулик Андрей Дмитриевич

Контактный тел: 256-2678-12 Оплачено:

Выполнен: Примечание: День рождения

Код заказа	Дата	Время	ФИО	Контактный	Выполнен	Оплачено	Примечани
1	27.05.2018	13:25	Кулик Андрей	256-2678-12	<input type="checkbox"/>	<input type="checkbox"/>	День рождения
2	27.05.2018	15:00	Макаров Олег	89-89-89	<input type="checkbox"/>	<input type="checkbox"/>	Встреча выпуск
3	27.05.2018	19:00	Лариса Михай	22-22-22	<input type="checkbox"/>	<input type="checkbox"/>	Деловой ужин
4	27.05.2018	21:00	Максим	28-28-28	<input type="checkbox"/>	<input type="checkbox"/>	Романтически
5	27.05.2018	23:00	Влад Михайлс	25-23-12	<input type="checkbox"/>	<input checked="" type="checkbox"/>	День рождения
* (№)	20.06.2018				<input type="checkbox"/>	<input type="checkbox"/>	

Рисунок 2.26 – Форма «Предварительные заказы»

Работа по оформлению, редактированию заказов производится с помощью формы «Блюда в заказе» (рисунок 2.27). Отображается список имеющихся блюд и предоставляет выбор количества блюд.

Администратор ресторана может контролировать оплату заказов (рисунок 2.28), а также может сформировать отчет по оплаченному заказу (рисунок 2.29).

Код заказа	Код блюда	Количество
1	Сельдь с печёной картошкой	2
1	Ассорти рыбное	2
1	Нарезка мясная	5
1	Солохин огород	5
1	Цезарь	4
1	Сок яблочный	3
1	Сок апельсиновый	3

- Сельдь с печёной картошкой
- Ассорти рыбное
- Нарезка мясная
- Солохин огород
- Прованс
- Греческий
- Цезарь
- Сок яблочный
- Сок апельсиновый
- Сок ананасовый

Рисунок 2.27 – Форма «Блюда в заказе»

Код заказа	1	Подсчитать сумму
Сумма заказа	3 202,00р.	
Сумма предоплаты	1 000,00р.	
Необходимо доплатить	2 202,00р.	Подсчитать сдачу
Сумма доплаты	2 500,00р.	
Сдача	298,00р.	Распечатать
		Закорить

Рисунок 2.28 – Оплата предварительного заказа

Заказы	Предварительные заказы	Оплата заказов подтв	Счёт по предзаказу
Счёт по предзаказу			
Код заказа	1		
Дата	27.05.2018		
Время	13:25		
ФИО	Кулик Андрей Дмитриевич		
Сумма заказа	3 202,00р.		
Сумма предоплаты	1 000,00р.		
Сумма доплаты	2 500,00р.		
Сдача	298,00р.		

Рисунок 2.29 –Сформированный отчет по оплате

Администратор базы данных сможет просмотреть список заказов по дате, с помощью отчета «Загруженность по дате», для выбора нужно ввести дату, просмотр отчета на рисунке 2.30.

Загруженность по дате		27.05.2018			
Время	Зал	Заказано мест	Осталось мест	Примечание	Вид заказа
10:27:00	Банкетный	1	19	День рождения	заказ
11:20:00	Караоке	2	13		заказ
12:17:00	Банкетный	1	19		заказ
12:24:00	Банкетный	2	18		заказ
12:24:00	Летняя терраса	1	7		заказ
14:00:00	Банкетный	1	19		заказ
14:20:00	Основной зал	1	9		заказ
14:20:00	Основной зал	1	9	Свадьба	заказ
15:22:00	Летняя терраса	1	7		заказ
13:25:00	Банкетный	5	15	День рождения	предварительный заказ
15:00:00	Банкетный	8	12	Встреча выпускников	предварительный заказ
19:00:00	Караоке	4	11	Деловой ужин	предварительный заказ
21:00:00	Основной зал	2	8	Романтический ужин	предварительный заказ
23:00:00	Банкетный	3	17	День рождения	предварительный заказ

Рисунок 2.30 –Сформированный отчет по дате

В режиме SQL данный запрос формулируется следующим образом select *, "заказ" from [Заказы по дате]

UNION ALL select *, "предварительный заказ" from [Предзаказы по дате];

На рисунке 2.31 изображен отчет «Меню» где мне можно ознакомиться с блюдами ресторана, ценами и количеством.

Код блюда	Название	Количество	Цена
7	Цезарь	200	120,00р.
6	Греческий	200	135,00р.
5	Прованс	150	140,00р.

Код блюда	Название	Количество	Цена
4	Солохин огород	400	80,00р.
3	Нарезка мясная	200	270,00р.
2	Ассорти рыбное	200	300,00р.
1	Сельдь с печёной картошкой	200	90,00р.

Код блюда	Название	Количество	Цена
10	Сок ананасовый	200	32,00р.
9	Сок апельсиновый	200	32,00р.
8	Сок яблочный	200	32,00р.

20 июня 2018 г. Стр. 1 из 1

Рисунок 2.31 – Отчет по меню ресторана

Для дальнейшего развития разработанной базы данных администратор базы данных может выполнить экспорт базы данных в список SharePoint (Рисунок 2.32), далее создать учетную запись и выполнить публикация веб-приложения на сайте SharePoint, которые предоставляет Office 365 в "облаке".

Тогда администратор ресторана сможет получить удаленный доступ к базе данных посредством интернет-браузера.

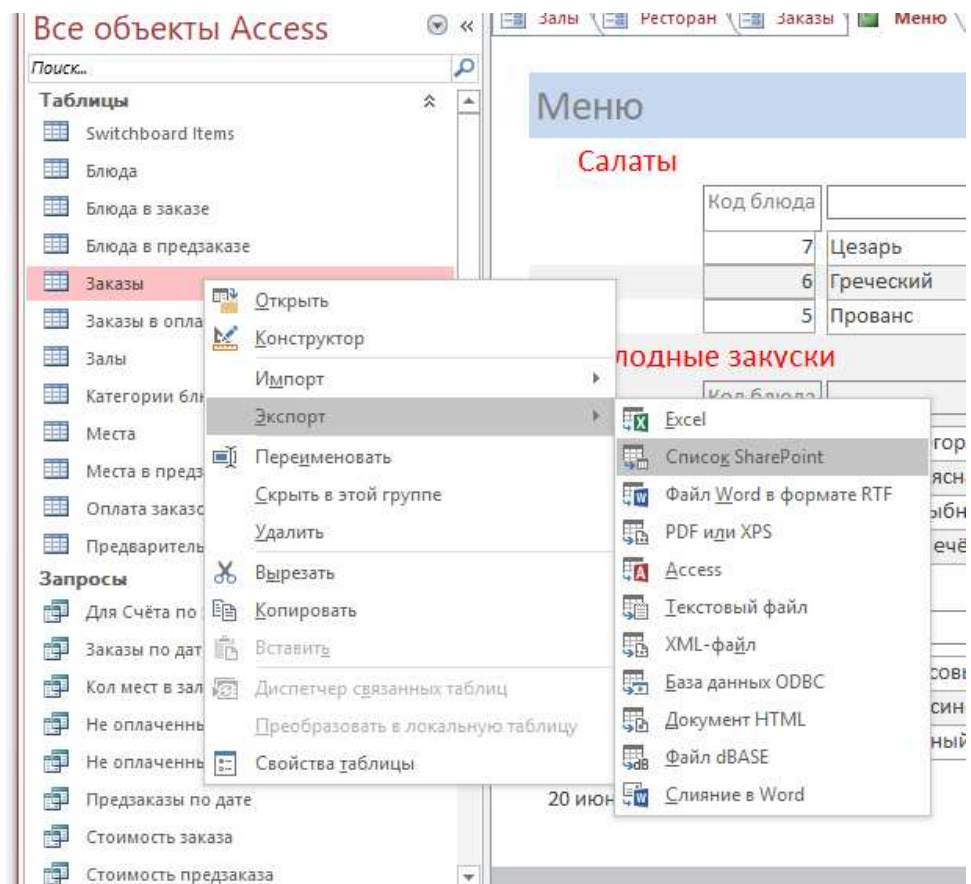


Рисунок 2.32 – Экспорт базы данных в список SharePoint

ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы были решены все поставленные задачи:

- Выполнен анализ методологии проектирования баз данных;
- Произведен анализ организации «Инвест Мастер»;
- Выполнено обоснование выбора СУБД;
- Осуществлен выбор технологии разработки;
- выполнено проектирование базы данных для автоматизации ресторана ООО «Инвест Мастер»;
- внесены предложения по дальнейшему развитию ресторана.

В ходе анализ методологии проектирования баз данных было выявлено, что проектирование баз данных является длительным и трудоемким процессом, занимающим сотни человеко-часов и большое число циклов улучшений и доработок. При этом, в качестве основных ресурсов проектировщика баз данных выступают его собственная интуиция и опыт, что достаточно часто может негативно сказаться на качестве проектируемых решений.

Можно выделить основные причины низкой эффективности проектируемых баз данных:

- недостаточно глубокая степень анализа требований на начальных этапах проектирования, включая их семантический аспект и взаимосвязи данных;
- длительный и утомительный процесс структурирования данных, трудно выполняемый при ручной обработке информации.

Это обуславливает необходимость автоматизации разработки баз данных на разных этапах проектирования.

В практической части работы была произведена проектирование базы данных для ресторана ООО «Инвест Мастер».

Применение разработанной базы данных позволяет:

- повысить эффективность обработки заявок клиентов;

- уменьшить время, необходимое для обработки и движения документов по подразделениям;
- повысить производительность труда за счет сокращения времени обработки заказов;
- повысить оперативность обновления данных.

Это исключает ручные расчеты сотрудников, обеспечивает обмен данными между ресторанами компании, повышает оперативность и общую эффективность работы организации.

Были внесены предложения по дальнейшему развитию ресторана, предлагается публикация веб-приложения на сайте SharePoint, которые предоставляет Office 365 в "облаке".

Получаемым эффектом будет улучшение корпоративного взаимодействия, повышение производительности труда сотрудников, уменьшение количества ошибок.

СПИСОК СОКРАЩЕНИЙ

ODBC – Open Database Connectivity

БД – база данных

SQL – язык структурированных запросов

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Алоев Т.Б., Батова И.А., Бесланеев Р.А. Организация эффективного управления предприятием на основе информационных технологий // ИННОВАЦИИ И ИНВЕСТИЦИИ. М.: Изд. ООО Журнал Инновации и инвестиции, 2016. – С.62-66.
2. Бекаревич Ю., Microsoft Access 2013 / СПб.: БХВ-Петербург, 2014. – 465 с.
3. Бекаревич Ю., Пушкина Н. MS Office Access 2016, СПб.: БХВ-Петербург, 2017. –480 с.
4. Губина Е.А., Ирзаев Г.Х., Адеева М.Г. Проектирование информационной системы на основе связывания CASE-инструментария и реляционной базы данных // НАУКА И БИЗНЕС: ПУТИ РАЗВИТИЯ. Тамбов: Изд. Фонд развития науки и культуры, 2014. – С.75-79.
5. Гущин А. Н. Базы данных. Учебник / М.: Директ-Медиа, 2014.– 266 с.
6. Интернет-торговля в 2016 году [Электронный ресурс] // 2017. – Режим доступа: <http://www.retailer.ru/item/id/141702/>
7. Интернет-торговля в России: Главные новости, события, прогнозы [Электронный ресурс] // 2017. – Режим доступа: <http://www.bizhit.ru>
8. Карпова И.П. Базы данных: Учебное пособие / СПб.: Питер, 2013. – 240 с.
9. Лаврентьев К.А., Титова Е.А. Проблемы проектирования архитектуры распределенных баз данных // ВЕСТНИК ХАБАРОВСКОЙ ГОСУДАРСТВЕННОЙ АКАДЕМИИ ЭКОНОМИКИ И ПРАВА. Хабаровск, 2015. – С.33-38.
10. Карпова И.П. Базы данных: Учебное пособие / СПб.: Питер, 2013. - 240 с.
11. Карпова Татьяна. Базы данных: модели, разработка, реализация [Электронный ресурс] // Сайт, 2015. – Режим доступа: <http://www.intuit.ru>
12. Салбер Алена. Перспективы развития рынка электронной коммерции [Электронный ресурс] // 2015. – Режим доступа: <http://www.internetsales.ru/perspektivi-rzvitiya-rinka/>

- 13.Советов Б., Цехановский В., Чертовской В. Базы данных. Учебник / М: Издательство «Юрайт», 2015. – 464 с.
- 14.Уидом Дженнифер. Реляционные базы данных. Руководство / М:Лори, 2014. – 374 с.
- 15.Филипьев Н.В., Рекунков И.Ю., Шуляк Е.А. Особенности выбора оптимальных программных средств для разработки информационных систем // КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ В НАУКЕ, ПРОИЗВОДСТВЕ, СОЦИАЛЬНЫХ И ЭКОНОМИЧЕСКИХ ПРОЦЕССАХ. Сборник научных статей по материалам 15-ой Международной науч.-практ. конф. Новочеркасск: Изд. ЮРГПУ (НПИ), 2015. – С.90-92.
- 16.Швецов Владимир. Базы данных [Электронный ресурс] // Нижегородский государственный университет им. Н.И.Лобачевского, 2015. – Режим доступа: <http://www.intuit.ru>
17. SharePoint 2010 [Электронный ресурс] / 2017. – Режим доступа: http://sharepoint.mentoors.com/?category_name=sharepoint-book
- 18.Йордан, Э., Аргила, К. Структурные модели в объектно- ориентированном анализе и проектировании. – М.: ЛОРИ, 2014. – 264 с
- 19.Коннолли, Т., Бегг К., Страчан А. Базы данных: проектирование, реализация, сопровождение. Теория и практика, 2-е изд. : Пер. с англ. : Уч. пос. – М.: Изд. дом "Вильямс", 2013. – 1120 с.
- 20.Тиори, Т., Фрай, Дж., Проектирование структур баз данных : В 2-х кн.: Пер. с англ. – М.: Мир, 2013.

ПРИЛОЖЕНИЕ А (обязательное)

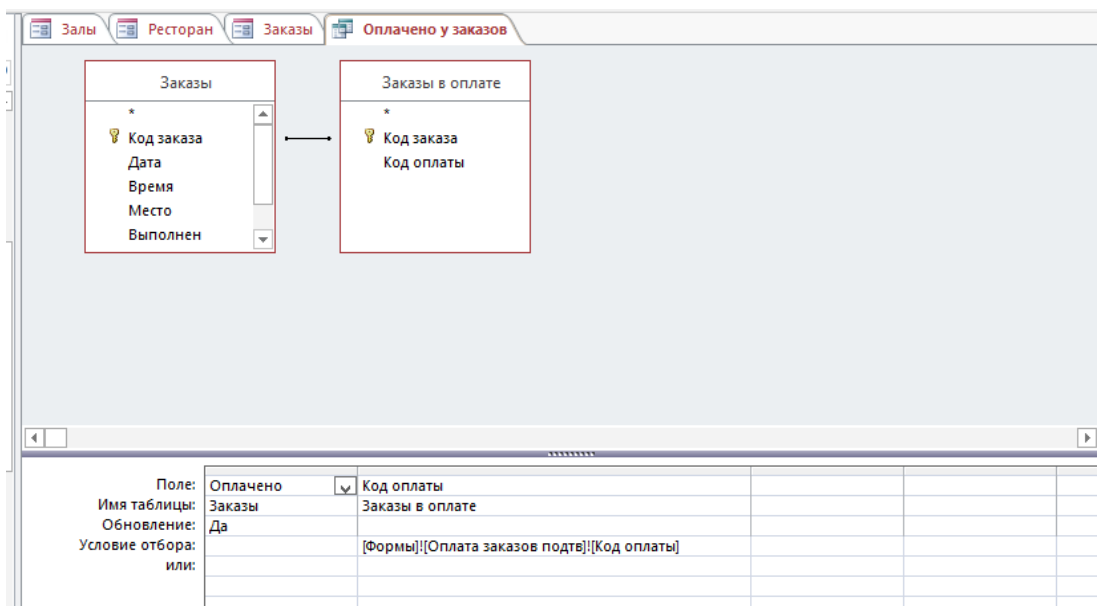


Рисунок А.1 – Запрос Access о проверке оплаты

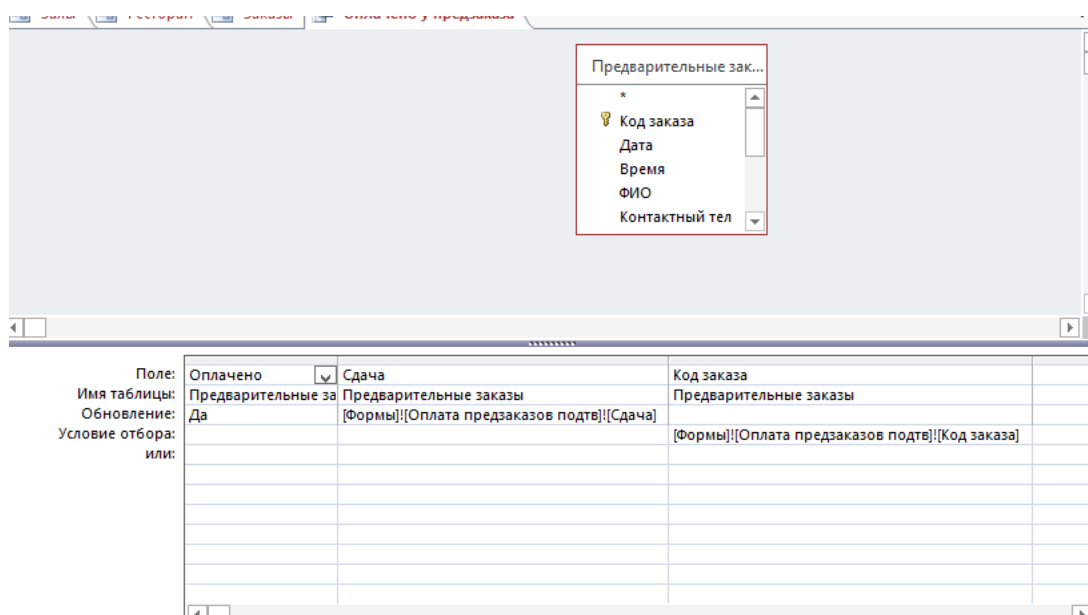


Рисунок А.2 – Запрос Access о сумме оплаты

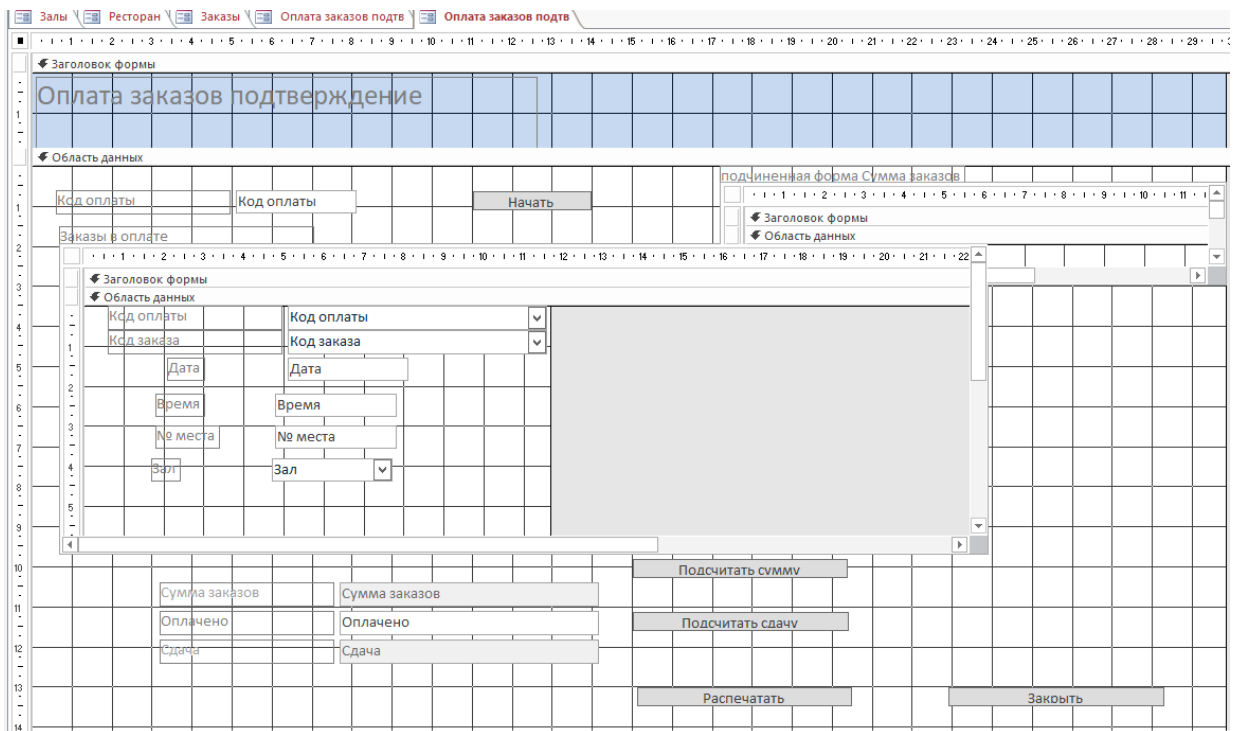


Рисунок А.3 – Форма «Предварительные заказы» в режиме конструктора

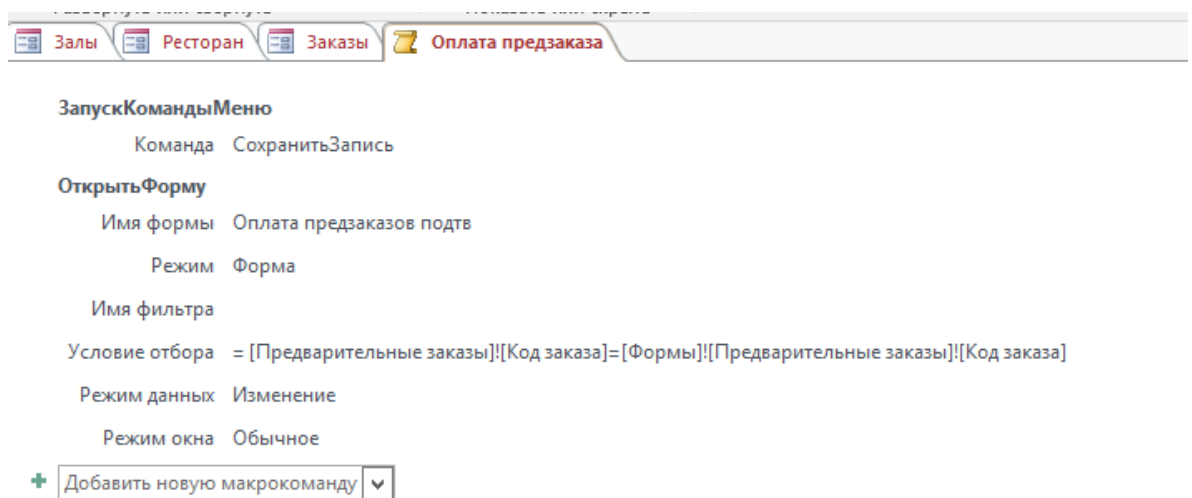



Рисунок А.4 – Макрос «Оплата предзаказа» в режиме конструктора

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ПЕДАГОГИКИ, ПСИХОЛОГИИ И СОЦИОЛОГИИ
КАФЕДРА СОВРЕМЕННЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ

УТВЕРЖДАЮ

Заведующий кафедрой

 И. А. Ковалевич

подпись инициалы, фамилия

« 19 »  2018 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.03.19 – Прикладная информатика в социальных коммуникациях

код – наименование направления

Создание базы данных для ресторана

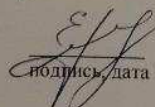
Руководитель


подпись, дата

к. пер. и. доц. каф. СОТ
должность, ученая степень

М.М. Манушкина
инициалы, фамилия

Выпускник


подпись, дата

Егоян Д.З.
инициалы, фамилия

Красноярск 2018