

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**
(**Н И У « Б е л Г У »**)

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК
КАФЕДРА МАТЕМАТИЧЕСКОГО И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
ИНФОРМАЦИОННЫХ СИСТЕМ

РАЗРАБОТКА WEB-ПРИЛОЖЕНИЯ УЧЕТА ЗАКАЗОВ ДЛЯ ООО «ВРК»

Выпускная квалификационная работа
обучающегося по направлению подготовки 02.03.02
Фундаментальная информатика и информационные технологии
заочной формы обучения, группы 07001350
Харитоновой Юлии Алексеевны

Научный руководитель
к.т.н., доцент
Муромцев В.В.

БЕЛГОРОД 2018

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1 ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДЕЯТЕЛЬНОСТИ ООО ВРК И ПУТИ ЕЕ СОВЕРШЕНСТВОВАНИЯ.....	6
1.1 Способы совершенствования информационного обеспечения ООО «ВРК»	6
1.2 Современное состояние информационной деятельности ООО «ВРК»	16
1.3 Описание целей, функций и задач предприятия.....	18
2 ПРОГРАММИРОВАНИЕ ВЕБ ПРИЛОЖЕНИЯ ДЛЯ ИНФОРМАЦИОННОЙ ПОДДЕРЖКИ ООО ВРК.....	22
2.1 Выбор средств разработки	22
2.2 Требования к СУБД	25
2.3 Проектирование архитектуры веб приложения.....	26
2.4 Проектирование структуры базы данных, датологическое программирование	29
2.5 Физическое программирование	32
3 ТЕСТИРОВАНИЕ И АПРОБАЦИЯ.....	35
3.1 Тестирование модулей веб приложения.....	35
3.2 Опытная эксплуатация, развертывание.	45
3.3 Анализ соответствия реализации заявленной спецификации.	47
ЗАКЛЮЧЕНИЕ	49
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ	50
ПРИЛОЖЕНИЯ.....	51

ВВЕДЕНИЕ

Представленная работа посвящена разработке Web-приложения учета заказов для ООО «ВРК» с использованием JavaScript, PHP и СУБД MySQL.

В настоящее время, в связи с глобальным развитием сети Интернет, в программировании все более резко выделяется отдельная его отрасль - web-программирование. Изначально, оно не могло даже сравниться по своей сложности с другими областями «программистского ремесла», не дотягиваясь не только до системного, но даже и до прикладного программирования. Речь идет, конечно, о программировании сценариев для интернет сайтов, или, Web-программировании.

Во время стремительного прогресса просто красиво оформленный текст и картинки на веб-сайте уже никого не удивят. Требования к сайтам, изменились - теперь для успешного представительства компании в Интернет необходимо предоставить своим посетителям различные возможности: обратную связь, форум, голосование, интернет-магазин, различные web-тесты, поиск по сайту, счётчик посещений и многое другое. При помощи обычного html этого не достичь, ведь html - это язык гипертекстовой разметки - инструмент для создания гиперссылок, вставки изображений, таблиц и др [4].

В современном мире очень развита конкурентная среда различных видов ресторанного бизнеса и необходимости и пользе представления кафе и ресторанов в интернете. Думаю, что любой бизнесмен заботится о том, чтобы его дело процветало, что бы количество клиентов непрерывно росло. Касается это и владельцев кафе и ресторанов. Наверняка хороший владелец ресторана просто так бьётся головой за каждого клиента, ведь каждый новый клиент – это в перспективе постоянный посетитель заведения и, соответственно постоянный источник дохода.

Для целей компании ООО «ВРК», необходимо создать web-приложение, которое должно решить ряд задач. Главные из которых, повышение продаж и более тесное общение с клиентами.

Логика приложения должна быть проста, приложение должно иметь простой и понятный интерфейс. Если однажды клиент не сможет сделать заказ через ваше приложение, скорее всего он уйдет и никогда не вернется в него.

Приложение должно быть удобно не только для пользователей, но и для сотрудников, которые обрабатывают заказы. Ведь если заказы не будут исполняться, либо исполняться не точно, никто не захочет пользоваться приложением. Это значительно упростит работу ресторанов компании и в конечном итоге увеличит прибыль ресторана.

Создание сайта представляет собой создание информационного ресурса, который предоставит возможность различным компаниям привлекать новых клиентов.

Цель данной работы – разработать web-приложение учета заказов для ООО «ВРК».

В рамках достижения поставленной цели были поставлены следующие задачи:

- проанализировать состояние информационной деятельности ООО «ВРК» и сформулированы требования к разрабатываемому ПО;
- изучить теоретические основы инструментальных средств разработки Web-приложения;
- выполнить проектирование web-приложения с использованием инструментальных средств;
- разработать web-приложение;
- выполнить тестирование web-приложения.

Работа включает в себя введение, теоретическая часть, исследовательская часть, тестированная часть, заключение, библиографический список.

Во введении обоснована актуальность темы исследования, определены цель, задачи, предмет и объект исследования.

Глава первая теоретическая, где раскрываются теоретические основы разработки веб-приложений и цели предприятия ООО «ВРК».

Глава вторая представляет собой исследовательскую часть. Где рассматриваются этапы проектирования web-приложения. Строятся физическая и логическая модели БД, описывается назначение модулей приложения, формируется модульная схема приложения.

Глава третья имеет исследовательский характер. Где рассматриваются этапы разработки и тестирования web-приложения.

В заключении подводятся итоги всей курсовой работы.

С целью подробного изложения данной темы мною, были использованы интернет сайты, научная и периодическая литература различных авторов, публикующие свои статьи в книгах, журналах, учебных пособиях, отражающая суть исследования данной проблемы на современном уровне.

1. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДЕЯТЕЛЬНОСТИ ООО ВРК И ПУТИ ЕЕ СОВЕРШЕНСТВОВАНИЯ

1.1. Способы совершенствования информационного обеспечения ООО «ВРК»

Важным элементом повышения эффективности производства в любой отрасли является улучшение информационного обеспечения. Совершенствование форм и методов информационного управления происходит на основе достижений научно-технического прогресса, дальнейшего развития информатики, занимающейся изучением законов, методов и способов накопления, обработки и передачи информации с помощью различных технических средств.

Различные информационно-технические новшества следует воспринимать как средство сокращения и удешевления аппарата управления. Так, например, появление телефона, радио, телевидения, персональных компьютеров, локальных компьютерных сетей и глобальной сети Интернет приводило в свою очередь к совершенствованию системы информационного обеспечения управления предприятием. В конечном итоге роль информации в организационном управлении фирмой постоянно возрастает, что связано с изменениями социально-экономического характера, появлением новейших достижений в области техники и технологий, результатами научных исследований [8].

С помощью совершенствования информационного обеспечения достигаются следующие задачи:

1. Экономия расходов за счёт снижения фонда заработной платы; коммунальных услуг; стоимости программного обеспечения; расходов на оформление договоров; расходов на перераспределение сырья и пр.

2. Устранение возможных расходов в будущем, а именно: избежание будущего роста численности персонала; уменьшение требований к обработке данных; снижение стоимости обслуживания и пр.

3. Приобретение нематериальной выгоды: улучшение качества информации; повышение производительности; улучшение и ускорение обслуживания; новые производственные мощности; улучшение контроля; уменьшение просроченных платежей; полное использование программного обеспечения.

Понятие информации является достаточно ёмким и широко распространено в настоящее время. Сам термин информация происходит от латинского слова *information* – разъяснение, осведомление, изложение.

Процесс передачи и получения информации представлен на рис. 1.1. Это простая схема передачи информации в одном направлении [7].

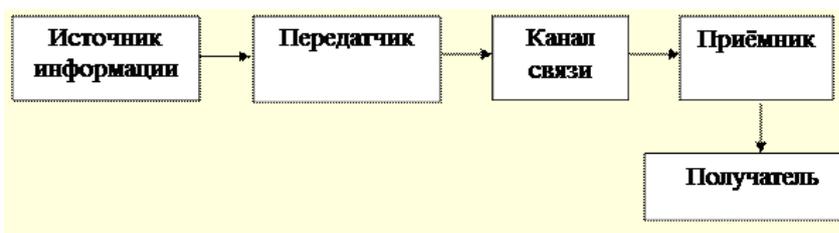


Рис. 1.1 Простая схема передачи информации в одном направлении

В процессе управления постоянно происходит обмен информацией. Причём направление перемещения информации может быть вертикальным (от руководителя к подчинённым или от подчинённых к руководителю), так и горизонтальным (между начальниками подразделений, подчинёнными одного уровня).

Управление в полной мере использует объективную и своевременную информацию, собираемую, обрабатываемую, сохраняемую и распространяемую с помощью современных научных методов и технических средств. Нужно не только располагать своевременной и точной информацией,

но уметь осмысливать ее, делать необходимые выводы и результативно воплощать в управленческих решениях.

Информацию можно трактовать как совокупность сведений, сообщений, материалов, данных, определяющих меру потенциальных знаний менеджера о процессах или явлениях в их взаимосвязи.

Суть информации составляют только те данные, которые уменьшают неопределенность интересующих менеджера событий. Информация в менеджменте — сумма нужных, воспринятых и осознанных сведений, необходимых для анализа конкретной ситуации, дающая возможность комплексной оценки причин ее возникновения и развития, позволяющая определить ряд альтернативных решений, из которых реально (исходя из конкретной ситуации) найти оптимальное управленческое решение, осуществить контроль за его выполнением [7].

Состав, содержание и качество информации, которая привлекается к руководителю, имеют определяющую роль в обеспечении действенности управления. Анализ информации не ограничивается только экономическими данными, а широко использует техническую, технологическую и другую информацию. Все источники данных делятся на плановые, учетные и внеучетные.

По отношению к объекту исследования информация бывает внутренней и внешней. Система внутренней информации - это данные статистического бухгалтерского, оперативного учета и отчетности, плановые данные, нормативные данные, разработанные на предприятии и т.д. Система внешней информации - это данные статистических сборников, периодических и специальных изданий, конференций, деловых встреч, официальные, хозяйственно-правовые документы и т.д.

По отношению к предмету исследования информация делится на основную и вспомогательную, необходимую для более полной характеристики изучаемой предметной области.

По периодичности поступления аналитическая информация подразделяется на регулярную и эпизодическую. К источникам регулярной информации относятся плановые и учетные данные. Эпизодическая информация формируется по мере необходимости, например сведения о новом конкуренте.

Важную роль в использовании информации играют способы ее регистрации, обработки, накопления и передачи; систематизированное хранение и выдача информации в требуемой форме; производство новой числовой, графической и иной информации. Другими словами необходимо рассмотреть технологию информационной деятельности.

Информатизация в современных условиях базируется в основном на использовании вычислительной техники. Этот процесс включает в себя несколько стадий:

1. Формирование соответствующей технико-технологической базы (с применением новейших информационных технологий и современных электронно-вычислительных и коммуникационных средств);
2. Организацию производства и сферы распределения информационно-вычислительных услуг;
3. Отработку механизма информационного обслуживания и создание действенной системы управления данными процессами;
4. Внедрение и эксплуатацию прогрессивных форм, методов и средств проведения информационной деятельности в процессах управления, а также создание для этого необходимых материально-технических, организационно-экономических и социальных условий.

Эффективное применение ЭВМ зависит от следующих условий:

- создания самих вычислительных машин;
- создания для них программного обеспечения;
- подготовленности среды применения.

Во внутрифирменной системе информации используются следующие виды вычислительной техники:

- ЭВМ крупногабаритные и персональные;
- микропроцессоры;
- средства телекоммуникаций;
- электронные пишущие машинки, композеры, терминальные устройства со встроенной микроЭВМ;
- средства автоматизированной обработки текстовой информации.

ЭВМ используются, прежде всего, для обработки данных и решения расчетных задач. В современных условиях ЭВМ стали все чаще применять для обработки нечисловой информации и термин "вычислительная техника" перестал соответствовать характеру задач, решаемых с помощью компьютера.

Развитие систем телекоммуникаций позволило объединить все технические средства обработки цифровой и текстовой информации в единую внутрифирменную систему информации.

Наиболее эффективной считается система информации, основанная на одновременном использовании вычислительной техники и средств автоматизированной обработки текстовой информации.

Комплексная автоматизированная обработка информации предполагает объединение в единый комплекс всех технических средств обработки информации с использованием новейшей технологии, методологии и различных процедур по обработке информации. Создание комплексной автоматизированной системы включает последовательные этапы:

- автоматизация процессов сбора, хранения и выдачи данных;
- использование устройств для автоматизированной обработки текстовой информации и фотонаборных устройств;
- интеграция обособленных процессов обработки информации в единую внутрифирменную систему;
- объединение технических средств цифровой и текстовой информации посредством электронной почты;
- использование всего комплекса технических средств обработки информации, переход к единой системе обработки всех видов информации [7].

Об автоматизации деятельности по управлению персоналом в России начали говорить недавно. Если бухгалтерские пакеты созданы давно и активно применяются практически во всех структурах, то над автоматизацией труда кадровых работников задумывались мало, а спрос на такие системы в последнее время значительно вырос. Автоматизация деятельности по управлению персоналом дает:

- оперативность контроля возможность получения непротиворечивых и полных данных о структуре предприятия, позициях штатного расписания и сотрудниках;
- жесткий контроль за деятельностью дочерних предприятий;
- настраиваемую систему разработки отчетной документации;
- переход на новые технологии и методы работы;
- улучшение условий труда;
- гибкость управления.

На качество управления предприятием влияют такие факторы, как оперативность и качество формирования документов, приема-передачи информации, согласованность работы справочно-информационной службы, четкая организация хранения, поиска и использования документов.

Когда на смену печатной машинке пришел компьютер, а переписка стала возможной без использования конвертов и почтовых марок, совершенно очевидным стал тот факт, что сегодня при работе с информацией не обойтись без использования новейших информационных технологий.

На современном этапе развития рыночной экономики полноценная аналитическая обработка учетной и иной экономической информации немислима без применения средств вычислительной техники. Качественное информационное обеспечение процесса управления хозяйственной деятельностью возможно только при использовании в полной мере всех достижений научно-технического прогресса. В первую очередь это касается применения в практике управления новейших информационных технологий:

средств вычислительной техники, телекоммуникаций и программного обеспечения.

Вместе с тем, в большинстве работ, посвященных вопросам автоматизации системы управления деятельностью хозяйствующих субъектов, значительно меньше внимания уделяется проблемам разработки и проектирования аналитических программных продуктов. Кроме того, прогресс в области информационных технологий не ослабевает с течением времени. Революционные преобразования, произошедшие в последние годы в сфере информационных технологий и приведшие к массовому внедрению в практику управления персональных компьютеров и связанных с ними интерактивных технологий, распределенной обработки данных, основанных на диалоге клиент/сервер, требуют дальнейшего развития теоретических и методологических концепций построения информационных систем.

Таким образом, представляется обоснованным вывод, что на сегодняшний день в экономической науке создана достаточно целостная, научно обоснованная концепция организации бухгалтерского и управленческого учета в компьютерной среде. Эта концепция основана на обобщении накопленного опыта построения программных систем с позиций современных информационных технологий и современных подходов к организации учетной работы на предприятиях.

При выборе и внедрении системы необходимо решить технические, административные и организационные задачи. Информационная система есть плод совместных усилий поставщика и персонала компании. Для успешной реализации проекта:

1. Необходимо провести обследование и проектирование будущей системы. Другими словами, необходим точный анализ текущей ситуации и модель будущей системы;

2. Разработчик должен произвести настройку системы. При внедрении требуется этап тестирования, с целью выявления потенциальных ошибок;

3. Внедрение разбивается на две крупные части: подготовка администраторов из персонала предприятия и непосредственно запуск подсистем в отделах и подразделениях.

При выборе программно-аппаратных платформ и отдельных бизнес приложений должны применяться непротиворечивые, согласующиеся технологии, соблюдаться единая технология эксплуатации и обслуживания системы. Помимо этих ключевых требований, есть еще целый ряд общих технических требований для любой информационной системы:

- быстроедействие, то есть достаточно малое время реакции системы (единицы секунд) при вводе, поиске и обработке информации;
- надежная защита от несанкционированного доступа к данным и регистрация действий персонала;
- удобный пользовательский интерфейс рабочих мест;
- возможность масштабирования и развития системы;
- интеграция с модулями, используемыми в системе передачи данных;
- возможность проведения конвертации данных из использовавшихся в прошлом приложений в новую систему;
- высокая надежность работы.

Методика создания корпоративных информационных систем содержит ряд следующих общих положений:

- Технология построения системы по моделям "как надо", без попыток программирования действующих сейчас алгоритмов, так как использование в работе программных приложений - это не просто сокращение бумажных документов и рутинных операций, но и переход на новые формы ведения документооборота, учета и отчетности;
- Технология построения систем с подходом "сверху вниз". Если решение об автоматизации принято и одобрено высшим руководством, то внедрение программных модулей осуществляется с головных предприятий и подразделений, а процесс построения корпоративной системы проходит

гораздо быстрее и эффективнее, чем при внедрении системы первоначально в низовые подразделения;

- Технология поэтапного внедрения. Первыми объектами автоматизации становятся те участки, на которых в первую очередь необходимо наладить процесс учета и формирования отчетных документов для вышестоящих органов и смежных подразделений;

- Привлечение к разработке будущих пользователей. Система должна поддерживать такую схему взаимодействия между модулями и автоматизированными рабочими местами, которая отвечала бы требованиям и техническим возможностям пользователя.

Трудности и сложности применения информационной системы для малого бизнеса заключаются в большом многообразии предприятий, в разных формах организации производства, в широком ассортименте выпускаемой продукции. Все это обуславливает создание информационной системы для малого бизнеса, по объему и функциональным возможностям соизмеримых с информационной системой крупных корпораций, поэтому, с одной стороны, можно проектировать информационную систему для отдельных компаний, учитывая их специфику. Но разработка индивидуальной информационной системы для каждого предприятия экономически невыгодна. Особенно это касается малых предприятий, когда практически каждое из них несет особенность организации процессов, позволяющих данному быть конкурентоспособным [4].

На предприятии ООО «ВРК» используются веб-приложения. Основные функциональные возможности веб-приложений:

- сокращение времени и ошибок при обслуживании посетителей;
- взаимодействие в единой базе систем без необходимости специального обмена данными;
- получение данных бухгалтерского и налогового учета непосредственно из оперативных документов;

- возможность оперативного получения и анализа информации (прибыльности подразделений/номенклатуры, оборачиваемости запасов, динамики цен, объемов продаж и т.д.) для принятия управленческих решений;
- специализированный интерфейс - набор экранных форм, учитывающих специфику работы сенсорных экранов POS-систем для быстрого формирования первичных документов: заказов покупателей, выпуска продукции (приготовлению блюд, коктейлей и т.д.), оплаты (чеки);
- разграничение прав доступа по различным категориям пользователей (официант, бармен, кассир, администратор зала и управляющий);
- печать предварительных счетов на оплату клиентам;
- ведение продаж по нескольким организациям (при подключении к одному рабочему месту нескольких фискальных регистраторов);
- дополнительные печатные формы технологических и калькуляционных карт, отражающих особенности технологии учета.

Конфигурация поддерживает работу со следующими видами торгового оборудования: фискальные регистраторы, принтеры марок, считыватели магнитных карт.

Самым лучшим способом анализа обслуживания, да и анализа работы предприятия в целом, является способ практического исследования.

Руководительская деятельность сосредоточена на одном человеке, что позволяет точно знать, кому дано задание и кем. Так же это позволяет держать в более жестком контроле все процессы производства, так как все сводится к одному человеку. У этой системы организации работы есть и свои минусы, например из-за того, что вся основная работа по организации лежит на одном человеке, в масштабе предприятия, она заставляет подотчетного человека работать в постоянном состоянии напряжения. А это ведет к снижению внимательности, сосредоточенности и, как следствие, ошибками снижению работоспособности.

Для того, чтобы облегчить и избавить администратора от решения некоторых организационных вопросов, связанных с организацией

обслуживания, в штат предприятия была внесена должность старшего официанта. Этот человек является прямым заместителем администратора со своими должностными обязанностями.

В данный момент такой метод организации работы является самым подходящим и удобным.

1.2. Современное состояние информационной деятельности ООО «ВРК»

За качественные разработки и успех воплощения в жизнь принятия в организации стратегии управления основную ответственность несёт непосредственно менеджер. Он - специалист по управлению, который создаёт планы, определяющие, не только что и когда делать, но также кто и как будет выполнять намеченное (управление персоналом), определяет рабочие процедуры (технологии) применительно ко всем стратегиям управленческого цикла, осуществляет контроль.

При этом предложение ИТ для предприятий общественного питания существует, а опыт имевших место внедрений показывает, что ИТ в ресторанном бизнесе не является лишним. Современный ресторан — это не просто место, где продают еду и выделяют место для ее поглощения, это красивый способ времяпрепровождения, один из основных элементов досуга в современных городах. Организация подобного процесса крайне сложна по своему содержанию и наполнению, требует соблюдения санитарных и технологических норм, контроля за стилем и культурой поведения обслуживающего персонала, за учетным процессом, а также анализа транзакций, учета поступления продуктов, формирования стоимости блюд и полуфабрикатов, процедур списания продуктов «под ноль» и т.д. Требование автоматизации всех этих процессов вытекает, прежде всего, из необходимости учёта большого количества деталей. Удобство автоматизации и информатизации процессов на предприятии общественного питания очевидно

не только с точки зрения «ведения дел», но и с позиций клиентов, так как ИС позволяют более оперативно работать с расчетами с посетителями, очередностью обслуживания, обеспеченностью предлагаемого меню всеми необходимыми ингредиентами.

Современные ИТ, разработанные специально для предприятий общественного питания, позволяют значительно упростить, оптимизировать и ускорить целый ряд рутинных повседневных, специфических для этого бизнеса операций.

Во-первых, ИТ берут на себя процедуру формирования блюда на основе набора ингредиентов (произвольно и часто меняющегося во времени) и схемы закупки продуктов (не только от организаций, но и с рынка по закупочным актам) и так далее. Автоматизируется ведение списка блюд с учетом нормативов расхода продуктов, сезонных норм закладок продуктов на основе использования справочников продуктов и блюд.

Во-вторых, современные приложения автоматически определяют расход ингредиентов по каждому блюду, списывают нужное количество продуктов и рассчитывают себестоимость блюд, формируют калькуляционные карточки на блюда в условиях динамического изменения закупочных, учетных и продажных цен. В большинстве ИТ заложены возможности ведения количественно-суммового учета продуктов и блюд с контролируемым процентом наценки при назначении розничных цен на блюдо, а также ведение истории отпускных цен на блюда и продукты.

В-третьих, ИТ значительно облегчают и делают более строгим ведение учета продуктов и блюд на нескольких кухнях и точках реализации. В данном случае ИТ позволят автоматически устанавливать разные цены на блюда и услуги, в зависимости от места реализации и единиц измерения. За счет автоматизации упрощается процесс движения товаров, например, организация поступления товаров непосредственно на кухню или место реализации, минуя кладовую или перемещение продуктов на кухни для производства блюда, либо в розничную продажу на точки реализации.

Наконец, ИТ позволяют формировать меню для зала и преискуранта барной продукции, а также всего комплекса документов и отчетов по общественному питанию (меню, наряд, марочный отчет, товарный отчет, заборный лист и прочее).

Таким образом, целью работы является повышение эффективности работы сети ресторанов ООО «ВРК» на основе разработки и внедрения базы данных, позволяющей автоматизировать и упростить множество операций в данной деятельности. Для создания и разработки базы данных ресторана используются в данном курсовом проекте программы Microsoft Office Access и ERwin.

1.3. Описание целей, функций и задач предприятия

В состав ООО «ВРК», входит ряд реторанов и кафе различной кухни. Рестораны «Корчма», Кофейня «Грильяж», Караоке-бар «Шарманка», Ресторан «Генацвале».

ООО «ВРК» развивает дополнительные сервисы. Рестораны компании, включают в себя различную кухню, рестораны Русской, Украинской, Кавказской кухни, Итальянские кофейни. Организация использует автоматизированные системы обслуживания - веб-приложения. Данные веб-приложения нацелены на увеличение обращений в компанию клиентами. Реализуются такие услуги как: заказ еды на дом, бронирование столов, хранение данных постоянных клиентов дал их поощрения бонусами и скидками.

Основные направления деятельности – рестораны и кафе. В сферу деятельности работников компании, отвечающих за информационное обеспечение, входит создание и сопровождение сайтов, веб-приложений, баз данных, которые хранят информацию о системах заказа и бронирования, таблиц постоянных клиентов. Деятельность сотрудников ИТ отдела, направлена на развитие компании и улучшение качества обслуживания.

Для достижения поставленной цели решаются следующие задачи:

- изучение предприятия и его функциональной структуры;
- изучение особенностей веб-приложений;
- изучение и анализ инструкций и технической документации систем веб-приложений;
- изучение принципов технической и программной поддержки систем веб-приложений;
- рассмотрение вопросов обеспечения безопасности жизнедеятельности на предприятии;
- освоение приемов и правил поддержки автоматизированных систем веб-приложений и работы с ними;
- изучение возможностей внедрения и разработка собственных веб-приложений.

Исходя из представленных выше целей и задач, время прохождения данной практики на базе компании ООО «ВРК» были поставлены следующие конкретные цели: изучить состояние информатизации на предприятии, получить и закрепить навыки администрирования информационных систем, используемых на предприятии.

Для реализации данной цели было необходимо выполнить следующие задания:

- изучение теоретической информации о веб-приложении;
- изучение возможностей внедрения технологии в информационную структуру предприятия;
- произвести проектирование автоматизированной системы обработки данных на сайте организации;
- разработка собственной системы – веб-приложения.

В данном отчете описываются результаты выполнения представленных выше заданий и задач производственной практики.

Целью деятельности ресторана является удовлетворение потребительских потребностей населения, в услугах и товарах в сфере общественного питания.

Основными задачами ресторана являются организация розничной торговли.

На рисунке 1.2. отображена организационная структура ресторана, включающая ее основные элементы и функциональные связи между ними.

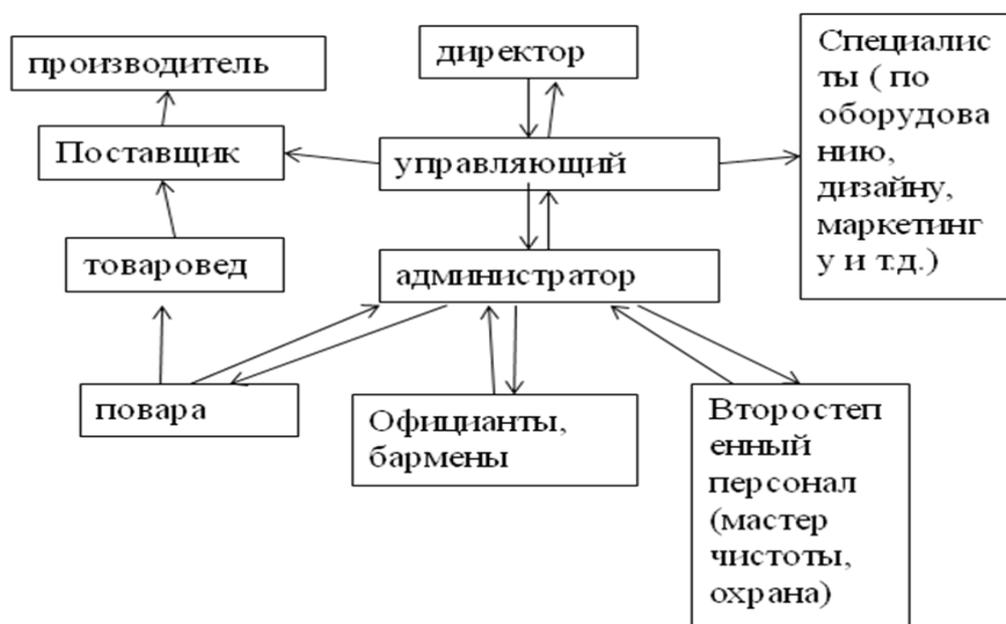


Рис. 1.2. Общая схема организации ресторана

Для реализации указанных задач ресторан осуществляет следующие виды деятельности:

- организация общественного питания;
- виды деятельности, подлежащие лицензированию и осуществляющиеся после получения соответствующих лицензий.

Рестораны предприятия - это общедоступное предприятие общественного питания, предоставляющее потребителям широкий ассортимент блюд сложного приготовления, а также вино-водочные, табачные и кондитерские изделия.

Высокий уровень обслуживания сочетается с организацией отдыха посетителей.

В ресторанах организуются обслуживание приемов, семейных торжеств, банкетов, проведение тематических вечеров. Основные посетители ресторана это жители близлежащих районов и работники находящиеся рядом офисов. Потребителей обслуживают официанты, бармены, прошедшие специальную подготовку.

В состав здания входят: производственные помещения, административные помещения, бытовые помещения для персонала, торговый зал, фойе.

Блюда и напитки готовят высококвалифицированные повара и бармены. Обслуживающий персонал имеет форменную одежду и обувь единого образца. В практику обслуживания входит устройство семейных обедов. Для этого составлено специальное меню в расчете на детей (детское меню), где предлагаются блюда, которые могут заинтересовать детей своим названием и оформлением. Рестораны имеют кроме обычной вывески, вывеску световую с элементами оформления.

В случае, когда в ресторане все цели достигнуты и задачи выполнены, ресторан и гости счастливы [7].

Структура управления рестораном - совокупность и соподчиненность взаимосвязанных организационных единиц или звеньев, выполняющих определенные функции. Элементом структуры служит орган управления, представляющий собой группу работников, которые объединены решением одной задачи - удовлетворить запросы потребителей. Во главе этой группы стоит руководитель-работник, выполняющий функции управления в соответствии с доверенным ему участком. Во главе коллектива ресторана находится администрация.

2. ПРОГРАММИРОВАНИЕ ВЕБ ПРИЛОЖЕНИЯ ДЛЯ ИНФОРМАЦИОННОЙ ПОДДЕРЖКИ ООО ВРК

2.1. Выбор средств разработки

Интерпретируемый серверный язык PHP.

PHP (англ. PHP: Hypertext Preprocessor – «PHP: препроцессор гипертекста», англ. Personal Home Page Tools (устар.) – «Инструменты для создания персональных веб-страниц») – скриптовый язык программирования общего назначения, интенсивно применяющийся для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков программирования, применяющихся для создания динамических веб-сайтов.

Язык и его интерпретатор разрабатываются группой энтузиастов в рамках проекта с открытым кодом. Проект распространяется под собственной лицензией, несовместимой с GNU GPL.

Одной из сильнейших сторон PHP является возможность расширения ядра путем разработки подключаемых модулей, «расширений» для работы с базами данных, динамической графикой, криптографическими библиотеками, документами формата PDF и др.

PHP поддерживает ООП (деструкторы, открытые, закрытые и защищённые члены и методы, интерфейсы и клонирование объектов). PHP поддерживает XML.

При запросе пользователя web-сервер просматривает документ, выполняет найденные в нем PHP-инструкции, а результат их выполнения возвращает пользователю. При этом статическая часть документа, написанная на языке HTML, фактически является шаблоном, а изменяемая часть формируется при исполнении PHP-инструкций. Сами скрипты находятся на сервере и их содержимое посетителю сайта просмотреть невозможно. В итоге

пользователь видит обычную веб-страницу, которая от других отличается лишь расширением.

Интерпретатор PHP подключается к веб-серверу либо через модуль, созданный специально для этого сервера (например, для Apache или IIS), либо в качестве CGI-приложения.

Основные достоинства PHP:

1. Простота модификации внешнего вида готового программного продукта (возможность создания множества шаблонов и переключения между ними).
2. Большое количество бесплатных решений на этой платформе.
3. Бесплатность самого интерпретатора и всех компонентов (MySQL, PHP).
4. Кроссплатформенность.

Основные недостатки:

1. Нестрогая типизация.
 2. Отсутствие поддержки многобайтовых кодировок в ядре языка
- Технология Microsoft ASP.Net.

ASP.NET — это технология создания динамических веб-приложений. Она является частью платформы .NET Framework. Приложения ASP.NET можно писать на любом языке, совместимом с технологиями .NET. В их число входят Visual Basic, C# и J#. Страницы ASP.NET (веб-формы) проходят компиляцию, что обеспечивает лучшую производительность по сравнению с приложениями на основе сценариев. Веб-формы позволяют создавать многофункциональные веб-страницы. При разработке можно использовать серверные элементы управления ASP.NET для создания общих элементов пользовательского интерфейса и программирования общих задач для них.

Процесс создания веб-форм ускоряется за счет использования стандартных встроенных компонентов и пользовательских компонентов, помогающих при написании кода страницы [3].

Технология Java EE.

Java Platform, Enterprise Edition, сокращенно Java EE (до версии 5.0 – Java 2 Enterprise Edition или J2EE) – набор спецификаций и соответствующей документации для языка Java, описывающей архитектуру серверной платформы для задач средних и крупных предприятий.

Основная цель спецификаций – обеспечить масштабируемость приложений и целостность данных во время работы системы. J2EE во многом ориентирована на использование её через веб как в интернете, так и в локальных сетях.

Платформа J2EE использует модель многоуровневого распределенного приложения. Логически приложение разделено на компоненты в соответствии с их функциональностью. Различные компоненты, составляющие J2EE-приложение, установлены на различных компьютерах в зависимости от их уровня в многоуровневой среде J2EE, которой данный компонент принадлежит. J2EE-приложения состоят из компонентов. J2EE-компонента представляет собой законченный функциональный программный модуль, встроенный в приложение J2EE с соответствующими классами и файлами и взаимодействующий с другими компонентами. J2EE-спецификация определяет следующие J2EE-компоненты:

1. Клиентские приложения и апплеты – это компоненты, работающие на клиентской машине.
2. Компоненты технологии Java-сервлет и JavaServer Pages (JSP) – это Web-компоненты, работающие на сервере.
3. Корпоративные компоненты – это бизнес-компоненты, работающие на сервере.

Спецификация Java EE предоставляет обширные возможности для разработки клиент-серверных приложений и в особенности удобные пути реализации серверной части приложения на языке Java. Клиентская же часть чаще всего основывается на стандартных html-страницах, дополненных java-апплетами и клиент-приложениями.

Обоснование выбора технологии.

Практически все рассмотренные технологии разработки веб-приложений функционально практически идентичны, т.е. все, что можно реализовать, используя одну из них, можно сделать и на любой другой. Основные различия состоят в скорости и общей стоимости разработки, масштабируемости полученного решения и удобства средств разработки.

Подводя итог сравнения, технология ASP.NET не подходит для решения поставленной задачи, т.к. требует дополнительных расходов на приобретение лицензий. Технология Java в целом удовлетворяет выдвинутым требованиям, однако, имеет достаточно большой порог входа и требует больше времени на освоение. Интерпретируемый серверный язык PHP подходит для реализации поставленной задачи по всем параметрам.

2.2. Требования к СУБД

Выбор СУБД является одним из важных этапов при разработке приложений баз данных. Выбранный программный продукт должен удовлетворять как текущим, так и будущим потребностям предприятия, при этом следует учитывать финансовые затраты на приобретение необходимого оборудования, самой системы, разработку необходимого программного обеспечения на ее основе, а также обучение персонала. Кроме того, необходимо убедиться, что новая СУБД способна принести предприятию реальные выгоды. Очевидно, наиболее простой подход при выборе СУБД основан на оценке того, в какой мере существующие системы удовлетворяют основным требованиям создаваемого проекта информационной системы.

Более сложным и дорогостоящим вариантом является создание испытательного проекта на основе нескольких СУБД и последующий выбор наиболее подходящего. Но и в этом случае используются определенные критерии отбора. Перечень требований к СУБД может изменяться в

зависимости от поставленных целей. Тем не менее, можно выделить несколько групп критериев:

- реализуемые режимы работы с БД ресторана и максимальное число пользователей одновременно обращающихся к базе, таких как заказчик, бухгалтер, повар;

- модель данных (предусмотренные типы данных, средства поиска, реализация языка запросов, средства поддержания целостности базы данных);

- особенности разработки приложений (средства проектирования, поддержка большого количества национальных языков, возможности разработки Web-приложений, поддерживаемые языки программирования);

- производительность (Рейтинг TPC (Transactions per Cent) т. е. отношение количества запросов, обрабатываемых за некий промежуток времени, к стоимости всей системы, возможности параллельной обработки данных, возможности оптимизирования запросов);

- надежность (сохранность информации при сбоях, обеспечение защиты данных от несанкционированного доступа);

- требования к рабочей среде (минимальные требования к оборудованию, максимальный размер адресуемой памяти, операционные системы, под управлением которых способна работать СУБД);

- требуемый уровень квалификации персонала ресторана;

- смешанные критерии (качество и полнота документации, стоимость, стабильность производителя, распространенность СУБД) [1].

2.3. Проектирование архитектуры веб приложения

В соответствии с требованиями, предъявляемыми к разрабатываемому программному изделию, оно должно иметь клиент-серверную архитектуру, что, прежде всего, означает разделение комплекса программных средств, необходимых для ее функционирования на две группы: клиентскую и

серверную, и влечет за собой необходимость использования большого числа стороннего программного обеспечения.

Моделируя деятельность ресторана, мы можем выделить как входную так и выходную информацию, так же стоит еще учесть и другие факторы, влияющие на деятельность предприятия - это законодательство, правила приготовления блюд, техническое обеспечение и другие факторы. При анализе деятельности предприятия общественного питания было выделено три основные работы входящие в состав предприятия.

"Клиент-сервер" - это модель, по которой могут взаимодействовать компьютеры в сети. Клиентский процесс запрашивает в этой модели некоторые услуги, а серверный процесс обеспечивает их выполнение при этом один серверный процесс может обслуживать множество клиентских процессов расположенных на разных удаленных компьютерах.

Клиент-серверная СУБД позволяет обмениваться клиенту и серверу минимально необходимыми объёмами информации. При этом основная вычислительная нагрузка ложится на сервер. Клиент может выполнять функции предварительной обработки перед передачей информации серверу, но в основном его функции заключаются в организации доступа пользователя к серверу.

К целям и задачам проекта можно отнести:

- закрепление практических навыков программирования, полученных на лабораторных занятиях;
- углубление теоретических и практических знаний в области методологии и программирования и разработки приложений баз данных;
- развитие навыков самостоятельного планирования и выполнение научно-исследовательской работы;
- получение опыта сбора и обработки исходного материала, анализа научно-технической литературы, справочников, стандартов и технической документации;

- приобретение навыков обоснования принимаемых проектных решений и профессионального оформления проектной документации.

Процесс проектирования БД представляет собой последовательность переходов от неформального словесного описания информационной структуры предметной области к формализованному описанию объектов предметной области в терминах некоторой модели. В общем случае можно выделить следующие этапы проектирования:

- Словесное описание информационных объектов предметной области;
- Проектирование инфологической модели предметной области;
- Логическое проектирование БД, то есть описание БД в терминах принятой диалогической модели данных;
- Физическое проектирование БД, то есть выбор эффективного размещения БД на внешних носителях для обеспечения наиболее эффективной работы приложения.

Инфологическое проектирование, в свою очередь, связано с попыткой представления семантики предметной области в модели БД. Реляционная модель данных в силу своей простоты не позволяет отобразить семантику, то есть смысл предметной области. Ранние теоретические модели и графы в большей степени отображали семантику предметной области. Они в явном виде определяли иерархические связи между объектами предметной области.

Процесс проектирования БД представляет собой последовательность переходов от неформального словесного описания информационной структуры предметной области к формализованному описанию объектов предметной области в терминах некоторой модели. В общем случае можно выделить следующие этапы проектирования:

1. Словесное описание информационных объектов предметной области;
2. Проектирование инфологической модели предметной области;
3. Логическое проектирование БД, показано на рис. 2.1. то есть описание БД в терминах принятой диалогической модели данных;

4. Физическое проектирование БД, то есть выбор эффективного размещения БД на внешних носителях для обеспечения наиболее эффективной работы приложения.

Инфологическое проектирование прежде всего связано с попыткой представления семантики предметной области в модели БД. Реляционная модель данных в силу своей простоты и лаконичности не позволяет отобразить семантику, то есть смысл предметной области. Ранние теоретико-графовые модели в большей степени отображали семантику предметной области.

2.4. Проектирование структуры базы данных, датологическое программирование

В соответствии с требованиями, вся информация, обрабатываемая разрабатываемым программным изделием, должна храниться в базе данных.

Логическая модель базы данных ресторана содержит следующие сущности:

1) «Сотрудник», атрибуты – ключевой «Номер сотрудника», «Фамилия», «Имя», «Отчество», «Дата рождения», «Должность», «Адрес», «Телефон», «Оклад», «Номер отдела», «Номер помещения»;

2) «Поставщик», атрибуты – ключевой «Номер поставщика», «ИП», «Адрес», «Телефон», «Номер сертификата», «Контактное лицо»;

3) «Меню», атрибуты – ключевой «Номер блюда», «Наименование блюда», «Вес», «Цена», «Категория»;

4) «Материальная база», атрибуты – ключевой «Номер помещения», «Наименование помещения», «Оборудование»;

5) «Отдел», атрибуты – ключевой «Номер отдела», «Наименование отдела»;

6) «Поставки», атрибуты – ключевой «Номер поставки», «Дата поставки», «Сумма», «Номер сотрудника», «Номер поставщика»;

7) «Меню_Сотрудник», атрибуты - «Номер блюда», «Номер сотрудника», которые образуют первичный составной ключ.

В логической модели данных ресторана установлены следующие типы связей:

1) идентифицирующая - между сущностями «сотрудник» и «меню» (М-М), «сотрудник» и «поставки» (1-М), «поставщик» и «поставки» (1-М);

2) неидентифицирующая – между сущностями «сотрудник» и «отдел», «сотрудник» и «материальная база».

На рисунке 2.2 приведена ненормализованная логическая модель ресторана. Для ее основы использовалась построенная в предыдущем пункте модель «сущность – связь» ресторана.

Каждый экземпляр сущности должен быть уникален и отличаться от других атрибутов.

При установлении идентифицирующей связи атрибуты первичного ключа родительской сущности автоматически переносятся в состав первичного ключа дочерней сущности. Эта операция дополнения атрибутов дочерней сущности при создании связи называется миграцией атрибутов. В дочерней сущности новые атрибуты помечаются как внешний ключ - (FK).

Таким образом, в модели ресторана в сущности «Поставки», находящейся в идентифицирующей связи со сущностями «Сотрудник» и «Поставщик», отмечены внешние ключи «номер сотрудника» и «номер поставщика» над чертой. Сущности «Отдел» и «Сотрудник», «Материальная база» и «Сотрудник» независимы друг от друга, поэтому связь между ними не идентифицирующая, а у сущности «Сотрудник» зафиксированы внешние ключи «номер отдела» и «номер помещения» под чертой.

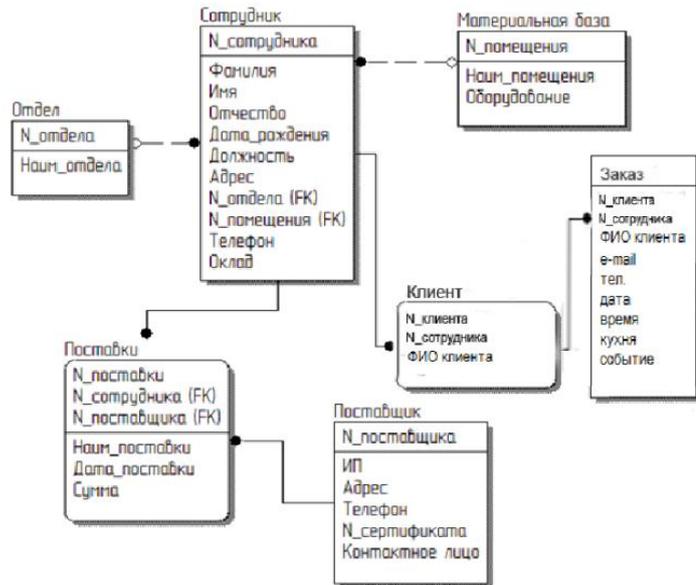


Рис. 2.1. Логическая модель базы данных ресторана

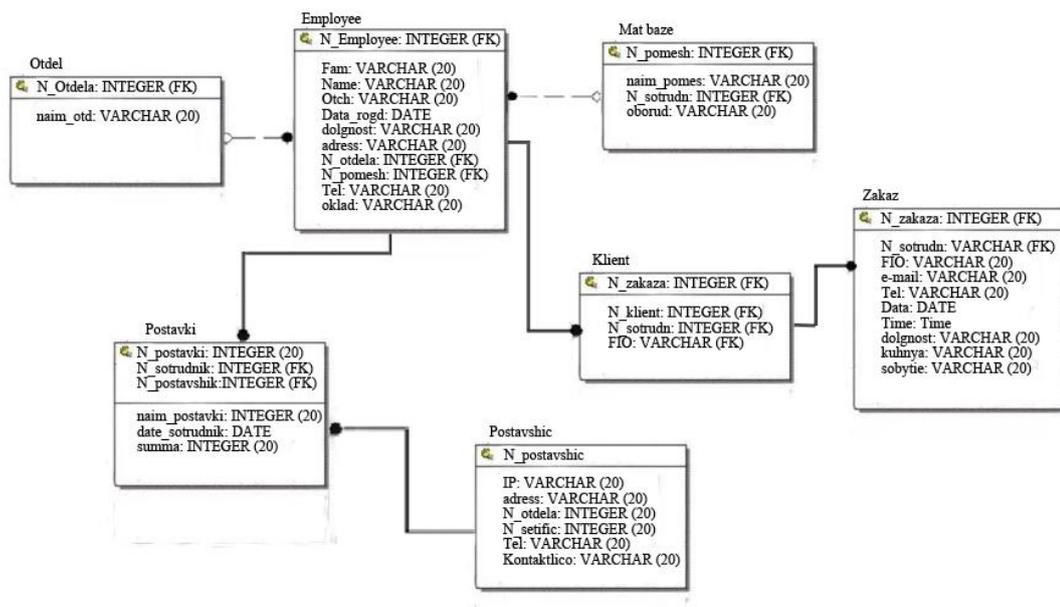


Рис. 2.2. Физическая модель базы данных ресторана

2.5. Физическое программирование

После того, как проектирование Web-приложения закончено, можно перейти непосредственно к реализации проекта. В плане внешнего оформления, страницы являются однотипными. Отличия в содержимом страниц, формах, необходимых для работы с БД, сценариях, реализующих взаимодействие БД и пользователя. Администратор web-приложения редактирует базу данных входя в личный кабинет сайта и получает доступ к выбору таблиц. Выбрав нужную ссылку нужной таблицы, пользователь переходит на страницу где может просматривать данные таблицы, добавлять данные, удалять, осуществлять поиск необходимой информации.

Рассмотрим реализацию доступа к базе данных. Основные настройки, необходимые для подключения к БД, а также сама функция подключения `mysql_pconnect(host,user,pass)`, находятся в каждом файле осуществляющем доступ к данным базы.

Добавление данных в БД реализовано с помощью страницы `zakaz.php`, `employee.php`, `otdel.php`, `mate_baze.php`, `zakaz.php`, `klient.php`, `postavki.php`, `postavshik.php` и модулях `l6.php` и `l7.php`.

На страницах расположена форма для заполнения данных таблиц и последующего отправления запроса на запись. На этой форме все поля текста, списка соответствуют полям таблицы.

В листинге 2.1 представлен сценарий добавления записи в таблицу «Заказы».

Поиск данных реализован при помощи модулей. На этих страницах происходит формирование параметров поиска: выбирается таблица, в которой будет произведен поиск, происходит выбор поля, по которому будет происходить поиск и, естественно, вводится параметр поиска. После этого, модуль формирует и производит запрос к БД и, в зависимости от выбранного параметра поиска, производит вывод результатов поиска. Поиск производится с помощью запроса `select * from table_name where field_name like`

'%search_parametr%'. Где table-name – название таблицы. Field-name – название поля таблицы. Search_parametr- параметр поиска.

Листинг 2.1. Добавление записи в таблицу «заказы»

```
<?php
mysql_query("SET NAMES 'utf8'");
$id=$_POST['id'];
$name org=$_POST['name org'];
$kollich=$_POST['kollich'];
$name pred=$_POST['name pred'];
$db = mysql_pconnect("172.23.64.64", "name", "1");
mysql_select_db('db_name ');
$query = "INSERT INTO `postav`(`id`, `name org`, `kollich`, `name pred`)
VALUES (`$id`, `"$name org`", $kollich, $name pred);";
$result = mysql_query($query);
header('Location: http://172.23.64.64/ftp/21/.../l/r14.php');
die();
?>
```

Реализация запросов к данным в БД:

Запросы к данным в БД реализованы посредством стандартных средств языка PHP.

Сначала формируется запрос \$query. Затем с помощью функции mysql_query мы присваиваем переменной \$result указатель на результат запроса. Затем, для вывода таблиц, формирования списков используются функции:

- mysql_query(\$query),
- mysql_num_rows(\$result),
- mysql_fetch_array(\$result)

Функция mysql_query(\$query) - Посылает запрос MySQL

Функция `mysql_num_rows($result)` - Возвращает количество рядов результата запроса.

Функция `mysql_fetch_array($result)` - Обрабатывает ряд результата запроса, возвращая ассоциативный массив, численный массив или оба.

Удаление из базы данных происходит посредством выполнения следующих функций:

```
$query = "DELETE FROM tovdannye WHERE name = " . $inf_del . "" -
```

Данная функция формирует запрос на удаление из базы данных

```
$result = mysql_query($query, $db) – данное выражение отправляет ранее составленный запрос в базу данных для его последующей обработки.
```

Листинг 2.2. Удаление записей таблицы

```
<?php
$num=$_POST['num'];
@ $db = mysql_pconnect("172.23.64.64", "991841", "1");
mysql_select_db('db_name');
mysql_query('SET NAMES utf-8');
$dquery="delete from `adres` where `id`=$num";
mysql_query($dquery);
header('Location: http://172.23.64.64/ftp/18/991841/kurs/rl1.php');
die(); ?>
```

Таким образом нами были реализованы модули приложения, в том числе страница администратора, с которой он может просматривать, удалять, добавлять необходимые записи базы данных.

3. ТЕСТИРОВАНИЕ И АПРОБАЦИЯ

3.1. Тестирование модулей веб приложения

База данных не имеет смысла, если между ней и пользователем должна быть обеспечена связь, реализующая ряд операций над вводом, проверкой, сохранением, поиском и редактированием данных. Данная связь и будет предоставлять набор функциональных возможностей сайта ремонта компьютерной техники.

На сайте необходимо должна быть главная страница, на которой пользователь, желающий получить информацию, всегда сможет её получить. Данная страница является статической и никак не взаимодействует с базой данных.

В случае заказа нового товара или услуги, потребуется завести запись в соответствующей таблице. Для этого нужна форма для создания записи. Наконец, нужно иметь возможность связи с постоянными покупателями и поставщиками.

Таблица должна иметь возможность удалить данные в случае необходимости. Например, удаление нужно, если клиент хочет вернуть товар или отказаться от услуги или удалить данные о себе при совершении заказа.

Важной составляющей сайта компании занимающейся ремонтом компьютерной техники является поиск. Особенно, когда в базе данных имеется много записей.

После того, как web-приложение создано, будет логичным протестировать это приложение. Рассмотреть работу каждого модуля отдельно.

Пользовательский интерфейс представлен в виде страниц. Для верстки страниц использовалась таблица. Главная страница имеет расширение .html и является статической. Остальные страницы формируются сервером и имеют расширения .php. Главная страница представляет собой «визитную карточку», приветствующую посетителя. В навигационном меню, расположенном в левой

части главной страницы расположены ссылки на основные интерфейсы подсистем Web-приложения. На страницах подсистем, также имеется навигационное меню, позволяющее перемещаться между подсистемами, а также вернуться на главную страницу.

Рассмотрим интерфейсы подсистем web-приложения по порядку. Наше web-приложение было создано в виде лендинга или Landing Page .

Landing Page (лендинг) является отдельной страницей, которая создана с целью ознакомления потенциального покупателя с минимальным объемом информации таким образом, чтобы он не только заинтересовался, но и совершил какое-либо действие, например перешел по. Преимущества лендинга перед многостраничным сайтом кроются в относительно легком и быстром создании. Разработка лендинга имеет смысл в том случае, когда в процессе рекламной кампании необходимо привлечь как можно больше внимания к товару или услуге и продать его в большом количестве. Лендинг пейдж значительно влияет на конверсию трафика в сторону повышения, что имеет огромное значение при раскрутке сайта.

На целевую страницу, как правило, попадают с различных источников, например, благодаря контекстной рекламе или результатам выдачи поисковых систем. В таком случае вполне уместен вопрос: почему под определение «лендинг пейдж» не подходит любая страница сайта? Дело в том, что если рассматривать лендинг с технической точки зрения, то при том, что она состоит, как и любая страница в интернете, из html, css, тестовой и графической информации, цель ее заключается несколько в ином. Главная задача любого лендинга — собрать как можно больше лидов (клиентов, которые заполнили предложенную на странице форму, оставили свои контактные данные, т.е. заинтересовались и желают узнать о товаре больше или сразу его приобрести). Чтобы разобраться, почему лендинг пейдж, цена которого ниже создания сайта-визитки и сайта-каталога, во многих случаях эффективнее справляется со своей задачей, рассмотрим основные ее преимущества.

Преимущества лендинг-пейдж перед страницей обычного сайта.

Рекламное предложение, которое несет в себе лэндинг-пейдж, всегда конкретно и не размыто (как правило, это один товар или услуга), что повышает вероятность заинтересованности клиента по сравнению с сайтом, где товаров много и нужно потратить определенное количество времени, чтобы найти необходимый.

Навигация лендинга максимально проста, чего не скажешь о сайте, где нужно найти товар, прочитать о нем, а потом — перейти на вкладку с контактами.

Создание лендинг-пейдж проще с технической точки зрения, то же касается и изменения на нем информации, которое не требует больших временных затрат.

Такая страница быстро загружается, т.к. на ней не присутствует тяжелых графических и других элементов, замедляющих этот процесс.

Несколько лендингов могут быть размещены на одном домене.

Наблюдать за статистикой страницы удобно, а изменив одну лендинг-пейдж можно отследить и проконтролировать то, как контент и графика отражаются на конверсии страницы.

Благодаря наличию на странице лендинга формы обратной связи с основными контактными данными, создается клиентская база, при работе с которой можно достичь увеличения продаж до 30%. Главная страница изображена на рис. 3.1.

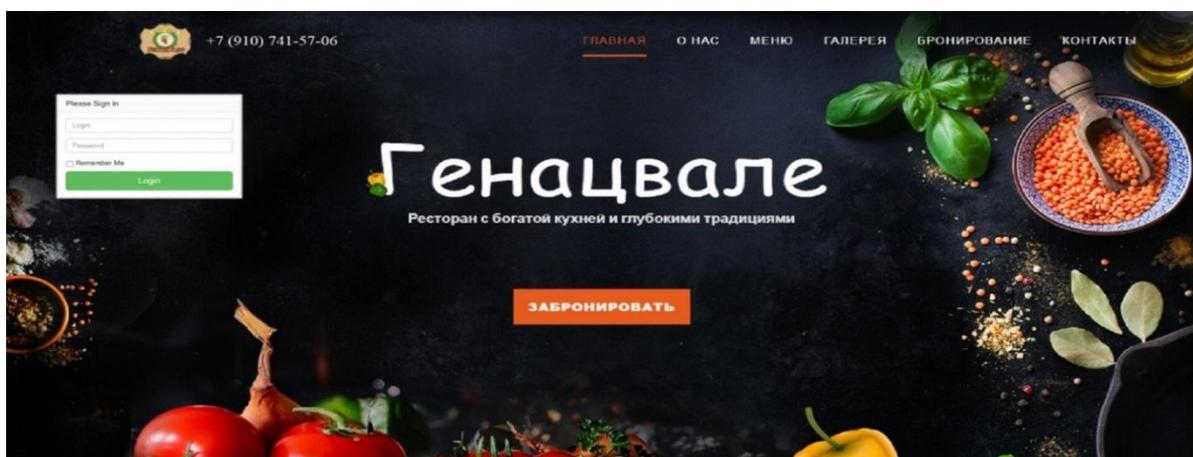
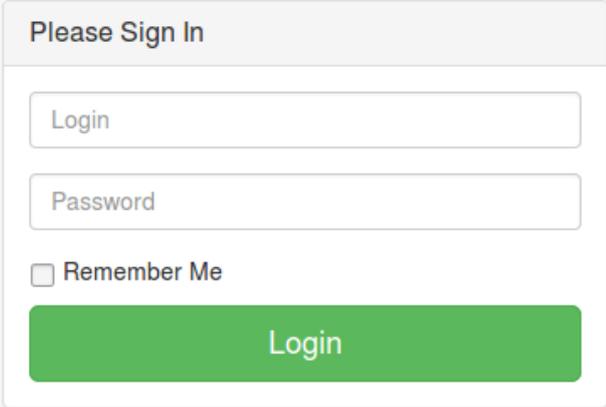


Рис. 3.1. Главная web-приложение.

Для работы с базами данных сайта администратором предусмотрена форма авторизации. Пользователь вводит свои данные и попадает в меню для редактирования и просмотра баз данных, где осуществляется работа с данными по заказам, поставщикам, постоянным клиентам, сотрудникам. Администратор сайта, через панель авторизации, заходит в меню администратора, вводя личный логин и пароль. Далее администратор получает доступ к редактированию таблиц базы данных (рис. 3.2.).



The image shows a login form with the following elements:

- Title: Please Sign In
- Input field: Login
- Input field: Password
- Checkbox: Remember Me
- Button: Login (green)

Рис. 3.2. Форма авторизации

Одним из важных элементов web-приложения является «Бронирование». Многие клиенты всё больше бронируют столик именно через интернет. На сегодняшний день этот метод наиболее удобный, выгодный и быстрый, что и объясняет причину его популярности. При этом необходимо понимать, что метод отличается своими особенностями, а также технологией

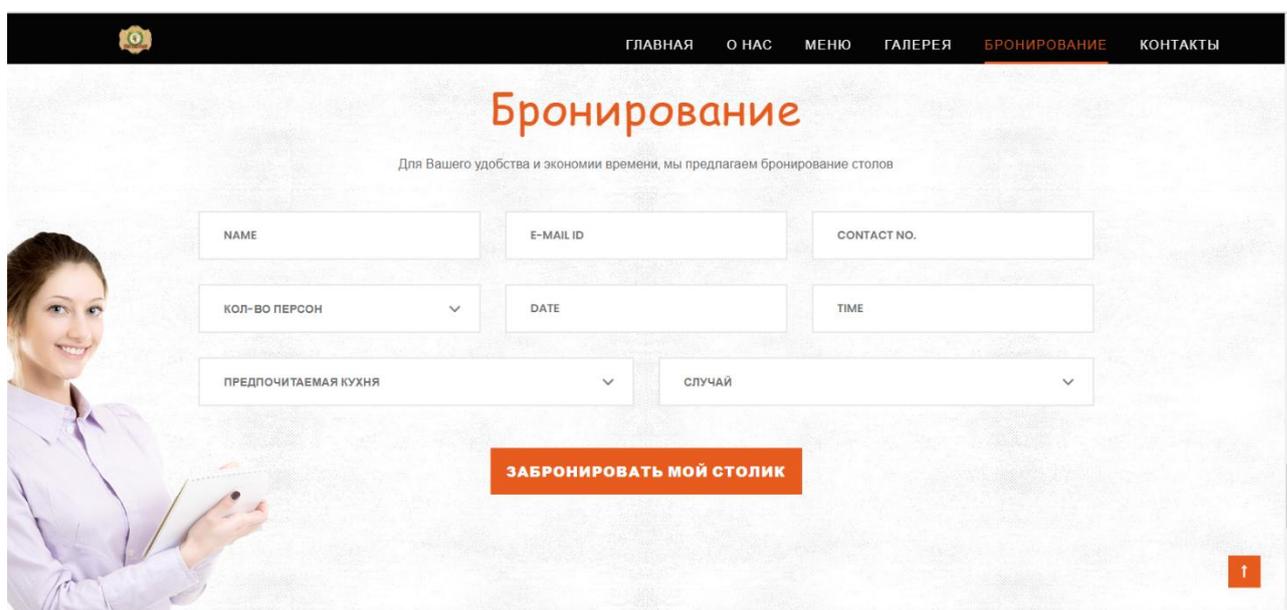
Довольно часто, перед тем как забронировать столик, изучаются отзывы. Это позволяет клиенту сориентироваться, а также получить информацию о ресторане, сделать правильный выбор. Речь идет про встраивание системы, предназначенной для бронирования онлайн, непосредственно в ресурс отеля. Она используется в качестве формы бронирования относительно запроса.

Оценивая все этапы бронирования, можно увидеть, что они выполняются в течение нескольких минут. Таким образом, быстрота и экономия времени – важные преимущества бронирования онлайн. Такие ресурсы пользуются особенной популярностью, поскольку позволяют посетителям забронировать нужные столы в нужный день.

Особенно удобна данная функция, когда пользователь бронирует стол и дату для важного события, что так же отражено в форме заказа, например годовщина или свадьба. Для подобных событий, администратор оперативно связывается с клиентами для уточнения подробностей заказа

Пользователь вводит в форме свои данные, которые заносятся в базу данных, после чего с этими данными работает администратор, данная страница показана на рис. 3.3.

Услуга бронирования заказа. Наличие сайта дает такую возможность. Наличие услуги доставки и оформления заказа в режиме он-лайн добавит популярности или ресторану и дополнительно увеличит доход.



The image shows a screenshot of a website's reservation page. At the top, there is a navigation bar with links: ГЛАВНАЯ, О НАС, МЕНЮ, ГАЛЕРЕЯ, БРОНИРОВАНИЕ (highlighted), and КОНТАКТЫ. The main heading is "Бронирование" in orange. Below it, a sub-heading reads: "Для Вашего удобства и экономии времени, мы предлагаем бронирование столов". The form contains several input fields: NAME, E-MAIL ID, CONTACT NO., КОЛ-ВО ПЕРСОН (with a dropdown arrow), DATE, TIME, ПРЕДПОЧИТАЕМАЯ КУХНЯ (with a dropdown arrow), and СЛУЧАЙ (with a dropdown arrow). A prominent orange button labeled "ЗАБРОНИРОВАТЬ МОЙ СТОЛИК" is centered below the form. On the left side of the form, there is a photograph of a smiling woman in a light purple shirt holding a notepad. A small orange square icon with a white exclamation mark is located in the bottom right corner of the page.

Рис. 3.3. Форма бронирования

Обратная связь - ключевой элемент системы ресторанного бизнеса. По обратной связи ресторан может отслеживать мнение гостей, свой прогресс в

целях, делать прогнозы на будущее, корректировать свои цели, чтобы достигать их наиболее эффективным путем. Обратная связь дает понимание ошибок или верных решений, показывает, насколько правильная выбрана концепция ресторана, насколько качественные его услуги. Важно, чтобы механизм обратной связи поддерживался в рабочем состоянии. Если обратная связь - узкое звено – это приводит к регрессу бизнеса, потере денег и внутренним кризисам ресторана.

Форма подписки позволяет выделить наиболее частых клиентов сайта, и заинтересовать их, например, осуществляя рассылки с акциями, скидками, новостями и другой информацией ресторанов. Так же на сайте имеется интеграция с популярными соц. сетями, для повышения лояльности клиентов и получения обратной связи. Пользователь может перейти в соц. сети, участвовать в обсуждениях, смотреть новости, истории от поваров, новшества приготовления еды со всего мира. Форма подписки на рассылку показана на рис. 3.4.

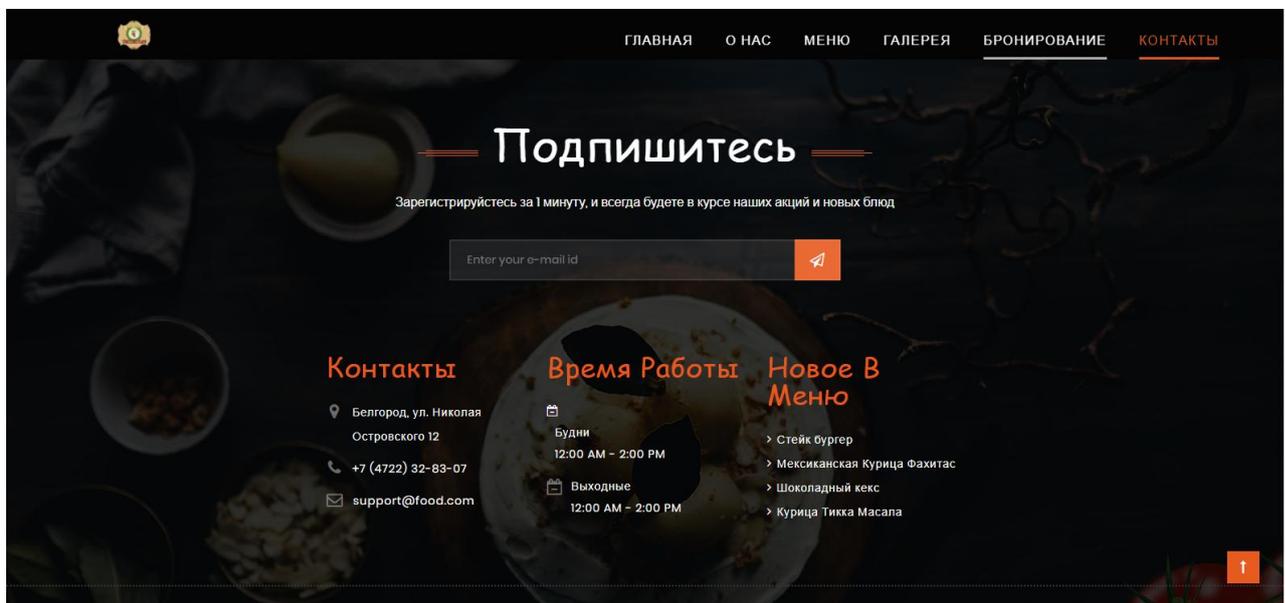


Рис. 3.4. Форма подписки

На сайте выставлены отзывы клиентов, которые пользователи могут просматривать. Отзывы о компании – это эффективнейший инструмент, позволяющий повысить продажи в разы. Прочитав этот текст, вы узнаете, как благодаря одному отзыву сделать из каждого клиента лучшего продавца.

Возможности размещения отзывов на корпоративном сайте.

Вероятность заказов в ресторане, имеющем более полусотни отзывов в 3,5 раза выше, чем в похожем магазине, ресторане с одним отзывом, и в 2,8 раза выше, чем в магазине, где посетители оставили десять отзывов.

Эксперты отмечают, что и негативные отзывы! оказывают положительное влияние на количество продаж интернет-заказов. Так интернет-магазины, оперативно реагирующие на негативные отзывы своих покупателей и помогающие им решить свою проблему, способствуют росту покупок! Именно поэтому владельцам интернет-магазинов, сайтов ресторанов нужно постоянно следить за отзывами, появляющихся на их сайте, и стремиться к оперативному решению проблем покупателей.

Кроме того, если на сайте размещены только положительные отзывы, клиенты часто задумываются в их правдоподобности и вероятность покупки снижается! Пример использования отзывов в нашем web-приложении показано на рис. 3.5.

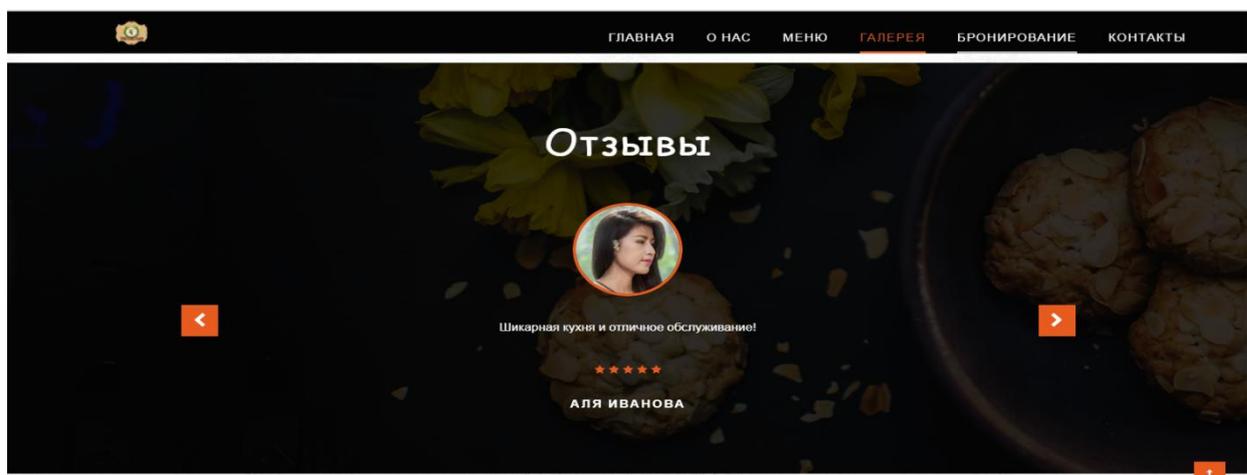


Рис. 3.5. Отзывы

На странице меню, рис. 3.6., пользователи сайта могут ознакомиться с меню ресторана, заранее выбрать блюда, узнать цены и увидеть фото блюд. На сайте ресторана будет размещена вся информация о заведении, услугах, ценах, фотографии интерьеров, подробное описание. Это поможет привлечь еще больше клиентов, ведь сайт для ресторана – это дополнительная реклама. Многие люди, посетив сайт, ознакомившись с ценами и услугами, со всей информацией, захотят посетить ресторан, а значит, станут новыми клиентами.

Постоянное размещение на сайте новых меню, новых блюд, информации об акциях – это тоже способ привлечь в ресторан новых посетителей, которые захотят попробовать новое блюдо или воспользоваться условиями акции.

Реальные фото блюд - это идеальный способ продемонстрировать ваши блюда, но при этом не стоит забывать про качество изображений, фото должны подходить выбранному дизайну сайта. Для этого были сделаны фотографии в одном стиле. Картинки должны привлекать пользователей.

Администратор сайта постоянно следит за актуальностью информации, обновляет меню.

Если человек не был в вашем заведении, но попробовал услугу доставки и ему захотелось насладиться не только кухней, но и атмосферой заведения в целом, ему нужно знать куда ехать. Или же он просто хочет сравнить качество блюд в заведении с едой из доставке, поблагодарить шеф-повара и т. д. Обязательно указывайте адрес и контактный телефон на главной странице сайта (в правом нижнем углу).

Страница «О нас» (Рис. 3.7.) знакомит пользователя с информацией о ресторане, месторасположении ресторана, достоинствах кухни и интерьера. Также на этой странице находятся специальные блюда, эта страница постоянно обновляется администратором сайта.

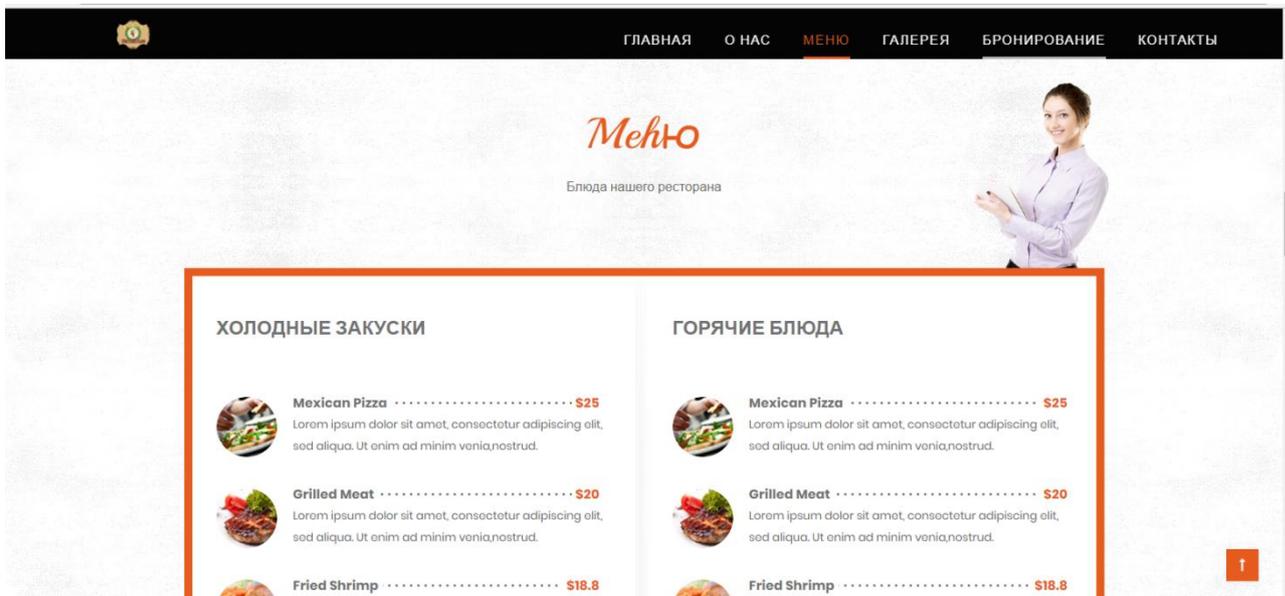


Рис. 3.6. Страница «меню»



Рис. 3.7. Страница «меню»

Администратор сайта, заходя в меню администратора, получает доступ к таблицам базы данных сайта (Рис. 3.8.). Переходя по ссылкам, администратор открывает нужную таблицу для работы с ней, добавления, удаления, просмотра информации и поиска.

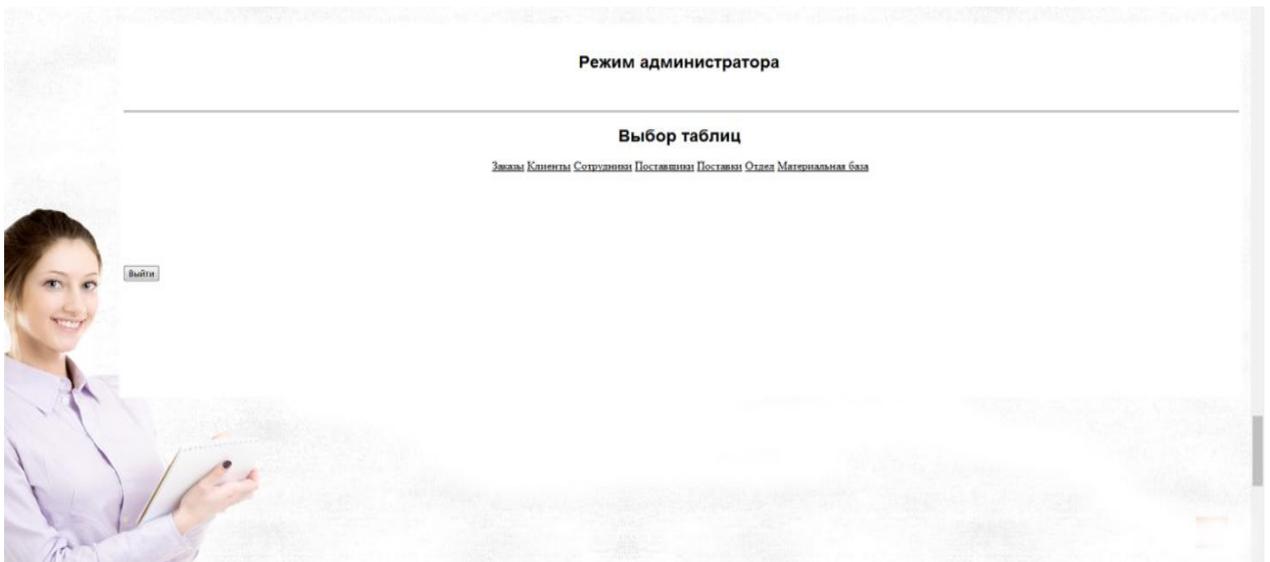


Рис. 3.8. Режим администратора

На примере таблицы «Заказы», изображённой на рисунке 3.9, показан пример работы с таблицами. Администратор может заносить данные в поле «добавление», пользователю необходимо заполнить все поля таблицы. Аналогично работает функция удаления, пользователь вводит номер заказа, который необходимо удалить. Для поиска необходимого заказа, администратор вводит данные, которые нужно найти.

Информация о заказах

Работа с таблицей

Номер заказа	номер сотрудника	ФИО клиента клиента	тел.	Дата	Время	e-mail	Кухня	Событие
1	1	Пронько Евгений Петрович	89034543423	15.06.2018	12:00	-	Европейская	-
2	1	Потапов Илья Сергеевич	89516547383	17.06.2018	13:00	-	Европейская	-
3	1	Корелин Станислав Игоревич	89067543823	17.06.2018	15:00	-	Европейская	-
4	2	Смирнова Ольга Дмитриевна	890614537485	17.06.2018	15:00	-	Кавказская	Годовщина
5	2	Савельев Олег Леонидович	89034392838	18.06.2018	16:00	-	Кавказская	-
6	1	Саченко Сергей Викторович	89038868888	22.06.2018	16:00	-	Европейская	-
7	1	Кирилова Анна Павловна	89038574434	23.06.2018	16:00	-	Европейская	День рождения
8	1	Григорьева Галина Петровна	89614543737	25.06.2018	17:00	-	Кавказская	-

Добавление заказа:

Номер заказа номер сотрудника ФИО клиента клиента

тел. Дата Время e-mail Кухня Событие

Удаление товара:

Введите номер товара, который нужно удалить

Рис. 3.9. Пример страницы редактирования таблицы «Заказы» базы данных

3.2. Опытная эксплуатация, развертывание.

Опытная эксплуатация - представляет собой использование программного комплекса персоналом Заказчика на реальных данных с целью определения фактических значений количественных и качественных характеристик программного комплекса, выявления и устранения недостатков.

Внедрение проекта позволит усовершенствовать контроль за движением товарно-продуктовых ценностей и денежных средств учреждения. Повысить эффективность разработки фирменного меню, минимизировать затраты, связанные с производственным процессом, уменьшить количество злоупотреблений со стороны персонала, сформировать привлекательную для посетителей атмосферу заведения.

Дальнейшее развитие системы может предусматривать создание дополнительной функциональности. В частности, ведение единого карточного счета клиента, составление оперативного плана закупок, доставки еды, реализация механизма подбора замен продуктов с учетом имеющегося складского запаса, формирование итоговых отчетов по прибыльности заведения.

Эксплуатационная документация ИС составлена с учетом потребности пользователей как для организации начального последовательного изучения функций системы, так и использования в качестве справочного пособия.

Известно, что любая более или менее сложная программа содержит ошибки. Web-приложения весьма и весьма сложны. Возникающие в них ошибки могут носить случайный характер или проявляться только при высокой загруженности Web-сервера. Поэтому нужно быть готовыми к тому, что наиболее коварные ошибки появятся уже после ввода системы в эксплуатацию. Наличие этапа опытной эксплуатации обязательно, и чем большее количество людей примет в нем участие, тем лучше.

Сохраняя последовательность повествования, мы переводим фокус нашего внимания с разработки на развертывание web-приложения. На стадии

развертывания, после проверки качества и безопасности приложения, мы переходим к оценке уязвимости и тестированию на проникновение для того чтобы убедиться в соответствии нашего приложения современным требованиям и его устойчивости к атакам. Смотря на безопасность web-приложений, как на непрерывный процесс с перекрывающимися этапами. Хотя мы и делим процесс на отдельные этапы, чтобы сделать повествование более понятным и структурированным, это совсем не значит, что один этап заканчивается, когда начинается другой. К примеру, продолжая использовать динамический анализ на этапе развертывания, а также продолжать поиск уязвимостей и тесты на проникновение на всех этапах. Также показывая общую картину и имеющиеся возможности.

На данном этапе на сайте определяются:

- Соответствие требованиям Техническому заданию и проекту визуальных форм;
- Устойчивость программного продукта к внешнему воздействию (попытка задания некорректных условий, подмена программных модулей, удаление модулей, изменение конфигурации и др.);
- Стабильность работы во времени и другие характеристики;

Web-приложение (сайт) создается на платном домене. Поэтому сайт изначально делался не на локальной машине, а на сервере, поэтому для начала эксплуатации, необходимо будет только открыть его к просмотру.

Для проведения опытной эксплуатации мной был создан вариант Web-приложение на основе данных ресторана "Генацвале", для любых пользователей которые хотели бы посмотреть данный сайт ресторана.

Разработка Web-приложения была своевременно завершена, передана руководству ООО «ВРК» и прошла опытную эксплуатацию, что подтверждено актом о вводе в эксплуатацию от 10.05.2018.

С данным Web-приложением работали сотрудники ресторана, ответственные за проверку данного Web-приложения на наличие

работоспособности всех модулей. Каждый из пользователей регистрировался и работал с функциями сайта, бронирования столов, подписки, регистрации.

Каждый из зарегистрированных пользователей проверял по своему усмотрению необходимые функции. Администратор заходя в администраторскую часть мог определить сделанные заказы на бронирование и сообщить исполнителям (официантам, администраторам зала) о заказе и ставил цели исполнения. Администратор тестировал функции работы с таблицами базы данных сайта. Программный продукт прошел опытную эксплуатацию.

3.3. Анализ соответствия реализации заявленной спецификации.

После создания web-приложения, необходимо, чтобы оно соответствовала реализации заявленной спецификации и поставленной задачи.

Web-приложение разработано с использованием языка разметки HTML, языков программирования PHP. Базы данных реализованы с помощью языка SQL. После окончания работ была обеспечена возможность заказчику самостоятельно вносить изменения (редактировать) в структуру и содержимое web-приложения.

По окончании работ мной был предоставлен полностью функционирующий сайт, исходные графические материалы по дизайну, все необходимые данные для доступа к системе управления сайтом (аккаунты, пароли и т.п.).

Дизайн выдержан в строгих и мягких тонах. Использовать преимущественно сине-голубые оттенки. Дизайн сайта должен быть выполнен с использованием языка HTML и CSS, при необходимости для создания отдельных графических элементов. Web-приложение выполнено в виде лэндинг-сайта

Техническим заданием было определено, что сайт должен корректно отображаться в браузерах Microsoft Internet Explorer; Mozilla FireFox; Opera.

Создана структура (шаблон) сайта, состоящая из следующих элементов:

- “шапка”. В данном блоке расположены логотипы предприятия, название предприятия, номер телефона;

- блок отображения главного меню сайта. Данный блок должен содержать перечень всех основных страниц. Меню располагается в головной части сайта;

Блок для входа администраторов в меню администратора;

Сайт позволяет пользователям осуществлять навигацию по сайту, совершать бронирование столов в ресторане.

Система управления сайтом позволяет:

- управлять базой данных (добавлять, удалять, изменять их содержимое);

- управлять элементами меню;

- добавлять/изменять/удалять новости на сайте;

Требования к содержимому сайта, поставленные заказчиком были выполнены, реализована основная информация о ресторане, меню ресторана с актуальными изображениями блюд. Было реализовано бронирование столов через меню «бронирования», панель администратора для редактирования сайта и баз данных. Добавлена возможность подписки на рассылку пользователей. Выделена контактная информация.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной выпускной квалификационной работы было разработано «web-приложение учёта заказов для ООО «ВРК» с использованием JavaScript, PHP и MySQL.

Были изучены основы работы MySQL баз данных и языка программирования для Интернет PHP и получены навыки работы с языком. Так же была разработана база данных и сайт, при помощи которого пользователь может добавлять и просматривать записи в базе данных.

Выполнены все задачи курсовой работы:

- проанализирована предметная область и выбраны инструментальные средства разработки web-приложения;
- спроектировано web-приложение;
- разработано web-приложения;
- протестированы информационные системы.

В данном проекте были использованы последние версии изучаемых языков и систем, что очень важно, так как в настоящее время появляется всё больше новых стандартов и версий приложений, что позволяет использовать большие возможности для разработки web-приложений.

Было разработано стабильное web-приложение для учета свежей информации учёта заказов, удовлетворяющее целям и задачам предприятия ООО «ВРК».

В ходе разработки нами были получены новые знания и отработаны практические умения и навыки по работе с HTML страницами.

Также нами были сделаны выводы, что PHP является мощной средой программирования, а MySQL очень удобной средой хранения и обработки данных. PHP и MySQL, являются развивающимися и перспективными средами для разработки веб-приложений.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Волкова, Т.В. Проектирование и создание БД: учебное пособие / Т.В. Волкова — Оренбург: ГОУ ОГУ, 2006. — 140 с.
2. Гарнаев, А.Ю. WEB - программирование на Java и Java Script. [Текст]/ А.Ю. Гарнаев -СПб.: БХВ - Петербург, 2003. - 264 с.
3. Колисниченко Д. Н. PHP 5/6 и MySQL 6. Разработка Web-приложений / Денис Колисниченко. — СПб: Вильямс, 2009. - 607 с. : ил., табл.
11. Колисниченко Д. Н. PHP 5/6 и MySQL 6. Разработка Web-приложений / Денис Колисниченко. — СПб: Вильямс, 2009. - 607 с. : ил., табл.
5. Кристиан Дари. AJAX и PHP. Разработка динамических веб-приложений. Издательство, 2006г.
6. Полякова, Л.Н. Основы SQL: Курс лекций. Учебное пособие [Текст]/ Л.Н. Полякова -- М.: ИНТУИТ.РУ, 2004. - 368 с.
7. Принципы работы современного ресторана [Электронный ресурс], Режим доступа: <https://businessman.ru/doljnostnyie-obyazannosti-menedjera-restorana-i-printsip-raboty.html>, свободный (Дата обращения:16.04.2018).
8. МакКоннел Стив. Совершенный код. — СПб.: Питер, 2005. — 860 с
9. Дейт К. Дж. Введение в системы баз данных. — 8-е изд. — М.: «Вильямс», 2006. — 1328 с
10. SQLite vs MySQL vs PostgreSQL: сравнение систем управления базами данных [Электронный ресурс]. Режим доступа: <http://devacademy.ru/posts/sqlite-vs-mysql-vs-postgresql>, свободный (Дата обращения: 23.04.2018).

ПРИЛОЖЕНИЕ

Приложение 1.

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, minimum-
scale=1.0">
  <link rel="shortcut icon" href="images/favicon.ico" type="image/x-icon">
  <link rel="icon" href="images/favicon.ico" type="image/x-icon">
  <title> Black Olives </title>
  <!-- CSS -->
  <link type="text/css" rel="stylesheet" href="css/bootstrap.min.css" />
  <link type="text/css" rel="stylesheet" href="css/owl.carousel.min.css">
  <link type="text/css" rel="stylesheet" href="css/font-awesome.css">
  <link type="text/css" href="css/animate.css" rel="stylesheet" />
  <link type="text/css" href="css/magnific-popup.css" rel="stylesheet" />
  <link type="text/css" href="css/bootstrap-datetimepicker.css" rel="stylesheet" />
  <link type="text/css" href="css/style.css?v=1.0" rel="stylesheet" />
  <link type="text/css" href="css/media.css?v=1.0" rel="stylesheet" />

</head>

<body id="mainBox" class="creamBg" data-spy="scroll" data-target="#bs-example-
navbar-collapse-1" data-offset="150">
  <div id="site-head">
    <!-- Header Section Begins -->
```

```

<header id="header" class="header-block" data-top="0" data-scroll="100">
  <div class="container-fluid">
    <div class="row">
      <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12 main-menu">
        <div class="menuBar scrollbtn">
          <nav class="navbar navbar-default">
            <div class="container-fluid">
              <div class="navbar-header">
                <button type="button" class="navbar-toggle collapsed"
data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-
expanded="false">
                  <span class="sr-only">Toggle navigation</span>
                  <span class="icon-bar"></span>
                  <span class="icon-bar"></span>
                  <span class="icon-bar"></span>
                </button>
                <div class="logo">
                  <a class="navbar-brand" href="index.html"></a>
                </div>
              </div>
            <div class="collapse navbar-collapse" id="bs-example-navbar-
collapse-1">
              <ul class="nav navbar-nav navbar-right">
                <li><a href="#banner"> Главная</a></li>
                <li><a href="#about">О нас</a></li>
                <li><a href="#our-menu">Меню</a></li>
                <li><a href="#gallery">Галерея</a></li>
                <li><a href="#reservation">Бронирование</a></li>
                <li><a href="#footer">Контакты</a></li>
              </ul>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```



```

        </div>
    </div>
</section>
<!-- Banner Section Ends -->
</div>

<div class="main">

    <!-- About Section Begins -->
    <section id="about" class="about-block pad-top-100 pad-bottom-100">
        <div class="container pr">
            <div class="row">
                <div class="col-lg-6 col-md-6 col-sm-12 col-xs-12 wow animated
fadeInLeft" data-wow-delay="0.5s">
                    <div class="about-img">
                        
                    </div>
                </div>
                <div class="col-lg-6 col-md-6 col-sm-12 col-xs-12">
                    <h2 class="block-title"> О нас <span class="side-img right-side-
img"><i></i><i></i><i></i><i></i></span></h2>
                    <h3> </h3>
                    <p> Расположенный в центре Белгорода, ресторан кавказской
кухни «Генацвале» знаменит фирменными блюдами на мангале, а также
внимательным и по-кавказски гостеприимным персоналом. По вечерам можно
услышать живое исполнение разнообразной музыки. На втором этаже есть
большой экран и мягкие диваны.

</p>

```

```

        </div>
    </div>
</div>
</section>
<!-- About Section Ends -->

<!-- Today's Special Menu Section Begins -->
<section id="special-menu" class="special-menu-block color-white parallax-fix
parallaximg-2 pad-top-100">
    <div class="container pr">
        <div class="row">
            <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
                <h2 class="block-title color-white text-align-center"> <span
class="side-img left-side-img"><em></em><em></em><em></em></span>
Специальные блюда <span class="side-img right-side-
img"><em></em><em></em><em></em></span></h2>
                <h5 class="title-caption text-align-center"> </h5>
                <div id="owl-special-menu" class="spcl-menu">
                    <div class="col-lg-4 col-md-4 col-sm-4 col-xs-12">
                        <div class="item wow animated fadeInUp" data-wow-
delay="0.5s">
                            <figure>
                                
                            </figure>
                            <div class="spcl-text">
                                <h4> <a href="#"> Стейк </a> </h4>
                                <p> Аппетитный стейк из баранины </p>
                                <div class="spcl-menu-btm">
                                    <ul class="stars">
                                        <li>

```

```

        <a href="#"> <i class="fa fa-star"></i> </a>
    </li>
    <li>
        <a href="#"> <i class="fa fa-star"></i> </a>
    </li>
    </ul>
    <h4> $17.8 </h4>
</div>
</div>
</div>
</div>
<div class="col-lg-4 col-md-4 col-sm-4 col-xs-12">
    <div class="item wow animated fadeInUp" data-wow-
delay="0.8s">
        <figure>
            
        </figure>
        <div class="spcl-text">
            <h4> <a href="#"> Пицца </a> </h4>
            <p> Неделя Итальянской пиццы</p>
            <div class="spcl-menu-btm">

```

```

<ul class="stars">
  <li>
    <a href="#"> <i class="fa fa-star"></i> </a>
  </li>
  <li>
    <a href="#"> <i class="fa fa-star"></i> </a>
  </li>
</ul>
<h4> $17.8 </h4>
</div>
</div>
</div>
<div class="col-lg-4 col-md-4 col-sm-4 col-xs-12">
  <div class="item wow animated fadeInUp" data-wow-
delay="1s">
    <figure>
      
    </figure>
    <div class="spcl-text">
      <h4> <a href="#"> Veg. roll </a> </h4>

```



```

<!-- Our Menu Section Begins -->
<section id="our-menu" class="our-menu-block pad-top-100 pad-bottom-100
mar-top-320 pr">
  <div class="container">
    <div class="row">
      <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
        <h2 class="block-title text-align-center"> <span class="side-img left-
side-img"><em></em><em></em><em></em></span> Меню <span class="side-
img right-side-img"><em></em><em></em><em></em></span></h2>
        <h5 class="title-caption text-align-center">Блюда нашего ресторана
</h5>
        <div class="our-menu-list">
          <div class="col-lg-6 col-md-6 col-sm-12 col-xs-12 no-pad-left">
            <div class="menu-list">
              <h3> ХОЛОДНЫЕ ЗАКУСКИ </h3>
              <h5> </h5>
              <ul>
                <li>
                  <figure>
                    
                  </figure>
                  <div class="menu-info">
                    <h5> Mexican Pizza </h5>
                    <h5 class="menu-price">$25</h5>
                    <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
                  </div>
                </li>
                <li>

```

```

<figure>
  
</figure>
<div class="menu-info">
  <h5> Grilled Meat </h5>
  <h5 class="menu-price">$20</h5>
  <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
</div>
</li>
<li>
  <figure>
    
  </figure>
  <div class="menu-info">
    <h5> Fried Shrimp </h5>
    <h5 class="menu-price">$18.8</h5>
    <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
  </div>
</li>
<li>
  <figure>
    
  </figure>
  <div class="menu-info">
    <h5> Cupcakes </h5>
    <h5 class="menu-price">$10</h5>
    <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>

```

```

        </div>
    </li>
</ul>
</div>
</div>
<div class="col-lg-6 col-md-6 col-sm-12 col-xs-12 no-pad-right">
    <div class="menu-list">
        <h3> Горячие блюда </h3>
        <h5> </h5>
        <ul>
            <li>
                <figure>
                    
                </figure>
                <div class="menu-info">
                    <h5> Mexican Pizza </h5>
                    <h5 class="menu-price">$25</h5>
                    <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
                </div>
            </li>
            <li>
                <figure>
                    
                </figure>
                <div class="menu-info">
                    <h5> Grilled Meat </h5>
                    <h5 class="menu-price">$20</h5>
                    <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
                </div>
            </li>
        </ul>
    </div>
</div>

```

```

        </div>
    </li>
    <li>
        <figure>
            
        </figure>
        <div class="menu-info">
            <h5> Fried Shrimp </h5>
            <h5 class="menu-price">$18.8</h5>
            <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
        </div>
    </li>
    <li>
        <figure>
            
        </figure>
        <div class="menu-info">
            <h5> Cupcakes </h5>
            <h5 class="menu-price">$10</h5>
            <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
        </div>
    </li>
</ul>
</div>
<div class="col-lg-6 col-md-6 col-sm-12 col-xs-12 no-pad-left">
    <div class="menu-list">
        <h3> Десерты </h3>

```

```

<h5> </h5>
<ul>
  <li>
    <figure>
      
    </figure>
    <div class="menu-info">
      <h5> Mexican Pizza </h5>
      <h5 class="menu-price">$25</h5>
      <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
    </div>
  </li>
  <li>
    <figure>
      
    </figure>
    <div class="menu-info">
      <h5> Grilled Meat </h5>
      <h5 class="menu-price">$20</h5>
      <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
    </div>
  </li>
</ul>
</div>
<div class="col-lg-6 col-md-6 col-sm-12 col-xs-12 no-pad-right">
  <div class="menu-list">
    <h3> Напитки </h3>

```

```

<h5> </h5>
<ul>
  <li>
    <figure>
      
    </figure>
    <div class="menu-info">
      <h5> Mexican Pizza </h5>
      <h5 class="menu-price">$25</h5>
      <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
    </div>
  </li>
  <li>
    <figure>
      
    </figure>
    <div class="menu-info">
      <h5> Grilled Meat </h5>
      <h5 class="menu-price">$20</h5>
      <h6> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed aliqua. Ut enim ad minim veniam, nostrud.</h6>
    </div>
  </li>
</ul>
</div>
</div>
</div>
</div>
</div>

```

```

    </div>
</section>
<!-- Our Menu Section Ends -->

<!-- Video Menu Section Begins -->

<!-- Video Menu Menu Section Ends -->

<!-- Everyday's Menu Section Begins -->

<!-- Everyday's Menu Menu Section Ends -->

<!-- Team Section Begins -->    <!-- Team Section Ends -->

<!-- Food Gallery Section Begins -->
<section id="gallery" class="gallery-block pad-top-100 pad-bottom-100">
    <div class="container pr">
        <div class="row">
            <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
                <h2 class="block-title text-align-center"> <span class="side-img left-
side-img"><i></i><i></i><i></i></span>    Галерея    от    шеф-повара<span
class="side-img right-side-img"><i></i><i></i><i></i></span> </h2>
                <h5 class="title-caption text-align-center"> Галерея блюд шеф-
повара нашего ресторана. Наша кухня не оставит Вас равнодушными! </h5>
                <div id="popup-gallery" class="food-gallery wow animated
fadeInUp" data-wow-delay="0.2s">
                    <div class="food-gallery-item gallery-small-height">
                        <a href="images/gallery-1.png">
                            
                            <div class="image-hover-overlay">

```

```

        <i class="fa fa-plus-circle" aria-hidden="true"></i>
    </div>
</a>
</div>
<div class="food-gallery-item wide gallery-small-height wow
animated fadeInUp" data-wow-delay="0.3s">
    <a href="images/gallery-2.png">
        
        <div class="image-hover-overlay">
            <i class="fa fa-plus-circle" aria-hidden="true"></i>
        </div>
    </a>
</div>
<div class="food-gallery-item gallery-small-height wow animated
fadeInUp" data-wow-delay="0.4s">
    <a href="images/gallery-3.png">
        
        <div class="image-hover-overlay">
            <i class="fa fa-plus-circle" aria-hidden="true"></i>
        </div>
    </a>
</div>
<div class="food-gallery-item gallery-big-height wow animated
fadeInUp" data-wow-delay="0.5s">
    <a href="images/gallery-4.png">
        
        <div class="image-hover-overlay">
            <i class="fa fa-plus-circle" aria-hidden="true"></i>
        </div>
    </a>

```

```

</div>
<div class="food-gallery-item wide gallery-big-height wow
animated fadeInUp" data-wow-delay="0.6s">
  <a href="images/gallery-5.png">
    
    <div class="image-hover-overlay">
      <i class="fa fa-plus-circle" aria-hidden="true"></i>
    </div>
  </a>
</div>
<div class="food-gallery-item gallery-small-height wow animated
fadeInUp" data-wow-delay="0.7s">
  <a href="images/gallery-6.png">
    
    <div class="image-hover-overlay">
      <i class="fa fa-plus-circle" aria-hidden="true"></i>
    </div>
  </a>
</div>
<div class="food-gallery-item gallery-small-height wow animated
fadeInUp" data-wow-delay="0.8s">
  <a href="images/gallery-7.png">
    
    <div class="image-hover-overlay">
      <i class="fa fa-plus-circle" aria-hidden="true"></i>
    </div>
  </a>
</div>
<div class="food-gallery-item gallery-small-height wow animated
fadeInUp" data-wow-delay="0.9s">

```

```

    <a href="images/gallery-8.png">
      
      <div class="image-hover-overlay">
        <i class="fa fa-plus-circle" aria-hidden="true"></i>
      </div>
    </a>
  </div>
  <div class="food-gallery-item gallery-small-height wow animated
fadeInUp" data-wow-delay="1s">
    <a href="images/gallery-9.png">
      
      <div class="image-hover-overlay">
        <i class="fa fa-plus-circle" aria-hidden="true"></i>
      </div>
    </a>
  </div>
  <div class="food-gallery-item wide gallery-small-height wow
animated fadeInUp" data-wow-delay="1.1s">
    <a href="images/gallery-10.png">
      
      <div class="image-hover-overlay">
        <i class="fa fa-plus-circle" aria-hidden="true"></i>
      </div>
    </a>
  </div>
</div>
</div>
</div>
</div>
</section>

```

```
<!-- Food Gallery Section Ends -->
```

```
<!-- Testimonial Section Begins -->
```

```
<section id="testimonial" class="testimonial-block text-align-center color-white  
parallax-fix parallaximg-5 pad-top-100 pad-bottom-100">
```

```
<div class="container pr">
```

```
<div class="row">
```

```
<div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
```

```
<h2 class="block-title text-align-center color-white"> <span  
class="side-img left-side-img"><em></em><em></em><em></em></span>  
Отзывы <span class="side-img right-side-  
img"><em></em><em></em><em></em></span></h2>
```

```
<div id="owl-testimonial" class="testimonial-slider">
```

```
<div class="item">
```

```
<figure>
```

```

```

```
</figure>
```

```
<p>Шикарная кухня и отличное обслуживание!</p>
```

```
<ul class="stars">
```

```
<li>
```

```
<a href="#"> <i class="fa fa-star"></i> </a>
```

```
</li>
```

```
<li>
```

```
<a href="#"> <i class="fa fa-star"></i> </a>
```

```
</li>
```

```
<li>
```

```
<a href="#"> <i class="fa fa-star"></i> </a>
```

```
</li>
```

```
<li>
```

```
<a href="#"> <i class="fa fa-star"></i> </a>
```

```
</li>
<li>
  <a href="#"> <i class="fa fa-star"></i> </a>
</li>
</ul>
<h5> Аля Иванова </h5>
</div>
<div class="item">
  <figure>
    
  </figure>
  <p>Приятная атмосфера, уютно и спокойно </p>
  <ul class="stars">
    <li>
      <a href="#"> <i class="fa fa-star"></i> </a>
    </li>
    <li>
      <a href="#"> <i class="fa fa-star"></i> </a>
    </li>
    <li>
      <a href="#"> <i class="fa fa-star"></i> </a>
    </li>
    <li>
      <a href="#"> <i class="fa fa-star"></i> </a>
    </li>
  </ul>
  <h5> Ирина Петровна </h5>
</div>
<div class="item">
  <figure>
```

```

        
    </figure>
    <p>Добродушный персонал и хорошая музыка </p>
    <ul class="stars">
        <li>
            <a href="#"> <i class="fa fa-star"></i> </a>
        </li>
        <li>
            <a href="#"> <i class="fa fa-star"></i> </a>
        </li>
        <li>
            <a href="#"> <i class="fa fa-star"></i> </a>
        </li>
    </ul>
    <h5> Елена Сидорова </h5>
</div>
</div>
</div>
</div>
</section>
<!-- Testimonial Section Ends -->

<!-- Blog Section Begins -->

<!-- Blog Section Ends -->

<!-- Catering Section Begins -->    <!-- Catering Section Ends -->

<!-- Reservation Section Begins -->

```

```

<section id="reservation" class="reservation-block pad-top-100 pad-bottom-
100">
  <div class="container pr">
    <div class="row">
      <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
        <h2 class="block-title text-align-center"> <span class="side-img left-
side-img"><em></em><em></em><em></em></span>          Бронирование<span
class="side-img right-side-img"><em></em><em></em><em></em></span></h2>
        <h5 class="title-caption text-align-center mar-bottom-25"> Для
Вашего удобства и экономии времени, мы предлагаем бронирование
столов</h5>
      <form class="reservation-block">
        <div class="col-lg-4 col-md-4 col-sm-4 col-xs-12">
          <div class="form-box">
            <input type="text" placeholder="Name">
          </div>
        </div>
        <div class="col-lg-4 col-md-4 col-sm-4 col-xs-12">
          <div class="form-box">
            <input type="text" placeholder="E-Mail ID">
          </div>
        </div>
        <div class="col-lg-4 col-md-4 col-sm-4 col-xs-12">
          <div class="form-box">
            <input type="text" placeholder="contact no.">
          </div>
        </div>
        <div class="col-lg-4 col-md-4 col-sm-4 col-xs-12">
          <div class="form-box">
            <select>

```

```

        <option selected disabled>Кол-во персон</option>
        <option>1</option>
        <option>2</option>
        <option>3</option>
    </select>
</div>
</div>
<div class="col-lg-4 col-md-4 col-sm-4 col-xs-12">
    <div class="form-box">
        <input type="text" id='date-picker' placeholder="Date" />
    </div>
</div>
<div class="col-lg-4 col-md-4 col-sm-4 col-xs-12">
    <div class="form-box">
        <input type="text" id='time-picker' placeholder="Time" />
    </div>
</div>
<div class="col-lg-6 col-md-6 col-sm-6 col-xs-12">
    <div class="form-box">
        <select>
            <option selected disabled>Предпочитаемая
кухня</option>
            <option>Русская</option>
            <option>Европейская</option>
            <option>Кавказская</option>
        </select>
    </div>
</div>
<div class="col-lg-6 col-md-6 col-sm-6 col-xs-12">
    <div class="form-box">

```

```

        <select>
            <option selected disabled>Случай</option>
            <option>Свадьба</option>
            <option>День рождения</option>
            <option>Годовщина</option>
        </select>
    </div>
</div>
<div class="col-lg-12 col-md-12 col-sm-12 col-xs-12 reserve-book-
btn text-align-center">
    <a href="#" class="orange-btn">Забронировать мой
столик</a>
</div>
</form>
</div>
</div>
</div>
</section>
<!-- Reservation Section Ends -->

<!-- Footer Section Begins -->
<section id="footer" class="footer-block color-white parallax-fix parallaximg-7
pad-top-100">
    <div class="container pr">
        <div class="row">
            <div class=" footer-section">
                <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12 text-align-
center">
                    <div class="newsletter">

```

```
<h2 class="footer-title color-white"><span class="side-img left-
side-img"><i></i><i></i><i></i></span> Подпишитесь<span class="side-img
right-side-img"><i></i><i></i><i></i></span> </h2>
```

```
<p> Зарегистрируйтесь за 1 минуту, и всегда будете в курсе
наших акций и новых блюд</p>
```

```
<form>
```

```
<input type="email" placeholder="Enter your e-mail id" />
```

```
<a href="#" class="orange-btn"><i class="fa fa-paper-plane-o"
aria-hidden="true"></i></a>
```

```
</form>
```

```
</div>
```

```
</div>
```

```
<div class="footer-content">
```

```
<div class="col-lg-4 col-md-4 col-sm-12 col-xs-12">
```

```
<div class="contact-us">
```

```
<h3 class="footer-title">Контакты</h3>
```

```
<ul>
```

```
<li> <i class="fa fa-map-marker"></i> <span> Белгород,
ул. Николая Островского 12 </span> </li>
```

```
<li> <i class="fa fa-phone"></i> <span> +7 (4722) 32-83-
07 </span> </li>
```

```
<li> <i class="fa fa-envelope-o"></i> <a href="#">
support@food.com </a> </li>
```

```
</ul>
```

```
</div>
```

```
</div>
```

```
<div class="col-lg-4 col-md-4 col-sm-12 col-xs-12">
```

```
<div class="opening-hours">
```

```
<h3 class="footer-title">Время работы</h3>
```

```

        <h6><em class="fa fa-calendar-minus-o" aria-
hidden="true"></em> <span>Будни <br>
        12:00 AM - 2:00 PM</span></h6>
        <h6><i class="fa fa-calendar-minus-o" aria-hidden="true"></i>
<span>Выходные<br/> 12:00 AM - 2:00 PM</span> </h6>
    </div>
</div>
<div class="col-lg-4 col-md-4 col-sm-12 col-xs-12">
    <div class="new-on-menu">
        <h3 class="footer-title">Новое в меню</h3>
        <ul>
            <li> <a href="#"> Стейк бургер </a> </li>
            <li> <a href="#">Мексиканская Курица Фахитас </a>
</li>
            <li> <a href="#"> Шоколадный кекс </a> </li>
            <li> <a href="#"> Курица Тикка Масала </a> </li>
        </ul>
    </div>
</div>
</div>
</div>
</div>
</div>
<div class="container">
    <div class="row">
        <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
            <div class="social-media-icons">
                <ul>

```

```
</li>
  <a href="#" target="_blank"> <i class="fa fa-
facebook"></i> </a>
</li>
</li>
</li>
  <a href="#" target="_blank"> <i class="fa fa-twitter"></i>
</a>
</li>
</li>
  <a href="#" target="_blank"> <i class="fa fa-google-
plus"></i> </a>
</li>
</li>
  <a href="#" target="_blank"> <i class="fa fa-linkedin"></i>
</a>
</li>
</li>
  <a href="#" target="_blank"> <i class="fa fa-youtube"></i>
</a>
</li>
</li>
</ul>
</div>
<h6 class="copy-title"> COPYRIGHT &copy; 2018 Харитонов
<a href="https://store.multidots.com/" target="_blank">Юлия</a> </h6>
</div>
</div>
</div>
</section>
</section>
<!-- Footer Section Ends -->
```

```

    <!-- Back to top Section Begins -->
    <a href="javascript:void(0);" class="back-to-top"><i class="fa fa-long-arrow-
up" aria-hidden="true"></i></a>
    <!-- Back to top Section Ends -->

</div>

<!-- JQuery Js -->
<script type="text/javascript" src="js/jquery.min.js"></script>
<script type="text/javascript" src="js/bootstrap.min.js"></script>
<script type="text/javascript" src="js/element.js"></script>
<script type="text/javascript" src="js/modernizr.js"></script>
<script type="text/javascript" src="js/owl.carousel.js"></script>
<script type="text/javascript" src="js/jquery.localscroll-1.2.7-min.js"></script>
<script type="text/javascript" src="js/wow.min.js"></script>
<script type="text/javascript" src="js/imagesloaded.pkgd.min.js"></script>
<script type="text/javascript" src="js/jquery.isotope.min.js"></script>
<script type="text/javascript" src="js/jquery.parallax-1.1.3.js"></script>
<script type="text/javascript" src="js/jquery.magnific-popup.js"></script>
<script type="text/javascript" src="js/moment-with-locales.js"></script>
<script type="text/javascript" src="js/bootstrap-datetimepicker.js"></script>
</body>
</html>

```

Приложение 2.

```

<!doctype html>
<html>
<head>

```

```

<!-- <meta charset="UTF-8">-->
<title>заказы</title>
</head>
<body>

<table align="center" bgcolor=silver width="1200">
<tr>
<td align=center> <font face=Arial size=6 color=red> </font> </td>
</tr>
<tr>
<td colspan="2" align="center" style="border-bottom:double; border-
top:double; border-color:#000">
<br>

<br><br>
</tr>
<td>
<p align=center> <font size=5 face=Arial> <b> Информация о заказах</b> </font>
</p>
<?php
require_once 'settings.php';
$sql = 'select * from TOVARI';
$res = $dbh->query($sql);
echo '<table align=center border=1>
<tr class="active">
<td align=center> <b> Номер </b> </td>
<td align=center> <b> Номер поставки </b> </td>
<td align=center> <b> Наименование </b> </td>
<td align=center> <b> Описание </b> </td>
<td align=center> <b> Наличие </b> </td>

```

```

        <td align=center> <b> Стоимость закупки </b> </td>
        <td align=center> <b> Количество </b> </td>
        <td align=center> <b> Стоимость продажи </b> </td>
        <td align=center> <b> Номер заказа </b> </td>
        <td align=center> <b> Номер поставщика </b> </td>
    </tr>;
    $result = $res->FetchAll(PDO::FETCH_NUM);
    foreach($result as $row) {
        echo "<tr>
            <td align=center>$row[0]</td>
            <td align=center>$row[1]</td>
            <td align=center>$row[2]</td>
            <td align=center>$row[3]</td>
            <td align=center>$row[4]</td>
            <td align=center>$row[5]</td>
            <td align=center>$row[6]</td>
            <td align=center>$row[7]</td>
            <td align=center>$row[8]</td>
            <td align=center>$row[9]</td>
        </tr>";
    }
    echo '</table>'
?>
<hr>
<p align=center> <font size=5 face=Arial> <b> Работа с таблицей </b> </font>
</p>

<p>&nbsp;</p>
<table width="955" border="1">

```

<p><i>Добавление заказа: </i> </p>

<form action="add_tov.php" method="post">

<p> Номер заказа

<input type="text" name="naimenovanie" size="15">

номер сотрудника

<input type="text" name="opisanie" size="15">

ФИО клиента клиента

<input type="text" name="nalichie" size="15"> </p>

<p> тел.

<input type="text" name="stoim_zak" size="15">

Дата

<input type="text" name="kolichество" size="15">

Время

<input type="text" name="stoim_prod" size="15">

e-mail

<input type="text" name="stoim_zak2" size="15">

Кухня

<input type="text" name="stoim_zak3" size="15">

Событие

<input type="text" name="stoim_zak4" size="15">

</p>

<p>

<input type="submit" value="Добавить">

</p>

</form>

<p> <i> Удаление заказа: </i> </p>

```

<form action="del_tov.php" method="post">
  <p> Введите номер заказа, который нужно удалить <input type="text"
name="delete" size="5">
  <input type="submit" value="Удалить"> </p>
</form>
<p> <i> <font size=4 face = Arial> Поиск: </font> </i> </p>
<form action="search_tov.php" method="post">
  <p> Введите информацию для поиска по наименованию <input type="text"
name="srh" size="5">
  <input type="submit" value="Найти"> </p>
</form>
</body>
</html>

```

Приложение 3.

```

<!doctype html>
<html>
<head>
<!-- <meta charset="UTF-8">-->
<title>Магазин электроники</title>
</head>
<body>
<table align="center" bgcolor=silver width="1200">
<tr>
<td align=center> </td>
</tr>
<tr>

```

```

        <td colspan="2" align="center" style="border-bottom:double; border-
top:double; border-color:#000">
        <br>
        <a href="index.html" >ГЛАВНАЯ СТРАНИЦА</a>
        <a href="tovari.php">ТОВАРЫ</a>
        <a href="zakaz.php">ЗАКАЗЫ</a>
        <a href="sotrudniki.php">СОТРУДНИКИ</a>
        <a href="postavshik.php">ПОСТАВЩИКИ</a>
        <a href="klienti.php">КЛИЕНТЫ</a>
        <br><br>
</tr>
<td>
<p align=center> <font size=5 face=Arial> <b> Информация о заказах </b>
</font> </p>
<?php
require_once 'settings.php';
$nom = $_POST["nomer"];
$npr = $_POST["nomer_prod"];
$dz = $_POST["data"];
$nok = $_POST["nomer_kl"];
$nos = $_POST["nomer_sotr"];

$sql1 = "insert into ZAKAZ values ($nom, '$npr', '$dz', '$nok', '$nos)";
$res = $dbh->exec($sql1);
$sql = 'select * from ZAKAZ';
$res = $dbh->query($sql);
echo '<table align=center border=1>
        <tr>
<td align=center> <b> Номер </b> </td>
        <td align=center> <b> Номер поставки </b> </td>

```

```

        <td align=center> <b> Наименование </b> </td>
        <td align=center> <b> Описание </b> </td>
        <td align=center> <b> Наличие </b> </td>
        <td align=center> <b> Стоимость закупки </b> </td>
        <td align=center> <b> Количество </b> </td>
        <td align=center> <b> Стоимость продажи </b> </td>
        <td align=center> <b> Номер заказа </b> </td>
        <td align=center> <b> Номер поставщика </b> </td>
        </tr>';
$result = $res->FetchAll(PDO::FETCH_NUM);
foreach($result as $row) {
    echo "<tr>
        <td align=center>$row[0]</td>
        <td align=center>$row[1]</td>
        <td align=center>$row[2]</td>
        <td align=center>$row[3]</td>
        <td align=center>$row[4]</td>

        </tr>";
    }
    echo '</table>'
?>
<hr>
<p> <a href="zakaz.php"> Назад </a> </p>
</body>
</html>

<!doctype html>

```

Приложение 4.

```

<html>
<head>
<!-- <meta charset="UTF-8">-->
<title> </title>
</head>
<body>

<table align="center" bgcolor=silver width="1200">
<tr>
<td align=center> <font face=Arial size=6 color=red> Магазин электроники
</font> </td>
</tr>
<tr>
<td colspan="2" align="center" style="border-bottom:double; border-
top:double; border-color:#000">
<br>
<a href="index.html" >ГЛАВНАЯ СТРАНИЦА</a>
<a href="tovari.php">ТОВАРЫ</a>
<a href="zakaz.php">ЗАКАЗЫ</a>
<a href="sotrudniki.php">СОТРУДНИКИ</a>
<a href="postavshik.php">ПОСТАВЩИКИ</a>
<a href="klienti.php">КЛИЕНТЫ</a>
<br><br>
</tr>
<td>
<p align=center> <font size=5 face=Arial> <b> Информация о товарах </b>
</font> </p>
<?php
require_once 'settings.php';
$del = $_POST["delete"];

```

```

$sql1 = "delete from ZAKAZ where NOMER=$del";
$res = $dbh->exec($sql1);
$sql = 'select * from ZAKAZ';
$res = $dbh->query($sql);
echo '<table align=center border=1>
    <tr>
<td align=center> <b> Номер </b> </td>
    <td align=center> <b> Номер поставки </b> </td>
    <td align=center> <b> Наименование </b> </td>
    <td align=center> <b> Описание </b> </td>
    <td align=center> <b> Наличие </b> </td>
    <td align=center> <b> Стоимость закупки </b> </td>
    <td align=center> <b> Количество </b> </td>
    <td align=center> <b> Стоимость продажи </b> </td>
    <td align=center> <b> Номер заказа </b> </td>
    <td align=center> <b> Номер поставщика </b> </td>

</tr>';
$result = $res->fetchAll(PDO::FETCH_NUM);
foreach($result as $row) {
    echo "<tr>
        <td align=center>$row[0]</td>
        <td align=center>$row[1]</td>
        <td align=center>$row[2]</td>
        <td align=center>$row[3]</td>
        <td align=center>$row[4]</td>

</tr>";
    }
echo '</table>'

```

```
?>
<hr>
<p> <a href="zakaz.php"> Назад </a> </p>
</body>
</html>
```

Приложение 5.

```
<!doctype html>
<html>
<head>
<!-- <meta charset="UTF-8">-->
<title> </title>
</head>
<body>

<table align="center" bgcolor=silver width="1200">
<tr>
<td align=center> <font face=Arial size=6 color=red> </font> </td>
</tr>
<tr>
<td colspan="2" align="center" style="border-bottom:double; border-
top:double; border-color:#000">
<br>
<a href="index.html" >ГЛАВНАЯ СТРАНИЦА</a>
<a href="tovari.php">ТОВАРЫ</a>
<a href="zakaz.php">ЗАКАЗЫ</a>
<a href="sotrudniki.php">СОТРУДНИКИ</a>
<a href="postavshik.php">ПОСТАВЩИКИ</a>
<a href="klienti.php">КЛИЕНТЫ</a>
```

```

        <br><br>
</tr>
<td>
<p align=center> <font size=5 face=Arial> <b> Информация о товарах </b>
</font> </p>
<?php
require_once 'settings.php';
$srh = $_POST["srh"];
$sql = "select * from ZAKAZ where NOMER_PROD like '%$srh%'";
$res = $dbh->query($sql);
echo '<table align=center border=1>
    <tr>
<td align=center> <b> Номер </b> </td>
    <td align=center> <b> Номер поставки </b> </td>
    <td align=center> <b> Наименование </b> </td>
    <td align=center> <b> Описание </b> </td>
    <td align=center> <b> Наличие </b> </td>
    <td align=center> <b> Стоимость закупки </b> </td>
    <td align=center> <b> Количество </b> </td>
    <td align=center> <b> Стоимость продажи </b> </td>
    <td align=center> <b> Номер заказа </b> </td>
    <td align=center> <b> Номер поставщика </b> </td>

</tr>';
$result = $res->fetchAll(PDO::FETCH_NUM);
foreach($result as $row) {
    echo "<tr>
        <td align=center>$row[0]</td>
        <td align=center>$row[1]</td>

```

```

        <td align=center>$row[2]</td>
        <td align=center>$row[3]</td>
        <td align=center>$row[4]</td>

    </tr>";
    }
    echo '</table>'
?>
<hr>
<p> <a href="zakaz.php"> Назад </a> </p>
</body>
</html>

```

Приложение 6.

register.php

```

<?
// Страница регистрации нового пользователя
# Соединяюсь с БД
mysql_connect("localhost", "myhost", "myhost");
mysql_select_db("testtable");
if(isset($_POST['submit']))
{
    $err = array();
    # проверяю логин
    if(!preg_match("/^[a-zA-Z0-9]+$/", $_POST['login']))
    {
        $err[] = "Логин может состоять только из букв английского алфавита и цифр";
    }
}

```

```

}
if(strlen($_POST['login']) < 3 or strlen($_POST['login']) > 30)

{
    $err[] = "Логин должен быть не меньше 3-х символов и не больше 30";
}
# проверяем, не существует ли пользователя с таким именем
$query = mysql_query("SELECT COUNT(user_id) FROM users WHERE user_login='".mysql_real_escape_string($_POST['login'])."'");
if(mysql_result($query, 0) > 0)
{
    $err[] = "Пользователь с таким логином уже существует в базе данных";
}
# Если нет ошибок, то добавляем в БД нового пользователя
if(count($err) == 0)
{

    $login = $_POST['login'];           # Убираем лишние пробелы и делаем двойное шифрование
    $password = md5(md5(trim($_POST['password'])));
    mysql_query("INSERT INTO users SET user_login='".$login."', user_password='".$password.'");
    header("Location: login.php"); exit();
}
else
{
    print "<b>При регистрации произошли следующие ошибки:</b><br>";
    foreach($err AS $error)
    {
        print $error."<br>";
    }
}

```

```

    }
}
?>
<form method="POST">
Логин <input name="login" type="text"><br>
Пароль <input name="password" type="password"><br>
<input name="submit" type="submit" value="Зарегистрироваться">
</form>

```

Приложение 7.

login.php

```

<?
// Страница авторизации
# Функция для генерации случайной строки
function generateCode($length=6) {
    $chars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
    $code = "";
    $clen = strlen($chars) - 1;
    while (strlen($code) < $length) {
        $code .= $chars[mt_rand(0,$clen)];
    }
    return $code;
}
# Соединяюсь с БД
mysql_connect("localhost", "myhost", "myhost");
mysql_select_db("testtable");
if(isset($_POST['submit']))

```

```

{

# Вытаскиваем из БД запись, у которой логин равняется введенному
$query = mysql_query("SELECT user_id, user_password FROM users WHERE u
ser_login='".mysql_real_escape_string($_POST['login'])."' LIMIT 1");
$data = mysql_fetch_assoc($query);

# Соавниваем пароли
if($data['user_password'] === md5(md5($_POST['password'])))
{
# Генерируем случайное число и шифруем его
$hash = md5(generateCode(10));
if(!@$_POST['not_attach_ip'])
{
# Если пользователя выбрал привязку к IP
# Переводим IP в строку
$sinsip = ", user_ip=INET_ATON('".$_SERVER['REMOTE_ADDR']."'");
}
# Записываем в БД новый хеш авторизации и IP
mysql_query("UPDATE users SET user_hash='".$hash.'" ".$sinsip." WHERE us
er_id='".$data['user_id']."'");

# Ставим куки
setcookie("id", $data['user_id'], time()+60*60*24*30);
setcookie("hash", $hash, time()+60*60*24*30);
# Переадресовываем браузер на страницу проверки нашего скрипта
header("Location: check.php"); exit();
}
else
{

```

```

        print "Вы ввели неправильный логин/пароль";
    }
}

?>
<form method="POST">
Логин <input name="login" type="text"><br>
Пароль <input name="password" type="password"><br>
Не прикреплять к IP(не безопасно) <input type="checkbox" name="not_attach_ip"
><br>
<input name="submit" type="submit" value="Войти">
</form>

```

Приложение 8.

check.php

```

<?
// Скрипт проверки
# Соединяюсь с БД
mysql_connect("localhost", "myhost", "myhost");
mysql_select_db("testtable");
if (isset($_COOKIE['id']) and isset($_COOKIE['hash']))
{
    $query = mysql_query("SELECT *,INET_NTOA(user_ip) FROM users WHERE
user_id = '".intval($_COOKIE['id'])."' LIMIT 1");
    $userdata = mysql_fetch_assoc($query);
    if((($userdata['user_hash'] !== $_COOKIE['hash']) or ($userdata['user_id'] !== $_C
OOKIE['id'])<br> or ((($userdata['user_ip'] !== $_SERVER['REMOTE_ADDR']) an
d ($userdata['user_ip'] !== "0"))))

```

```
{
    setcookie("id", "", time() - 3600*24*30*12, "/");
    setcookie("hash", "", time() - 3600*24*30*12, "/");
    print "Хм, что-то не получилось";
}
else
{
    print "Привет, ".$userdata['user_login'].". Всё работает!";
}
}
else
{
    print "Включите куки";
}

?>
```