

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Тольяттинский государственный университет»

Кафедра _____ Прикладная математика и информатика _____
(наименование)

_____ 09.03.03 Прикладная информатика _____
(код и наименование направления подготовки / специальности)

_____ Разработка программного обеспечения _____
(направленность (профиль) / специализация)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)**

на тему: «Разработка программного обеспечения информационной системы учета
заявок салона красоты»

Обучающийся _____ М.И. кызы Давудова _____
(Инициалы Фамилия) (личная подпись)

Руководитель _____ д.т.н, доцент, С.В. Мкртычев _____
(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Аннотация

Тема бакалаврской работ «Разработка программного обеспечения информационной системы учета заявок салона красоты».

Объектом исследования бакалаврской работы является процесс учета заявок салона красоты.

Предметом исследования является информационная система учета заявок салона красоты.

Цель выпускной квалификационной работы – разработка информационной системы учета заявок салона красоты.

Практическая значимость бакалаврской работы заключается в разработке информационной системы салона красоты для автоматизации учета заявок.

Первая глава посвящена анализу предметной области автоматизации и постановки задачи разработки информационной системы учета заявок салона красоты, как веб-ориентированного приложения.

Вторая глава посвящена проектированию информационной системы веб-приложения.

В третьей главе описан процесс программной реализации веб-приложения.

В заключении описываются результаты выполнения выпускной квалификационной работы.

Бакалаврская работа состоит из 53 страницы текста, 32 рисунков, 20 источников.

Оглавление

Введение.....	4
Глава 1 Постановка задачи на разработку программного обеспечения информационной системы учета заявок салона красоты	6
1.1 Описание задачи автоматизации учета заявок.....	6
1.2 Обзор и анализ аналогов	8
1.3 Описание архитектуры веб-приложения.....	11
1.4 Разработка требований к программному обеспечению	13
Глава 2 Проектирование программного обеспечения информационной системы учета заявок салона красоты.....	17
2.1 Варианты использования системы.....	17
2.2 Проектирование веб-страниц.....	21
2.3 Проектирование табличных форм.....	26
2.4 Проектирование интерактивных форм	27
2.5 Проектирование базы данных	28
Глава 3 Реализация и тестирование программного обеспечения информационной системы учета заявок салона красоты	30
3.1 Методы создания веб-страниц.....	30
3.2 Структура папок и файлов веб-приложения	33
3.3 Взаимодействие скриптов веб-приложения.....	36
3.4 Разработка базы данных	38
3.5 Управление доступом пользователей	42
3.6 Описание работы приложения.....	43
Заключение	51
Список используемой литературы и используемых источников.....	53

Введение

Салоны красоты, которые предоставляют услуги населению, всегда стремятся максимально расширить клиентскую базу, завоевать и удержать постоянных клиентов. Компании используют все возможные технологии, способствующие решению этой проблемы.

Лучшим и сравнительно недорогостоящим инструментом рекламы услуг компании 24 часа в сутки является наличие представительства компании в сети Интернет, к которому постоянно имею доступ, как постоянные клиенты компании, так и потенциальные клиенты, который в полном объеме информирует посетителей об услугах, ценах, достижениях компании.

Однако в современных условиях тотального использования информационно-коммуникационных технологий и сети интернет, наличие простого, информационного сайта является недостаточным условием успеха компании. Программное приложение должно обладать определенными интерактивными свойствами, позволяющими пользователям общаться с сотрудниками компании, задавать вопросы, оставлять свои отзывы, делать заявки онлайн на услуги компании.

Отсюда вытекает актуальность темы исследования выпускной квалификационной работы – программная реализация информационной системы учета онлайн заявок клиентов компании на услуги салона красоты.

Объект исследования выпускной квалификационной работы – информационная система учета заявок салона красоты.

Предмет исследования – программное обеспечение информационной системы учета заявок салона красоты.

Цель работы – разработка программного обеспечения информационной системы учета заявок салона красоты.

Задачи поставленные и решенные при выполнении ВКР:

- постановка задачи на разработку программного обеспечения информационной системы учета заявок салона красоты;
- проектирование программного обеспечения информационной системы учета заявок салона красоты;
- реализация и тестирование программного обеспечения информационной системы учета заявок салона красоты.

В рамках работы применялись такие методы исследования, как анализ литературных источников, аналогов систем, экспериментальная проверка и анализ средств разработки, выбор средств разработки веб-приложения.

Практическая значимость работы заключается в том, что разработанное веб-приложение можно практически использовать в деятельности любого салона красоты, как для привлечения клиентов, так и для увеличения потока заказов за счет удобного сервиса онлайн заявок через сеть интернет.

Работа состоит из введения, основной части и заключения. Во введении поставлена цель ВКР, описана актуальность проблемы создания веб-ориентированной системы.

Первая глава посвящена анализу предметной области автоматизации и постановки задачи разработки информационной системы учета заявок салона красоты, как веб-ориентированного приложения.

Вторая глава посвящена проектированию информационной системы веб-приложения.

В третьей главе описан процесс программной реализации веб-приложения.

В заключении описываются результаты выполнения выпускной квалификационной работы.

Бакалаврская работа состоит из 53 страниц текста, 32 рисунков, 20 источников.

Глава 1 Постановка задачи на разработку программного обеспечения информационной системы учета заявок салона красоты

1.1 Описание задачи автоматизации учета заявок

В компании «Салон Красоты» имеет несколько салонов, расположенных по всему городу. Миссия компании заключается в максимальном удовлетворении потребностей своих клиентов. Клиенты должны получать удовольствие от процесса и результата обслуживания.

Это главная по своим возможностям и прибыли сеть в городе. Все подразделения предоставляют самые широкие возможности в сфере парикмахерских и косметологических услуг.

Проведение мужских и женских стрижек мастерами высокого класса. Стрижки могут быть самыми простыми и для юбилеев и свадеб, с окраской, сочетать клонирование, мелирование, элиминирование и лечение волос. Салон занимается наращиванием ногтей, проведением процедур депиляции, наращивает шелковые ресницы, услуги солярия. Клиент обрабатывается отрядом профессиональных косметологов, массажистов, мастеров маникюра и искусства и педикюрными мастерами на постоянной основе. Компания использует самые современные выпуски оборудования косметологической индустрии, имеет команду из специально подготовленных высококвалифицированных специалистов.

Для расширения клиентской базы, удобства коммуникации сотрудников с клиентами, руководство компании вкладывает средства для создания и совершенствование сайта с целью развития функциональных возможностей веб-приложения, таких, как онлайн заказы услуг специалистов компании, рассчитывая на привлечение еще большего количества

потребителей услуг компании, обеспечивая их удобными средствами коммуникации с сотрудниками для заказа услуг посредством сети интернет в любое время и из любого места.

Руководством поставлена задача, состоящая в разработке веб-ориентированной информационной системы, по сути интерактивного сайта, который позволит клиентам заказывать услуги через сеть интернет, в любом месте локации и в любое время. Сотрудники компании накапливая эти данные, просматривают портфель заказов, принимая решения об их реализации. Это вызывает постановку задачи разработки централизованной базы данных [9] и программной среды, реализующей удобный веб-интерфейс работы с системой для различных специалистов, пользователей системы – посетителей, клиентов и сотрудников.

Составляющие организационной структуры управления компанией.

Компания относится к числу мелких и средних компаний, включающих в среду управления подразделения, типичные и для крупных компаний, включая отделы. Организационная структура компании показана на рисунке 1.



Рисунок 1 – Организационная структура салона красоты

Локальная сеть салона подключена к Интернету.

1.2 Обзор и анализ аналогов

Даже используя самый простой сайт, компания может эффективно продвинуться в сфере организации и продажи товаров и услуг. Пройдя по страницам веб-сайта, потенциальный покупатель может понять, насколько интересен для него ресурс, чтобы сделать заказ на товары и услуги.

Страница сайта – это файл текстового типа, «одетый» посредством языков HTML, CSS, таким образом, чтобы информация была представлена в удобной и красивой форме. Страница может формироваться программами, называемыми скриптами, реализующими все функции, логику сайта. Файлы отправляются сервером приложения, реализуя эту логику, загружая на компьютер пользователя с браузером, интерпретирующим результат в виде HTML, CSS уже в окончательном виде на средства отображения (монитор, экран КПК) информацию, заложенную в файле в приемлемом виде [3], [14].

Главной целью создания сайта любого типа является желание владельцев компании разработчика получить дополнительную прибыль посредством оперативной публикации данных о своих услугах.

Сайты простейшего типа, это сайты-визитки, являющиеся для компаний первой пробой при выходе компаний, предприятий, организаций в сеть интернет.

«Следующим типом сайтов являются интернет – витрины, которые выставляя упомянутый ранее контент, характерный для сайтов визиток, могут публиковать еще каталоги продукции, товаров, услуг компании.

Интернет - магазины во многом похожи на Интернет – витрины, но ключевыми их свойствами являются присутствие интерактивных подсистем, используя которые, клиенты могут в режиме онлайн формировать заказы-заявки, помещаемые в личные корзины, тележки.

Корпоративные сайты выполняют сразу множество функций, к которым относятся, как функции сайта-визитки, электронного магазина, каталога» [11].

Порталы содержат все изложенные функции, позволяющие организовывать большинство эффективных взаимодействий сотрудников компании и пользователей, клиентов компании.

Существует множество различных методов при создании веб-приложений, сайтов. Самый прямолинейный метод, это верстка html, css страниц вручную, используя специализированные редакторы. Динамические страницы разрабатываются на скриптовых языках программирования, интерпретируемых на стороне сервера. Другим направлением в способах разработки сайтов, связано с использованием специальных программ («движков»), которые предназначены для быстрой разработки. Общее название таких программ, это «системами управления контентом».

Рассмотрим несколько типовых решений для салонов красоты, которые применяв настоящее время. Результаты исследования по особенностям сайтов представлены в таблице 1.

В результате небольшая часть сайтов имеют функцию онлайн-записи, но и те отличаются неудобной формой записи.

Недостаток в том, что, заходя на сайт, клиент не должен испытывать затруднение в поиске информации и оформлении заказа, это должно делаться минимальным количеством усилий и тратой минимального количества времени заказчиком.

Среди положительных особенностей лидирует современный приятный дизайн, встречающийся на 50% анализируемых сайтов.

Чтобы понять, какие разделы нужно включить в проектируемый сайт, нужно также получить представление о разделах на проанализированных веб-сайтах и их частоте применения, используя информацию из таблицы 1. обратить внимание:

- на сайте должна быть реализована продуманная и логичная навигация;
- сайт должен содержать умеренное количество разделов;
- дизайн должен быть современным и не мешать воспринимать информацию;
- веб-приложение должно обладать удобным сервисом онлайн заказов на услуги с минимальным количеством действий пользователя (буквально в один клик по услуге).

Таблица 1 – Результаты сравнения аналогов

Название	Ссылка	Разделы	Возможность онлайн-записи	Достоинства	Недостатки
Марлен	salon-marlen.ru	Услуги Акции О нас Вакансии Контакты	есть	Современный приятный дизайн Удобная навигация	Работает с ошибками
Шоколад	chocolate40.ru	О нас Новости Подарочные сертификаты Услуги	нет		Дизайн выполнен в разных стилях Неудачное дизайнерское решение Отсутствие информации об услугах;
Версаль	versabeauty.ru	О нас Услуги Сертификат Акции Фотографии Контакты	нет		Неудачное дизайнерское решение Неудобная навигация.

Hair Lab	hairlab-kaluga.ru	О нас Мастера Услуги Стоимость Материалы Фотографии Контакты	нет	Удобная навигация	Неудачное дизайнерское решение
----------	-------------------	--------------------------------------------------------------------------------	-----	-------------------	--------------------------------

В данном проекте представлены простые решения, состоящие в том, каким образом, используя только интегрированных средств, используя только базовые средства и языки программирования создать динамический сайт, обеспечивающий эффективную функцию заказа услуг посредством интернет-приложения.

1.3 Описание архитектуры веб-приложения

Любая информационная система (ИС) поддерживает стандартные процессы получения и обработки информации:

- процессы получения информации из внешних источников;
- организацию хранения информации в системе;
- процессы обработки информации в соответствии с рядом запросов;
- процессы поиска информации и представления информации пользователям системы.

Все процессы обеспечиваются Программное обеспечение системы обеспечивает выполнение всех этих процессов. Важной составляющей программного обеспечения, поддерживающего сбор и выдачу информации, является интерфейс пользователей системы.

Работа информационных систем базируется на использовании баз данных (БД) и систем управления базами данных (СУБД) [4]. Под базой данных понимается совокупность взаимосвязанной формализованной в виде

модели информации, которая используется в режиме одновременного доступа многими пользователями и программами [6].

Информационная система получает «информационные потоки», преобразует их в «информационные потоки» необходимые пользователям, выполняя ряд операций с данными.

Клиентская часть, это браузер пользователя, которым отправляются запросы и получаются сообщения сервера (рисунок 2).

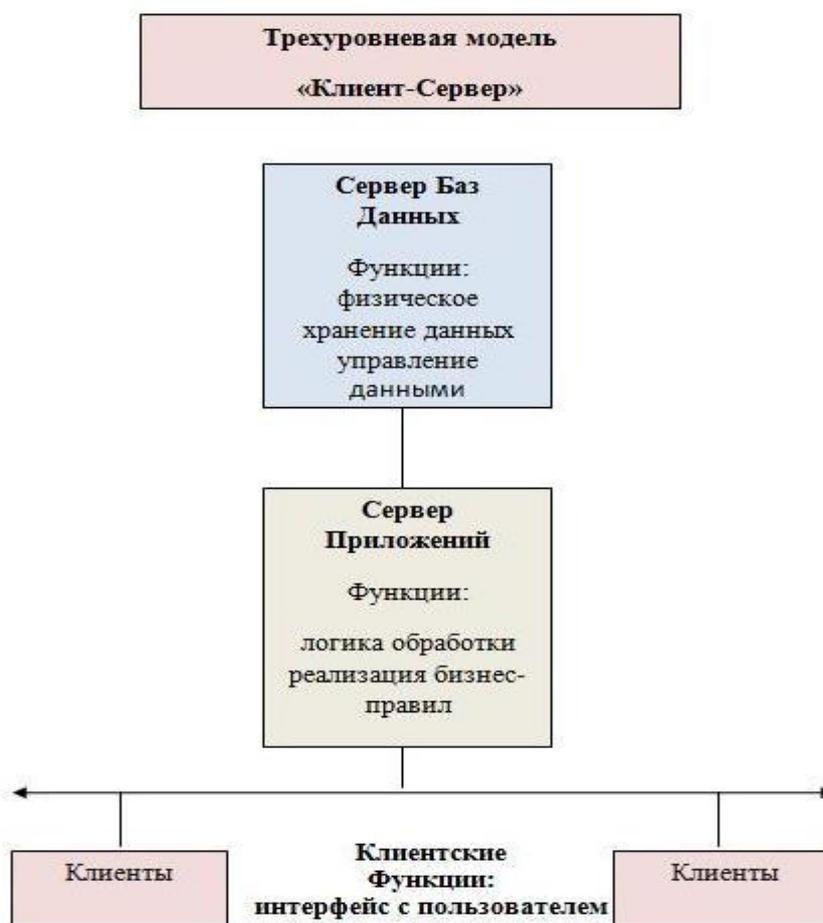


Рисунок 2 – Трехуровневое клиент-серверное приложение

Веб-ориентированная информационная система, это веб-приложение, в котором конкретно воплощается идея клиент-серверной архитектуры в среде интернет.

1.4 Разработка требований к программному обеспечению

Цель выпускной квалификационной работы является проектирование и программная реализация веб-ориентированной системы подачи и учета онлайн заявок на услуги салона.

Перечень функциональных требований [20] к веб-приложению:

- обеспечить все операций управления данными - ввод, обновление, удаление, добавление, поиск, учитывая права различных групп пользователей;
- обеспечить интуитивно-понятный интерфейс пользователей системы при выдаче информации на средства отображения, при интерактивных запросах пользователей;
- обеспечит функции регистрации и авторизации для групп пользователей, лежащие в основе разделения доступа;
- обеспечить разделение прав доступа групп пользователей посредством личных кабинетов пользователей.

Требования к поиску информации. ИС должна поддерживать поиск, добавление, обновление, удаление данных по задаваемым ключам поиска при операциях добавления, обновления, удаления услуг, заказов-заявок на услуги, оборудования салона.

Экранные формы поддерживаются в среде операционной системы Windows.

Требования к пользовательскому интерфейсу. Система должна поддерживать интуитивно-понятный интерфейс, который обеспечит простоту и привычность работы с данными.

Требования обеспечения защиты от несанкционированных операций. Получение доступа к некоторым функционалам системы, части базы данных, предваряется авторизацией пользователей, вход осуществляется по логинам и паролям зарегистрированных пользователей системы.

Требования по методам обеспечения сохранности данных. Для обеспечения сохранности данных должна быть организована процедура резервного копирования данных в определенное время.

Требования к клиентской части программного обеспечения. Клиентская часть программного обеспечения работает в среде Windows.

Серверная часть программного обеспечения должна работать на хостинге под управлением операционной системы Linux.

Требования к хостингу. Хостинг должен включать использование PHP, Apache и MySQL. Таким образом, полный набор программного обеспечения разработки и поддержки системы может быть представлена схемой, изображенной на рисунке 3. Это детализация клиент серверной архитектуры, изображенной на рисунке 2.

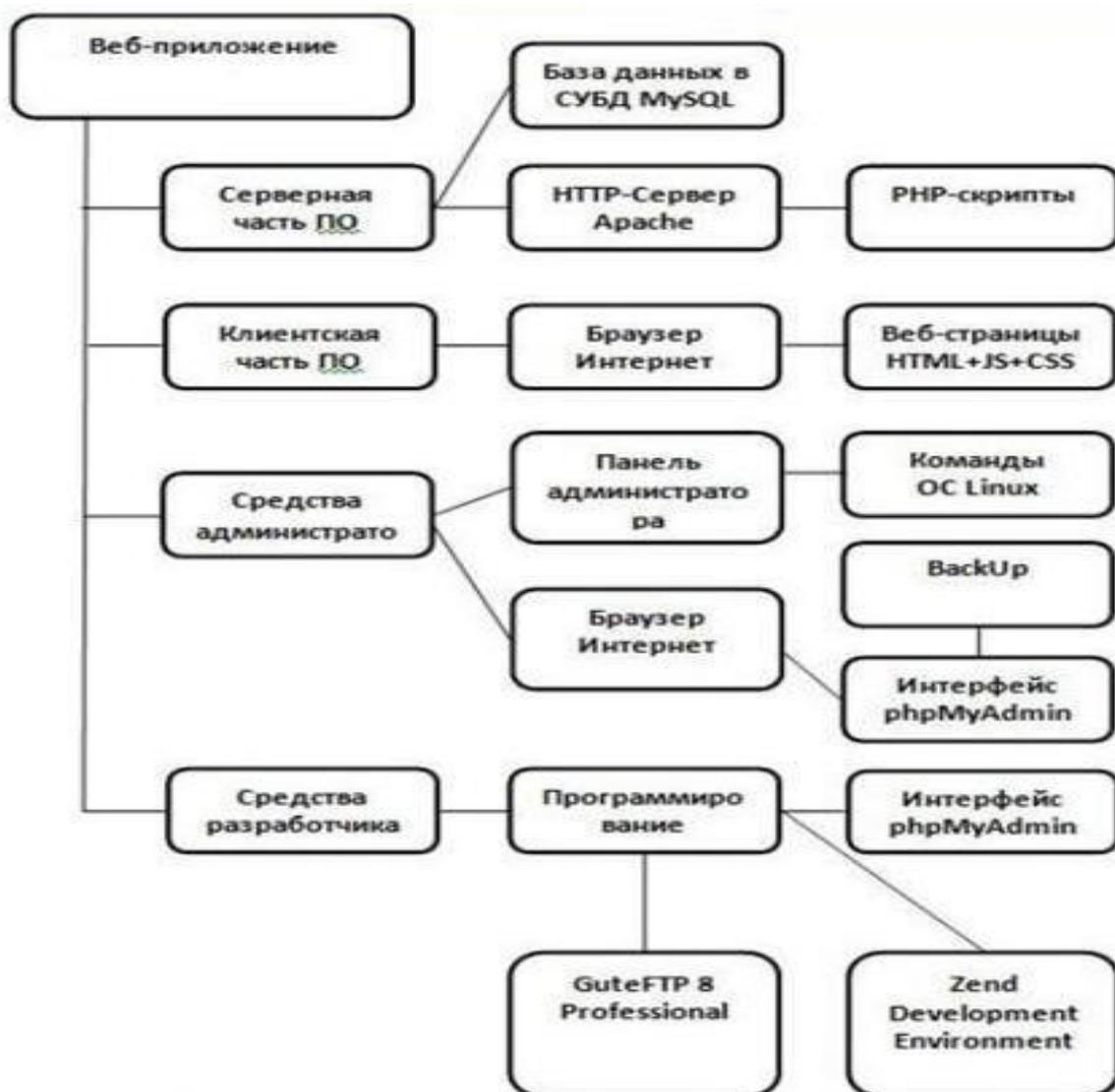


Рисунок 3 – Средства разработки и поддержки

Языки программирования серверной части программного обеспечения PHP, SQL. Языки программирования клиентской части HTML, CSS, JavaScript.

Администратором системы должны выполняться следующие функции:

- контроль регистрации и авторизации пользователей в плане корректности данных;
- контроль функций зарегистрированных пользователей;
- управляет данными базы данных веб-приложения посредством личного кабинета и веб-интерфейса администратора;

- помещает и обновляет на сайте информацию общего характера по деятельности салона.

В системе предполагается большой объем информации, которая должна подвергаться всестороннему анализу посредством использования запросов различного типа.

Поэтому в системе должна использоваться одна из промышленных систем управления базами данных (СУБД).

Глава 2 Проектирование программного обеспечения информационной системы учета заявок салона красоты

2.1 Варианты использования системы

Приведем диаграммы вариантов использования системы [12]. Это общепринятые методы представления функций пользователей в виде UML диаграмм вариантов использования [13].

Функции пользователей информационной системы состоят в следующем. Посетители, это пользователи, которые не зарегистрированы в системе, не являются клиентами компании.

Им доступен только просмотр информации, представленной в свободном доступе в системе, данные по компании, по услугам. Они должны зарегистрироваться в системе, чтобы делать заказ.

Когда пользователь-посетитель заполнит форму регистрации с логином и паролем, он становится клиентом компании.

Пользователь – клиент компании, приобретает доступ к личному кабинету.

Личный кабинет регламентирует функции этой группы пользователей.

Клиент может зайти в личный кабинет, выбрать услугу, затем оформить заказ.

Диаграмма представлена на рисунке 4.

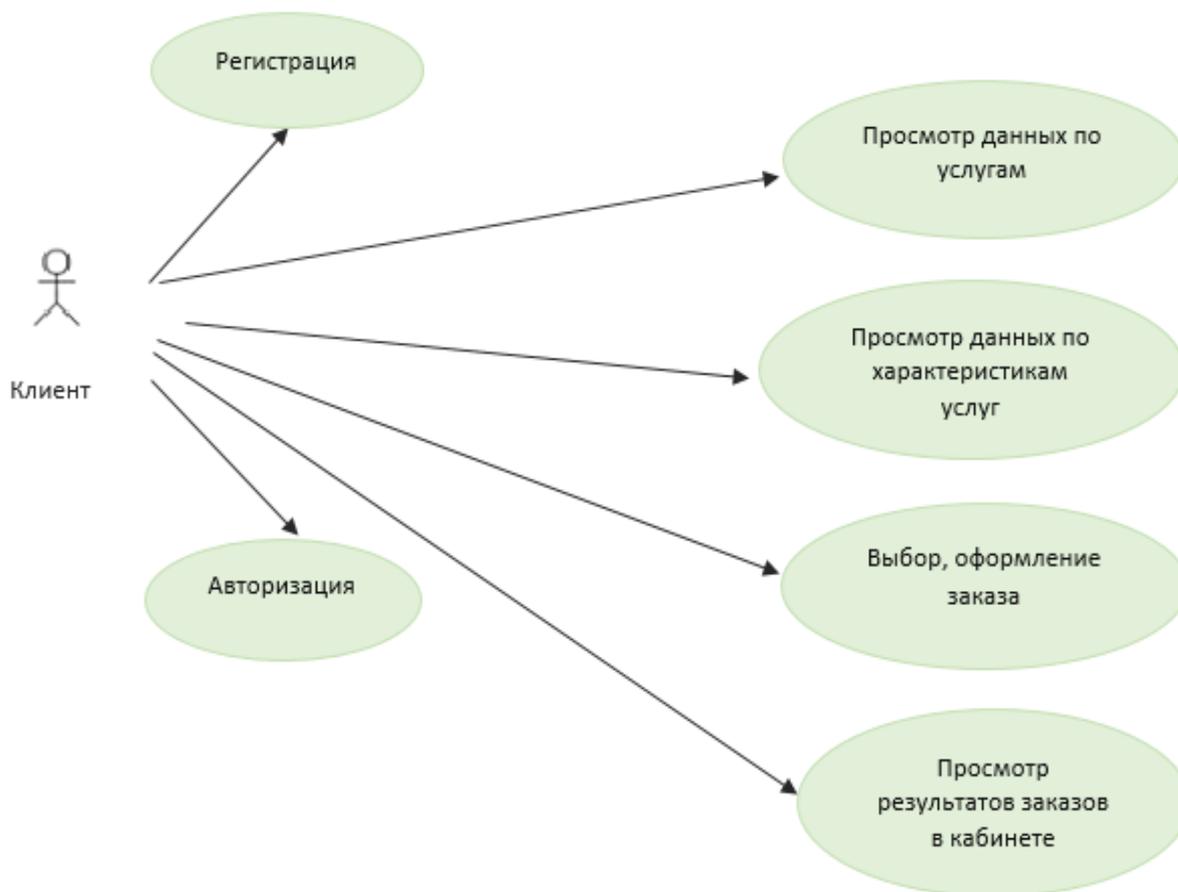


Рисунок 4 – Диаграмма вариантов использования системы клиентом

Следующую группу составляют сотрудники компании, в том числе менеджеры компании. Они также зарегистрированы в системе с логином и паролем, могут заходить в личный кабинет сотрудника.

Сотрудники, менеджеры определяют обработку заказов, влияющую на выполнение заказов в указанные сроки и с назначением ответственных сотрудников. Сотрудники, менеджеры компании, работающие с заказами, согласовывают и заполняют даты для выполнения заказов, согласовывая их с ответственными мастерами за выполнение заказов.

Для пользователя администратора доступны все операции над данными в системе. Он имеет права на добавление, удаление, обновление данных. Однако, по-видимому, следует ограничить действия по управлению заказами, чтобы избежать непредуманных действий и неопределенности в авторстве

изменений. Администратор системы может параметры поиска и получать отчетную информацию (рисунок 5).



Рисунок 5 – Диаграмма вариантов использования системы сотрудником

Пользователи клиенты входят в систему, как зарегистрированные пользователи систему, входят в систему с теми логином и паролем, который выбирают сами при регистрации.

Из предложенных услуг выбирают необходимый. Административный персонал, это также зарегистрированные в системе с определенными логинами и паролями пользователей.

Им доступны операции по изменению регламентированных данных в системе, таких как, новые услуги, характеристики существующих услуг (рисунок 6).

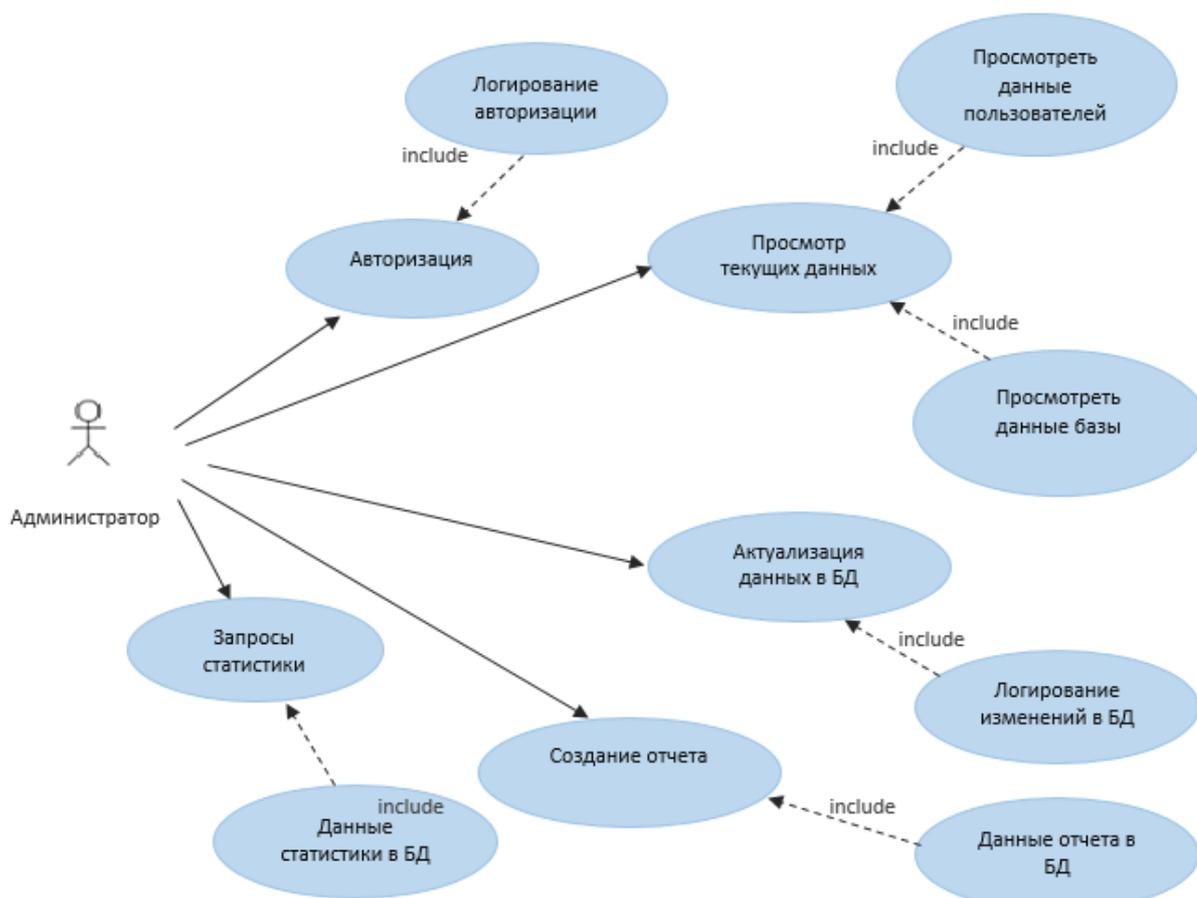


Рисунок 6 – Диаграмма использования системы администратором

Если описывать функции в общем, данная группа пользователей может следующее:

- добавлять, удалять, обновлять данные по характеристикам услуг, данные сотрудников, выполнению заказов (архивация данных);
- обеспечивают работоспособность системы в целом, надежность и безопасность системы и базы данных;
- обеспечивают сохранность базы данных посредством периодического резервного копирования базы данных.

Таким образом, различные функции системы могут использоваться регламентированным образом различными группами пользователей.

2.2 Проектирование веб-страниц

Структуру веб-страниц сайта можно представить в виде дерева функций, определяемых главным меню, изображенного на рисунке 7 [10].

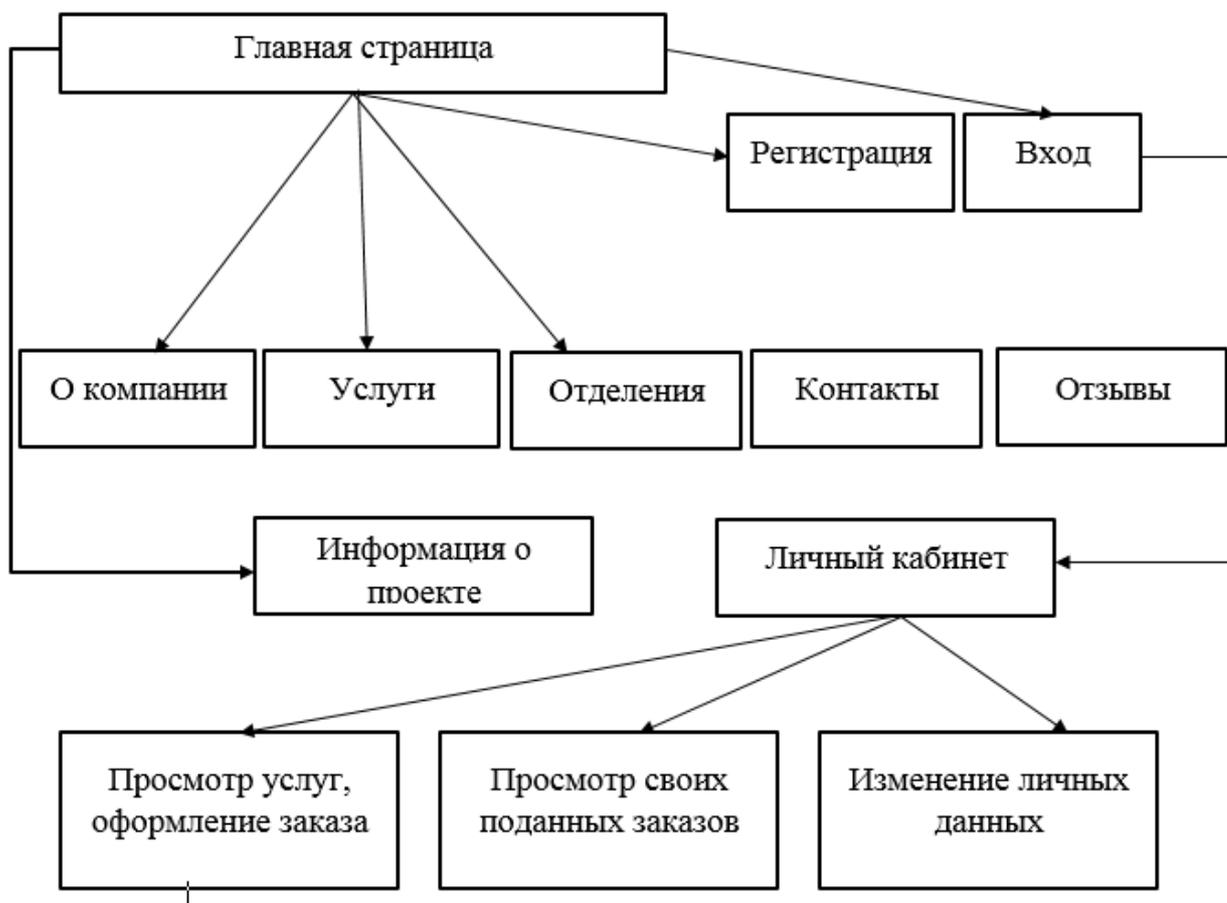


Рисунок 7 – Структура веб-приложения

Смысл вкладок выполняемых функции представлен в схеме на рисунке 8.

Здесь указаны наиболее типичные функции, свойственные для компаний, предоставляющих услуги.



Рисунок 8 – Схема, описывающая смысл вкладок меню

При нажатии вкладки происходит обращение к веб-странице, которые могут иметь статический характер или являться результатом отработки скриптов, например, когда динамически формируется список услуг.

На рисунке 9 представлены те головные модули (скрипты), которые при этом загружаются и выполняются.



Рисунок 9 – Схема, описывающая смысл вкладок меню

Для динамического формирования веб-страниц используется язык программирования PHP. Скрипты выполняются сервером приложения, реализуя логику необходимых функций. Каждый скрипт, это отдельный файл, который при выполнении может включать несколько скриптов из папки, в которой находятся все скрипты системы [19].

Таким образом, когда работают скрипты, общую схему взаимодействия скриптов, которые реализуют логику приложения и сервера можно представить в виде схемы, представленной на рисунке 10.

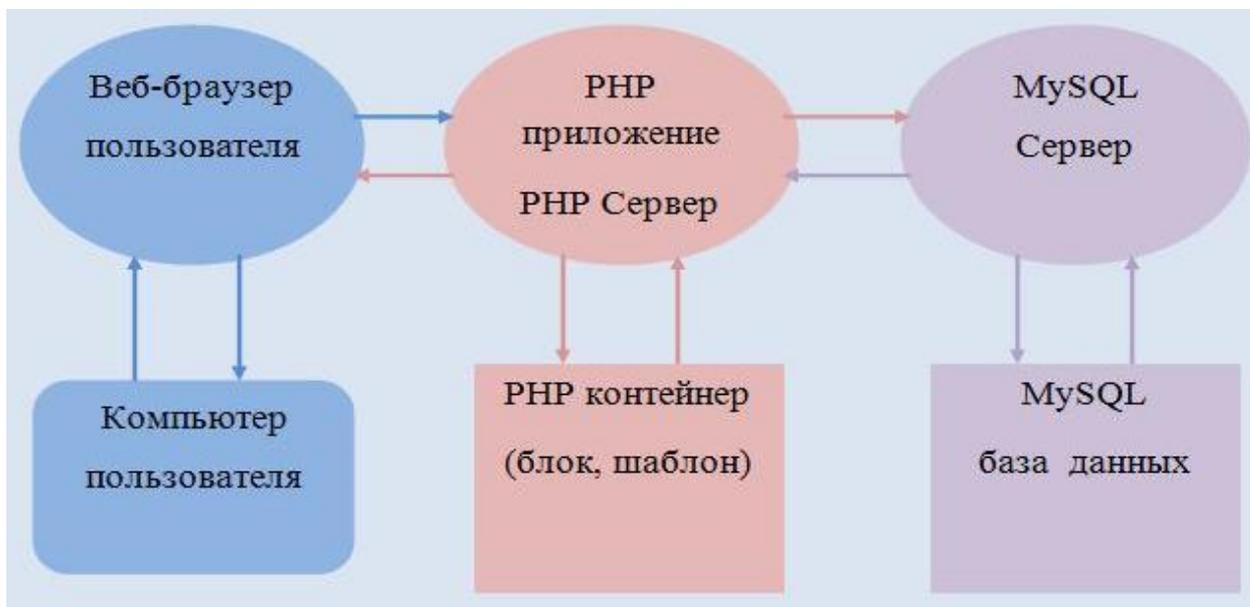


Рисунок 10 – Схема при динамическом формировании контента веб-страниц

На главная страница сайта включает логотип или характерное изображение, характеризующее деятельность компании, название компании, главное навигационное меню, ссылку на начало регистрации и ссылку на аутентификацию (вход для зарегистрированных пользователей)

Структура типичной веб-страницы показана на рисунке 11. Она включает "заголовок", который имеет наименование Header.php и статический или динамический контент.

Динамический контент генерируется PHP-скриптами и, возможно (хотя и необязательно), обращением SQL-запросами к базе данных системы.



Рисунок 11 – Типичная структура веб-страницы

Каждой вкладке навигационного меню соответствует обработка статической страницы или определенного функционального модуля системы. Вид главной страницы системы показан на рисунке 12. Вкладка «О ПРОЕКТЕ», это краткое описание проекта и использованных программных средств разработки.

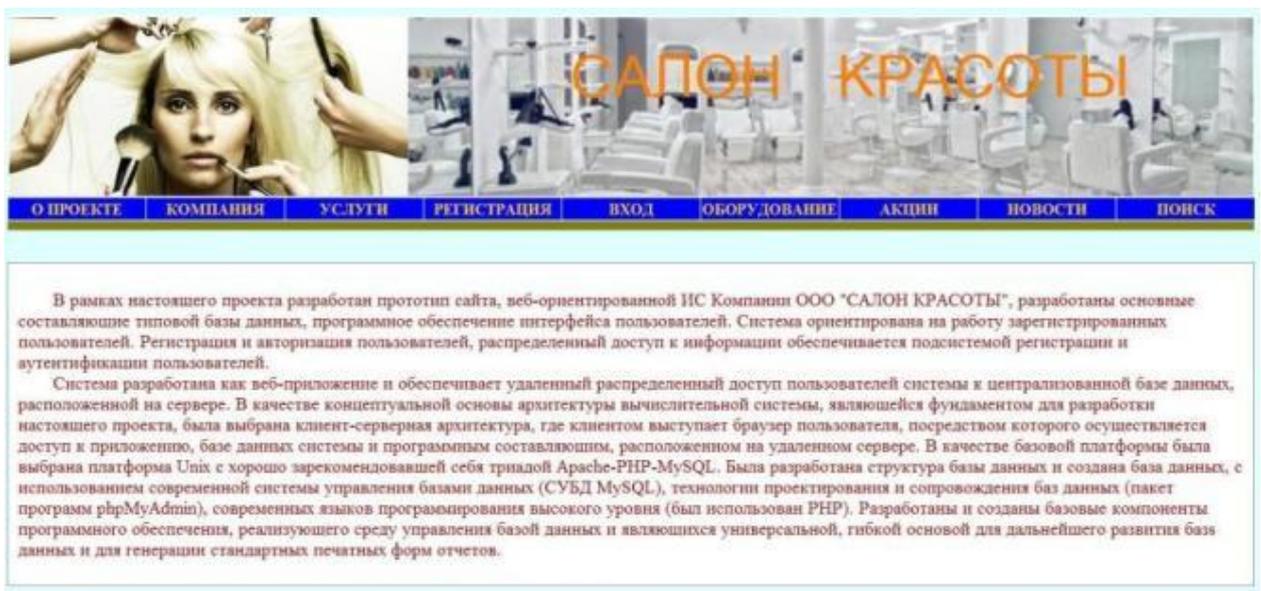


Рисунок 12 – Главная страница сайта

Вкладка «КОМПАНИЯ», это описание назначения и принципов работы КОМПАНИИ. Затем идет вкладка с описанием услуг – «УСЛУГИ». Далее вкладки «РЕГИСТРАЦИЯ» и «ВХОД» (авторизация) для зарегистрированных пользователей (рисунок 13).



Рисунок 13 – Страница входа в систему

При нажатии вкладки «РЕГИСТРАЦИЯ», пользователь переходит к последовательным шагам регистрации в системе. Здесь дается краткое описание, как проводится регистрация и зачем это делается.

2.3 Проектирование табличных форм

В информационных системах часто информация представляется в виде таблиц, которые формируются системой на основе данных, содержащихся в базе данных, поэтому необходимо спроектировать вид этих таблиц.

Для данной системы такие таблицы типичны по своему виду – это столбцы с наименованиями на русском языке, соответствующие доменам таблиц в базе данных.

Данные из базы помещаются в «обертку» из HTML-CSS кода для представления таблиц в наглядной, привычной для пользователей форме. Пример такой таблицы приведен на рисунке 14 - оборудование косметологии.

САЛОН КРАСОТЫ

О ПРОЕКТЕ | КОМПАНИЯ | УСЛУГИ | РЕГИСТРАЦИЯ | ВХОД | **ОБОРУДОВАНИЕ** | АКЦИИ | НОВОСТИ | ПОИСК

ОБОРУДОВАНИЕ КОСМЕТОЛОГИИ

Код оборудования	Наименование	Описание/Характеристика	Описание срока службы	Единица измерения	Количество всего
3000000004	Эпилятор OXC "Zenit" АЕПР56	революционная технология удаления волос, которая оказывает быстрый и заметный эффект	2016	шт	28
3000000005	Эпилятор "Zenit" АЕПР56	сочетает в себе лазерные технологии и преимущества метода пульсирующего света (IPL)	2017	шт	28
3000000006	Массажер "Zenit" АЕПР56	Аппараты для коррекции фигуры и похудения	2017	шт	28
3000000008	Система для удаления волос ICE SHR Medite 1	Принцип действия аппарата ЭЛОС (IPL+RF) MED 100С основан на универсальном сочетании 2 видов активных энергий - See more	2018	шт	28
3000000010	Аппарат для фотопластики IPL KES MED 100	профессиональный аппарат для работы в салонах красоты или косметологическом кабинете.	2018	шт	34
3000000012	Машинка для стрижки "Bosh" WSDF - 1567	1500 оборотов 100вт	2018	шт	20
3000000013	Устройство для удаления родинок TRM-220	профессиональный аппарат для работы в салонах красоты или косметологическом кабинете.	2017	шт	1500

Введите ID код для удаления, поиска или обновления:

[Посмотр структуры всех таблиц](#) [Добавить новую запись](#)

Рисунок 14 – Пример табличного представления информации из базы данных

Программное обеспечение системы, которое реализует какое представление таблиц, не зависит от количества и содержания доменов.

После выдачи всей таблицы, система добавляет кнопки управления содержимы, которые позволят добавлять строки в таблицу, обновлять выбранную строку данных и соответственно и в базе данных, удалять строки в таблицы. Таким образом реализуется табличный веб-интерфейс с базой данных системой.

2.4 Проектирование интерактивных форм

Любое веб-приложение должно поддерживать формы ввода-вывода данных для взаимодействия с системой и формирования запросов к ней.

Каждой таблице базы данных соответствует форма, которая формируется системой, а именно php-скриптами системы динамическим способом, используя данные php-блока (в системе он назван Table_Headers.php), шаблонов отображения форм и их содержимого. Заказ-заявка на услугу, подаваемый зарегистрированным пользователем системы, имеет вид, представленный на рисунке 15.

Наименование полей	Значимые или обновляемые данные
Код услуги	4000000001
Наименование услуги	
Описание/Характеристика	Стрижка, укладка, "Алена Австрийская"
Единица измерения	2
Стоимость	2000
Комментарий	Для постоянных клиентов скидка до 20%
<input type="button" value="Добавить"/>	<input type="button" value="Удалить"/>
<input type="button" value="Изменить"/>	<input type="button" value="Просмотр"/>

Рисунок 15 – Форма заявки на услугу

Аналогичные формы генерируются системой для добавления и обновления всех данных в таблицы базы данных администратором.

2.5 Проектирование базы данных

Для визуального проектирования базы данных использовалась программа MySQL Workbench. Первым шагом при создании логической модели БД является построение ER – диаграммы, включающей сущности, атрибуты и взаимосвязи. ER-диаграмма базы данных имеет имя control_salon и изображена на рисунке 16 [15].

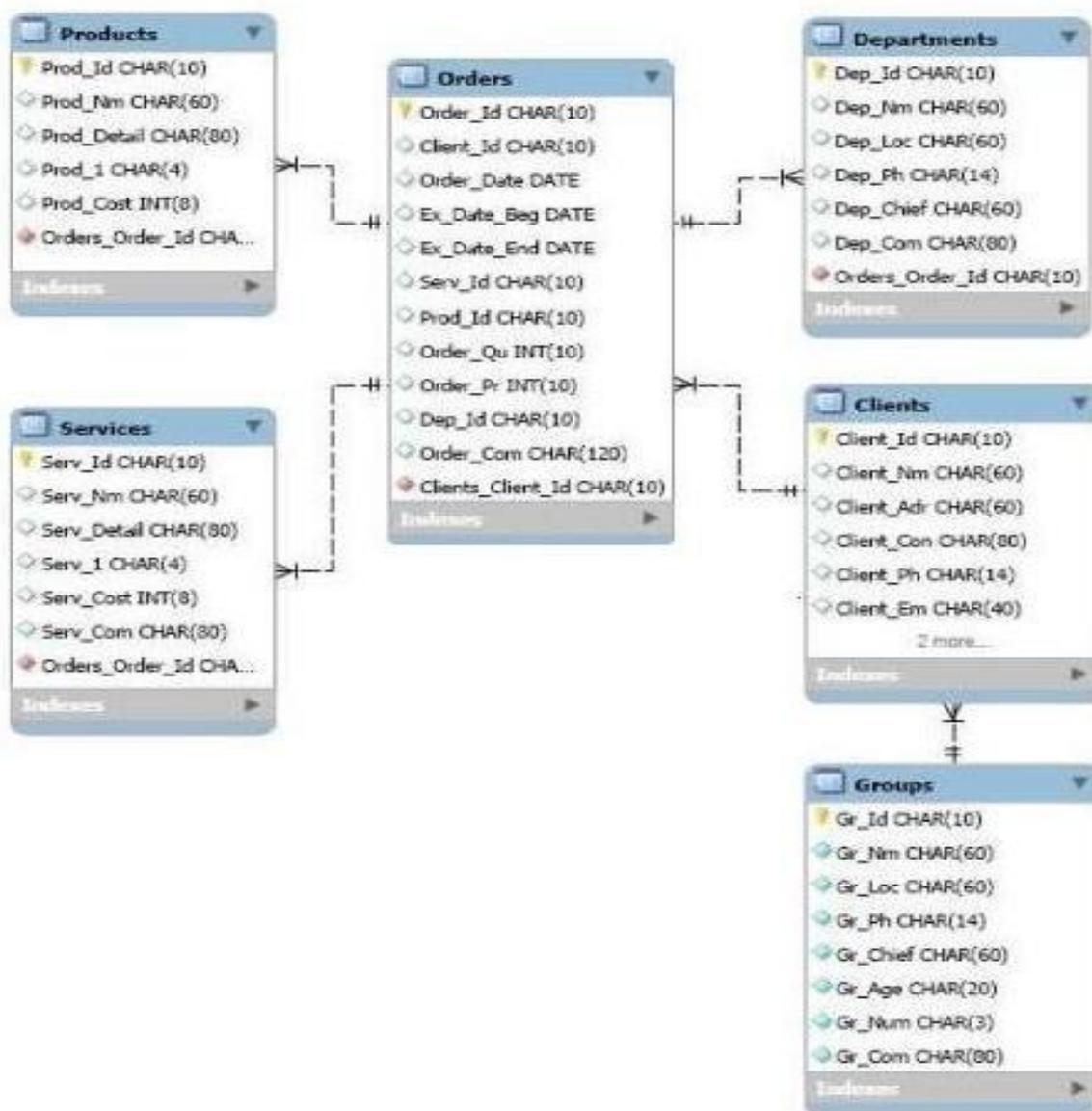


Рисунок 16 – Модель данных ИС

Показанная на рисунке 16 физическая модель базы данных устанавливает требования к структуре таблиц и связей.

Глава 3 Реализация и тестирование программного обеспечения информационной системы учета заявок салона красоты

3.1 Методы создания веб-страниц

В индустрии веб-разработок имеется множество методов разработки сайтов, веб-приложений. Самый базовый метод заключается в ручной верстке-html-css кода и программирование алгоритмов проекта на серверных языках программирования без использования шаблонов и наработанных готовых модулей, предоставляемых в рамках систем управления контентом.

Чаще всего в настоящее время используются «движки» управления контентом, которые существенно сокращают время для веб-приложений, и которые являются очень популярными сегодня, это инструменты, называемые «системы управления содержимым или контентом» (Content Management Systems, CMS).

В данном проекте не использовались CMS, которые безусловно сокращают время разработки, но накладывают определенные условия, ограничения на используемые методы и программные системы. Было решено использовать в проекте базовое программное обеспечение в виде языков разметки и языков разработки приложений, работающих на стороне сервера, где, собственно, реализуется вся логика работы системы.

Языки разработки веб-приложения.

«HTML (Hyper Text Markup Language), это язык разметки гипертекстовых документов [18]. HTML (HyperText Markup Language) - это базовый язык, который используется для создания структуры и содержимого веб-страниц. В контексте сайта салона красоты, HTML играет роль каркаса, на котором строится весь визуальный и информационный контент. Язык HTML используется для структурирования страниц, отображения и форматирования контента» [17].

«PHP. PHP – язык программирования для разработки сценариев, работающих на стороне сервера, использующий SQL-запросы для обращения к базе данных. Модули на php выполняют определенные в алгоритмах действия и в результате формируют веб-страницы, отправляемые от сервера приложения к браузеру пользователя» [2].

«Исследования говорят, что 76,4% всех сайтов в мире работает на PHP. PHP используется в самых различных видах интернет-ресурсов:

Лендинги и блоги. Позволяет создавать контент и управлять им: публиковать и редактировать статьи, изображения и комментарии.

Интернет-магазины, онлайн-платформы для продажи товаров и услуг. PHP позволяет сделать так, чтобы можно было добавлять товары в корзину, оформлять заказы и обрабатывать платежи.

Социальные сети. PHP отвечает за авторизацию, обмен сообщениями, создание профилей и управление контентом пользователей.

Веб-приложения. Например, для CRM-системы с помощью PHP можно создать виджет для добавления контакта пользователя.

Системы управления контентом. Например, на PHP написана платформа Wordpress.» [18].

«Чаще всего PHP используют для написания скриптов, которые работают на стороне сервера. Это позволяет обрабатывать данные форм, которые заполняет посетитель сайта, отсылать и принимать cookies. Возможности языка достаточно широки. PHP совместим с большинством современных баз данных, как с реляционными, такими как MySQL и PostgreSQL, так и с нереляционными, например, MongoDB, Redis.

В разработке систем управления контентом PHP используют, чтобы расширять функциональность систем. Большие возможности обеспечивает PHP при работе с FTP-серверами – приложениями, которые позволяют хранить и передавать файлы. С помощью PHP можно отправить изображения на сервер, загрузить обновлённый файл на сайт, удалить и переименовать

его. С помощью PHP можно также изменить права доступа к учётным данным и этим обеспечить их безопасность.

Также данный язык эффективно работает с cookie - файлами, которые собирают данные пользователей: количество посещений, переходы между страницами и т. д. С помощью PHP можно легко считать cookie-файл из браузера.

Конкуренты PHP более универсальны: например, Java можно использовать для разработки ПО на уровне операционной системы, Python – для машинного обучения, а Golang – для параллельной обработки нескольких процессов в одном приложении. Преимущества PHP перед другими языками – широкая известность и большая библиотека фреймворков» [18].

«Мощность языка увеличивается при использовании объектно-ориентированных средств PHP, реализуемым понятием класс php» [16].

При разработке сайта салона красоты PHP поможет в создании форм для обратной связи, заказа услуг, записи на прием и онлайн-консультаций.

PHP хорошо сочетается с различными системами управления базами данных (например, MySQL, PostgreSQL). Это позволяет хранить информацию, например, о клиентах, услугах и расписании.

PHP может использоваться для обработки онлайн-заказов, оплаты услуг и интеграции с платежными системами.

PHP поможет создать административную панель для управления содержимым сайта, добавления новых услуг, управления расписанием и отслеживания заказов.

Этот язык предоставляет множество возможностей для создания функционального и привлекательного сайта салона красоты, обеспечивая возможность легко управлять контентом, взаимодействовать с клиентами и обеспечивать безопасность данных.

«Все файлы веб-приложения, реализованные на HTML, PHP-SQL и расположены в папке ControlSalon.

Web-сервер Apache, это веб-сервер, реализующий запросы к веб-страницам, файлам системы.

SQL (Structured Query Language), это язык запросов к реляционным базам данных, обеспечивающий взаимодействие скриптов с базой данных посредством сервера баз данных, в данном случае, это MySQL-сервер» [1].

MySQL – многопользовательский, многопоточный сервер базы данных SQL [7].

3.2 Структура папок и файлов веб-приложения

Все папки и файлы веб-приложения находятся в папке ControlSalon. Структура папок и файлов представлена на рисунке 17.

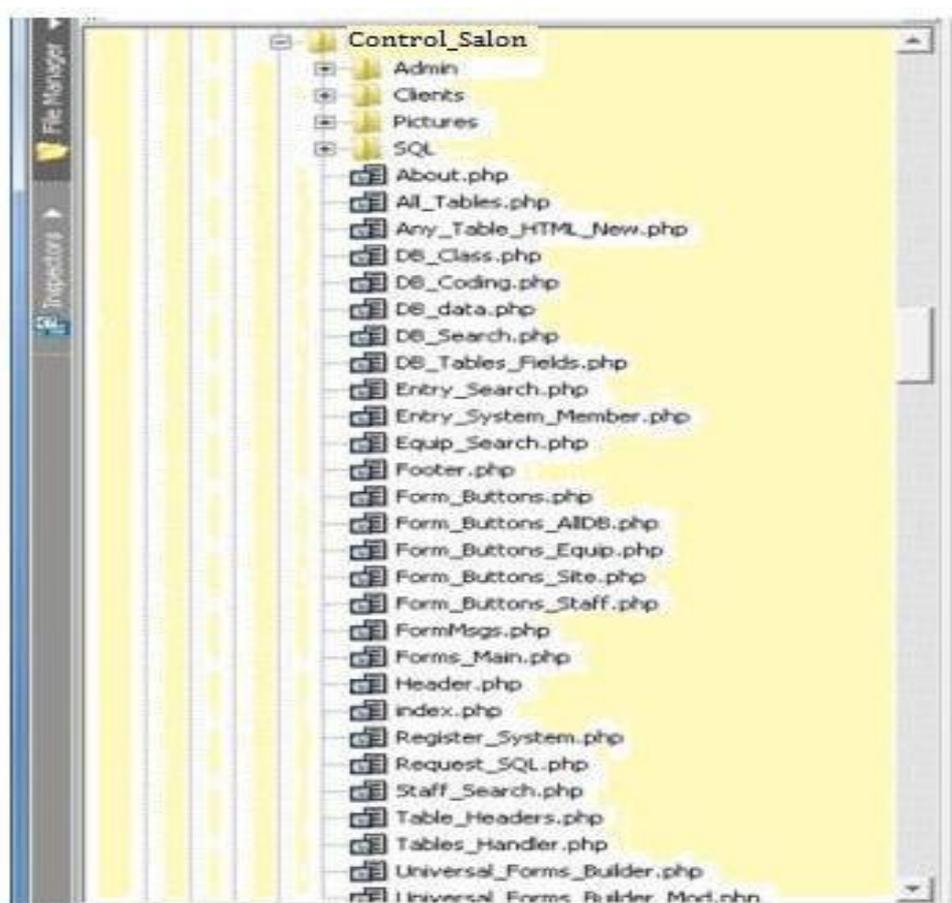


Рисунок 17 – Структура папок и файлов

Состав файлов системы, назначение и выполняемые функции представлены на рисунке 18.

📁 Control_Salon	Главная папка, содержащая все модули
📁 All_Tables_short.php	Главная страница с таблицами БД
📁 Any_Table_HTML_New.php	Класс таблиц отображения
📁 DB_Class.php	Класс работы с базой данных
📁 DB_Coding.php	Модуль перекодировки в кириллицу
📁 DB_data.php	Параметры соединения с БД
📁 Form_Buttons.php	Формы удаления, поиска, обновления
📁 Form_Buttons_Staff.php	Форма поиска по нескольким ключам
📁 FormMsgs.php	Сообщения
📁 Forms_Main.php	Форма добавления данных в таблицы
📁 Header.php	Стандартная шапка страниц
📁 Staff_Handler.php	Обработчик БД
📁 Staff_Search.php	Модуль поиска по нескольким ключам
📁 Table_Headers.php	Файлы настройки «шапок» таблиц
📁 Table_Headers_DB.php	Файлы настройки полей таблиц БД
📁 Universal_Forms_Builder.php	Построитель форм добавления

Рисунок 18 – Наименования и назначение файлов (модулей) системы

Разработан класс `DB_Class`, который состоит из методов, выполняющих запросы к базе данных системы.

При запросе к базе данных возвращается ассоциированный массив данных в случае поиска данных или код возврата об успешности удаления или добавлении данных. Фрагмент кода в интерпретации Notepad++ представлен на рисунке 19.

```
DB_Class.php x
1 <?php
2 class MySQL_DB
3 {
4     private $dbObj = null;
5     private $result = null;
6
7     /* Для конструктора - адрес, имя пользователя, пароль, имя базы данных, порт,
8     /*а также кодировку для соединения.
9     /* По умолчанию используется utf8 */
10
11     public function __construct($host, $user, $password, $base, $port = null, $charset = 'utf8')
12     {
13         $this->dbObj = new mysqli($host, $user, $password, $base, $port);
14         $this->dbObj->set_charset($charset);
15     }
16
17     /*основная и единственная функция, которая выполняет запрос и возвращает результат его работы*/
18
19     public function query($query)
20     {
21         if(!$this->dbObj)
22             return false;
23
24         /*очищаем предыдущий результат*/
25
26         if(is_object($this->result))
27             $this->result->free();
28
29         /*выполняем запрос*/
30
31         $this->result = $this->dbObj->query($query);
32
33         /*если есть ошибки - выводим их*/
34
35         if($this->dbObj->errno)
36             die("mysqli error #".$this->dbObj->errno." : ".$this->dbObj->error);
37         return $this->dbObj->error;
38     }
39 }
```

Рисунок 19 – Фрагмент кода класса DB_Class

Класс Any_Table_HTML.php формирует табличное представление данных из базы.

«Класс состоит из методов, которые получают данные из таблиц базы данных, формируют ассоциированный массив, который затем помещается в «обертку» HTM-CSS для привычного отображения таблиц на экране пользователя» [17]. Фрагмент кода представлен на рисунке 20.

```
1 <?php
2 class Any_Table
3 {
4     var $table_array = array();
5     var $headers      = array();
6     var $domns;
7     function Any_Table ($headers)
8     {
9         $this->headers = $headers;
10        $this->domns   = count($headers);
11    }
12
13    function AddRow ($row)
14    {
15        If ( count ($row) != $this->domns)
16            return false;
17        array_push( $this -> table_array, $row);
18        return true;
19    }
20
21    function SetArr($Arr)
22    {
23        $this -> table_array = $Arr;
24        return true;
25    }
26
27    function AddRowAssArr ($row_ass)
28    {
29        $row=array();
30        foreach ($this -> headers as $header)
31        { if (!isset($row_ass[$header]))
32            $row_ass[$header]='';
33            $row[ ]=$row_ass[$header];
34        }
35        array_push( $this -> table_array, $row);
36        return true;
37    }
38
39    function PrintArr ()
40    {
41        print "<pre>";
42        foreach ($this -> headers as $header)
```

Рисунок 20 – Фрагмент кода Any_Table_HTML

За счет описанного приема реализовано отображение таблиц независимо от содержания и наименования столбцов.

3.3 Взаимодействие скриптов веб-приложения

Каждый модуль веб-приложения включает несколько PHP скриптов, выполняемых сервером. Результатом является HTML, CSS, JavaScript веб-страница, отправляемая браузеру пользователя.

Верхние уровни являются основными, выполняющими поставленную перед ними общую задачу, например, отображение на экране пользователя данных в табличном виде (рисунок 21).

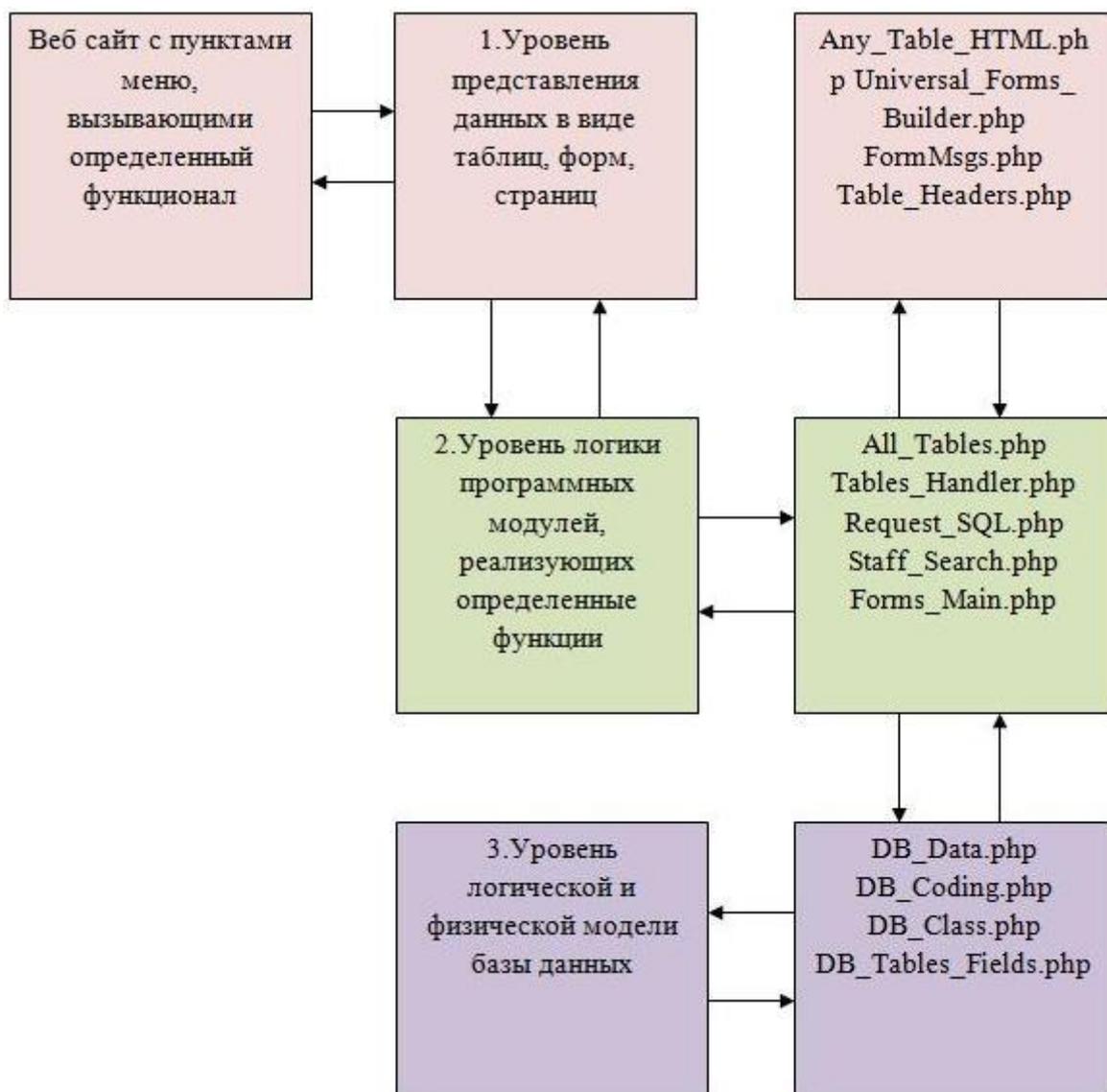


Рисунок 21 – Структурная схема взаимодействия скриптов

Для извлечения данных используются при этом скрипты работы с базой данных нижнего уровня, в частности, как уже было показано, DB_Class и DB_data.php, который содержит параметры соединения с базой данных. Алгоритм [5], [8] поиска и извлечения данных из базы данных (рисунок 22) предусматривает передачу имени таблицы в качестве параметра).

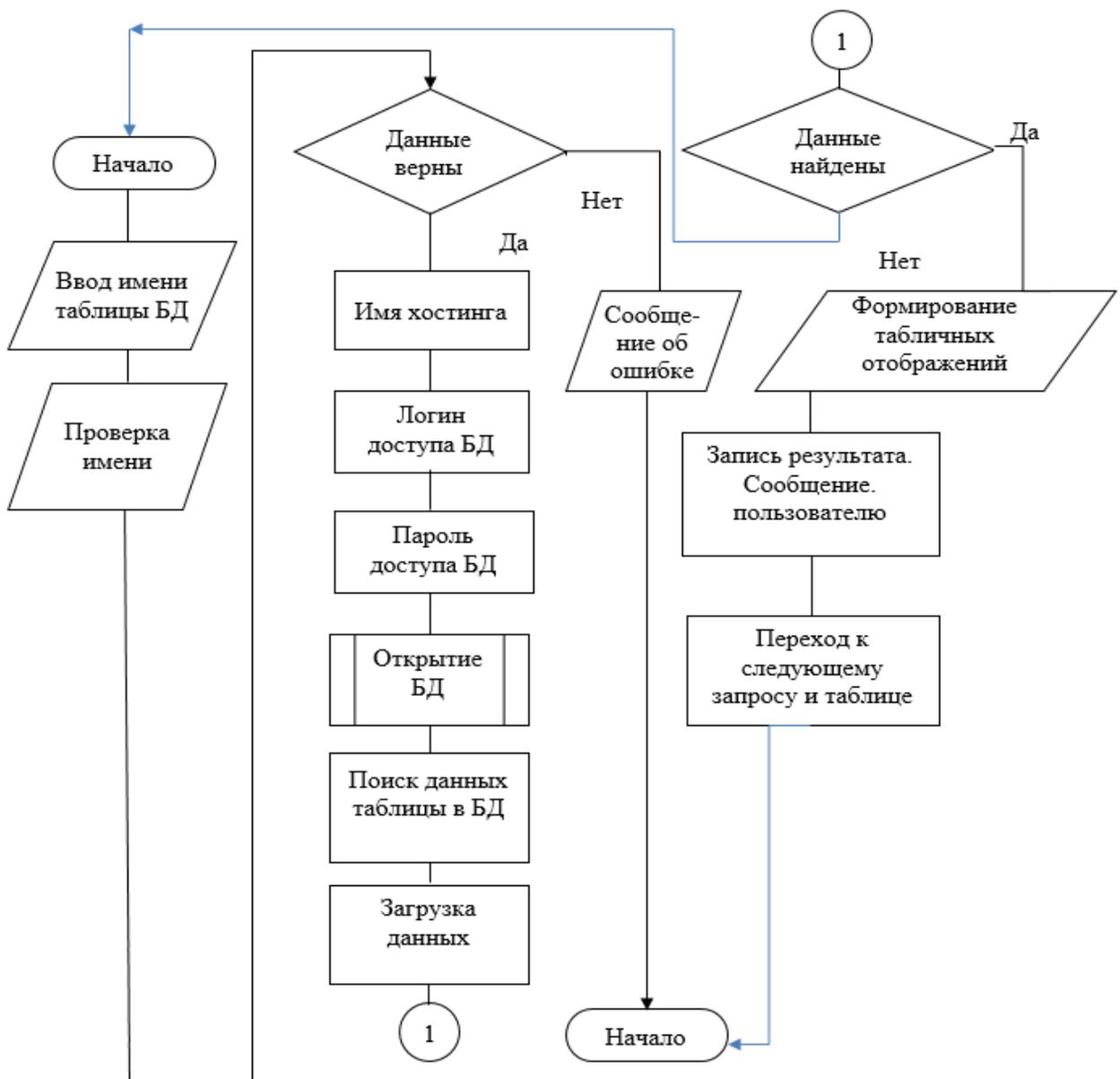


Рисунок 22 – Блок-схема алгоритма поиска, извлечения и отображения

В модуль View_Table.php – использует уровень доступа к данным DB_data, DB_Coding, DB_Class, чтобы получить данные из базы данных, а затем формирует табличные представления данных на экране.

3.4 Разработка базы данных

«Для создания базы данных используется система управления базы данных MySQL.

На текущий момент, она является одной из известнейших систем управления базами данных в Интернете. Это свободная СУБД, широко используемая такими крупными игроками на мировом IT-рынке, как Facebook, Twitter, YouTube и Airbnb. MySQL предлагает надежность, хорошую производительность и возможность масштабирования, что делает ее одним из лучших выборов для хранения и обработки данных в Интернете. Согласно информации сайта DB-Engines, MySQL находится на 2-ой позиции в рейтинге популярности СУБД.

MySQL разрабатывается и сопровождается Oracle, и доступна в двух вариантах: коммерческой (MySQL Enterprise Edition) и бесплатной (MySQL Community Edition)» [4].

«MySQL предоставляет мощные функции обработки данных. Этот программный продукт использует стандартный язык SQL для обработки данных, При этом, рассматриваемый продукт дает пользователю возможности для создания таблиц, выполнения запросов, вставки, обновления и удаления данных без знания языка запросов SQL.

MySQL сочетает в себе хорошую производительность, высокую надежность и простоту, что делает рассматриваемую СУБД популярным выбором у создателей и администраторов веб-сайтов, требующих обработки данных» [4].

«MySQL позволяет сохранять большие объемы данных в структурированном виде. Решает задачи надежного хранения данных и дает возможность быстро получать доступ к ним.

MySQL дает отличные возможности для управления данными, например, создание, изменение и удаление таблиц, индексов и отношений между ними. Дает контролировать и организовывать данные в удобном формате.

В эту СУБД заложена возможность горизонтального и вертикального масштабирования» [4].

«MySQL имеет в своем функционале различные функции безопасности, включая возможность установки привилегий пользователя, шифрование данных, аутентификацию и аудит, чтобы обезопасить информацию, хранящуюся в базах данных. При этом она обладает хорошей производительностью и быстродействием благодаря оптимизированным алгоритмам выполнения запросов - умеет работать с большими объемами данных с минимальными задержками.

СУБД может использоваться как для небольших веб-приложений, так и для серьезных корпоративных систем, поскольку предлагает различные методы масштабирования, включая горизонтальное и вертикальное масштабирование, что дает возможность расширять базы данных при возникновении необходимости. Это позволяет обрабатывать большое количество одновременных запросов и поддерживать беспрепятственный доступ к данным для пользователей» [4].

«MySQL используется во множестве различных областей и задач. Так она часто применяется для построения динамических сайтов и интернет-приложений. При этом СУБД используется в сочетании с веб-серверами (Apache, Nginx). Базы MySQL используются как хранилище данных для аналитических и отчетных систем. С помощью рассматриваемого программного продукта можно хранить большие объемы данных, а также выполнять сложные запросы и агрегировать данные, чтобы проводить анализ и создавать отчеты. MySQL используется также как СУБД для мобильных приложений, хранящих различные типы данных, такие как пользовательские профили, настройки, результаты и др. Во многом из-за своей надежности и хорошей производительности MySQL хорошо подходит для решения данных задач» [4].

«MySQL может быть встроен в другие приложения и устройства для взаимодействия с небольшими базами данных. Это могут быть системы управления контентом (CMS), блоги, форумы, физические устройства, такие как маршрутизаторы и т. д.

СУБД MySQL от компании Oracle представляет собой высокоэффективный и надежный программный продукт для хранения и обработки данных, нашедший свое применение в разнообразных областях и сферах человеческой деятельности.

MySQL имеет множество преимуществ, которые и делают эту СУБД одной из самых популярных в мире» [4].

«PHPMyAdmin, это веб-приложение с открытым кодом на PHP для администрирования данных под управлением СУБД MySQL, позволяющее создавать и управлять базой данных, таблицами базы данных и их содержимым в интерактивном режиме, используя браузер» [4].

На рисунке 23 представлена структура базы данных в представлении phpMyAdmin.

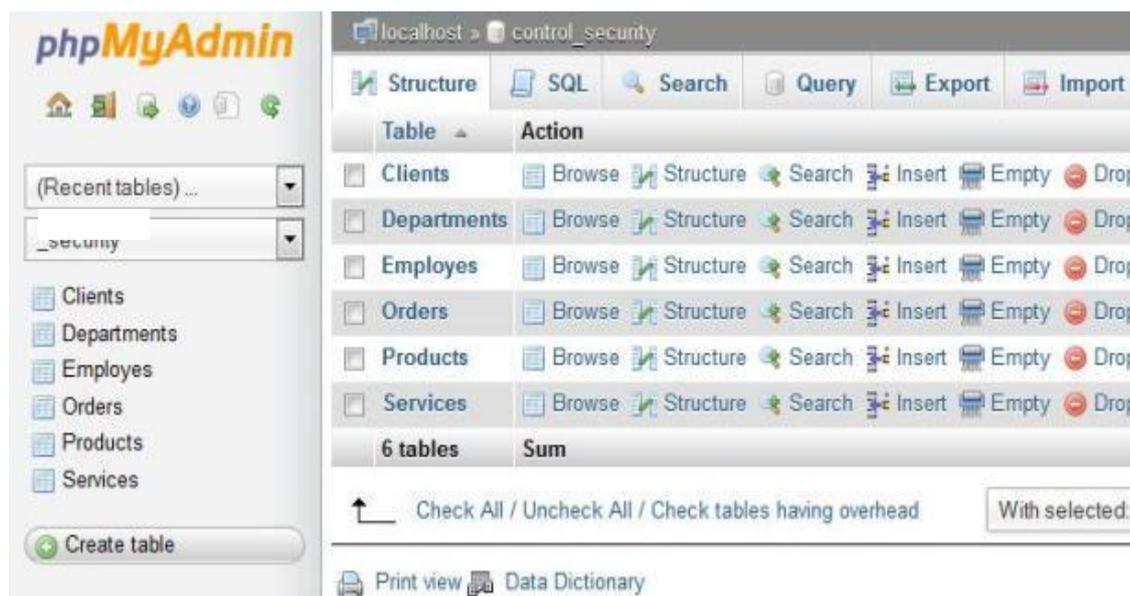


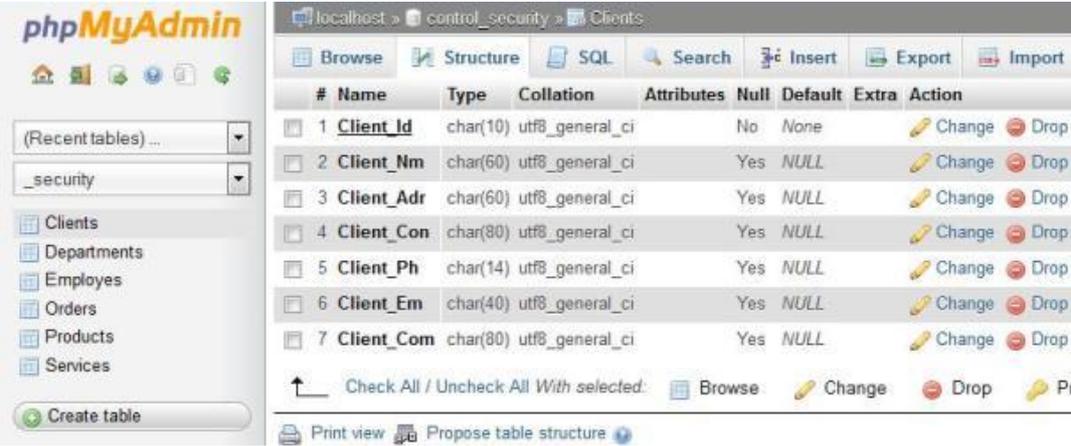
Рисунок 23 – Структура базы данных в представлении phpMyAdmin

Сначала с помощью этой программы создается база данных с именем. Затем последовательно создаются таблицы базы данных. Затем для каждой таблицы создается структура доменов с именами и характеристиками. После этого таблицы наполняются данными.

После того, как все эти операции с базой данных сделаны и прошла проверка правильного использования и отображения данных в системе, проводится операция экспорта базы данных. При этом данные экспортируются в виде SQL- файла, который сохраняется в папке SQL настоящего приложения.

«Когда система перемещается на хостинг общего пользования, все папки и файлы «заливаются» на удаленный сервер, следующим шагом является обратная операция импорта базы данных из SQL-файла в базу данных MySQL, расположенную уже на хостинге» [17].

Структура таблицы Clients изображена на рисунке 24.



#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	<u>Client_Id</u>	char(10)	utf8_general_ci		No	None		Change Drop
2	Client_Nm	char(60)	utf8_general_ci		Yes	NULL		Change Drop
3	Client_Adr	char(60)	utf8_general_ci		Yes	NULL		Change Drop
4	Client_Con	char(80)	utf8_general_ci		Yes	NULL		Change Drop
5	Client_Ph	char(14)	utf8_general_ci		Yes	NULL		Change Drop
6	Client_Em	char(40)	utf8_general_ci		Yes	NULL		Change Drop
7	Client_Com	char(80)	utf8_general_ci		Yes	NULL		Change Drop

Рисунок 24 – Структура таблицы Clients

Однако, администратор системы может работать не только с рассмотренными ранее пакетами для администрирования базы данных, но и со специальными средствами, предоставляемыми системой – веб-интерфейсом администратора.

3.5 Управление доступом пользователей

Для посетителей сайта открыта информация по услугам, оборудованию, сотрудникам салона.

Услуги салона представляются в виде таблицы с характеристиками услуг и ценами на услуги.

В системе предусмотрена регистрация пользователей, которые при этом становятся клиентами салона и могут заказать услугу через личный кабинет клиента. Это делается для того, чтобы, став клиентом салона, клиент пользователь не вводил в систему свои данные, а система уже знала, кто это и вносила его данные в заказ автоматически.

«Используя веб-ориентированную систему, клиент сокращает количество очных посещений салона или звонков по телефону с целью реализации его пожеланий и заказа на услуги» [11].

В личном кабинете пользователя клиент видит свой заказ, перечень своих заказов.

Администратор системы имеет логин и пароль для входа в свой личный кабинет, в котором регламентируется работа администратора с базой данных системы. Для работы с базой данных системы реализован веб-интерфейс администратора, которые предусматривает все операции с определенным перечнем данных.

3.6 Описание работы приложения

Общий алгоритм работы пользователей – клиентов и посетителей в системе представлен на рисунке 25.

Сначала, это всегда регистрация, а затем авторизация, если пользователь еще не регистрировался или авторизация пользователя для зарегистрированных пользователей.

Построенные блок-схемы алгоритмов работы программного обеспечения определяют структуру веб-приложения, взаимодействие программных модулей системы.

Рассмотри блок-схему алгоритма регистрации пользователей.

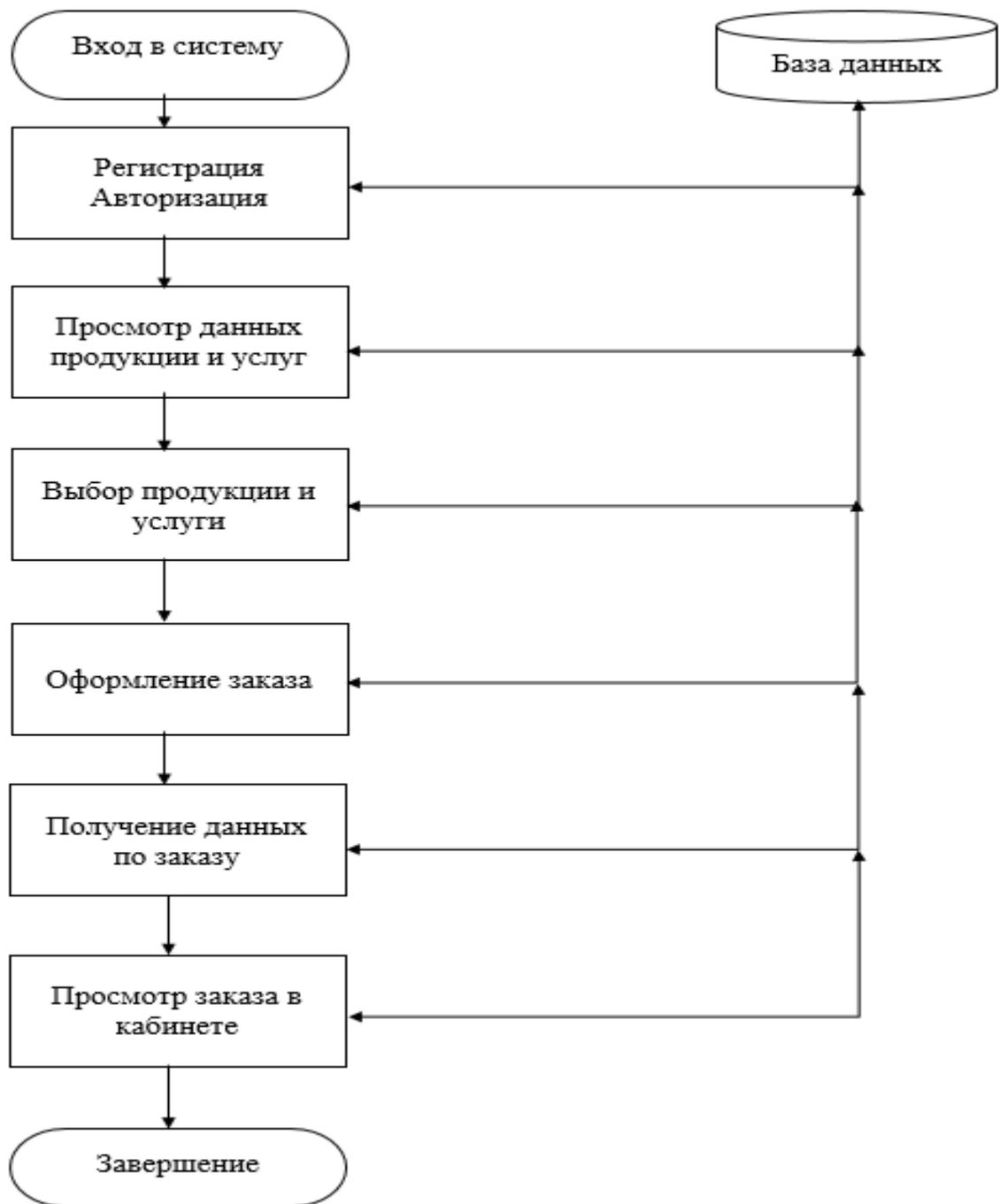


Рисунок 25 – Алгоритм работы пользователя-клиента

Пользователь нажимает на вкладку регистрация, появляется форма регистрации с полями для занесения данных.

Подсистема регистрации и авторизации (комплекс модулей) соединяется с базой данных, проверяет наличие такого логина.

Если он уже есть, то выдается сообщение об ошибке, если такого нет, то после завершения заполнения всех полей система выдает сообщение об успешной регистрации в качестве клиента компании.

Блок-схема алгоритма регистрации пользователей представлена на рисунке 26.

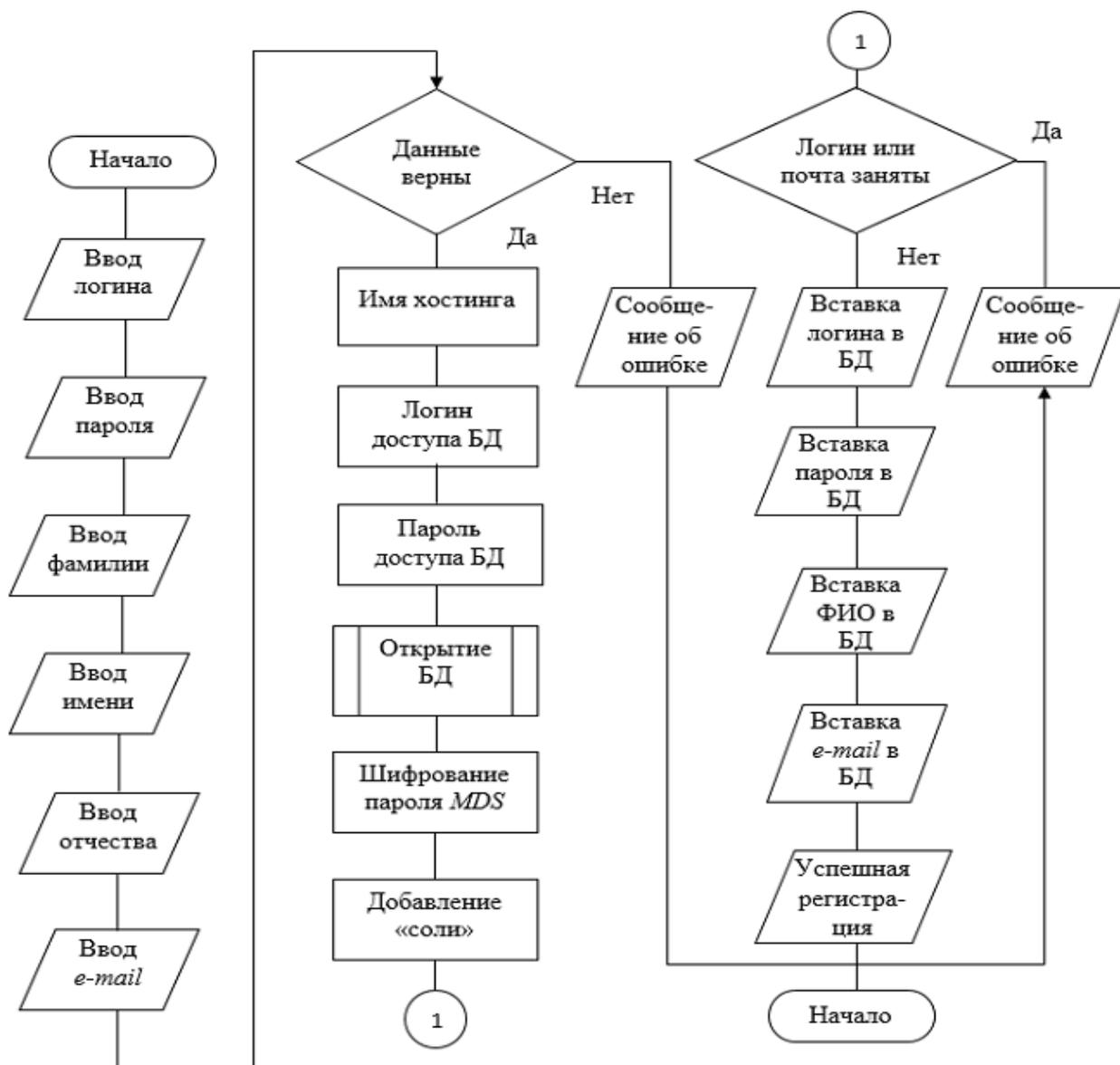


Рисунок 26 – Блок-схема алгоритма регистрации пользователей-клиентов

Данные нового клиента сохраняются в базе данных системы. Пароли всех пользователей хранятся в зашифрованном виде.

Когда пользователь нажимает «ВХОД», система предлагает заполнить поля с логином и паролем в форме авторизации. Пользователь заносит эти данные.

Система проводит поиск логина и пароля в базе данных зарегистрированных пользователей. Пароль пользователя шифруется в соответствии с тем видом, который представлен в базе данных системы.

Для этого осуществляется предварительная проверка наличия логина в системе.

Если внесенные логин и пароль найдены, система «знает» такого зарегистрированного пользователя, то он может войти в личный кабинет клиента для формирования заказов или просмотра ранее сформированных заказов.

Система сохраняет все данные, полученные после авторизации пользователей, и использует их во всех модулях, где они необходимы, например, эти данные используются при выборе клиентом услуг и оформлении заказа.

Тогда в форме заказа автоматически заполняются поля, связанные с данными пользователя-клиента.

Блок схема алгоритма работ программ при авторизации пользователя представлена на рисунке 27.

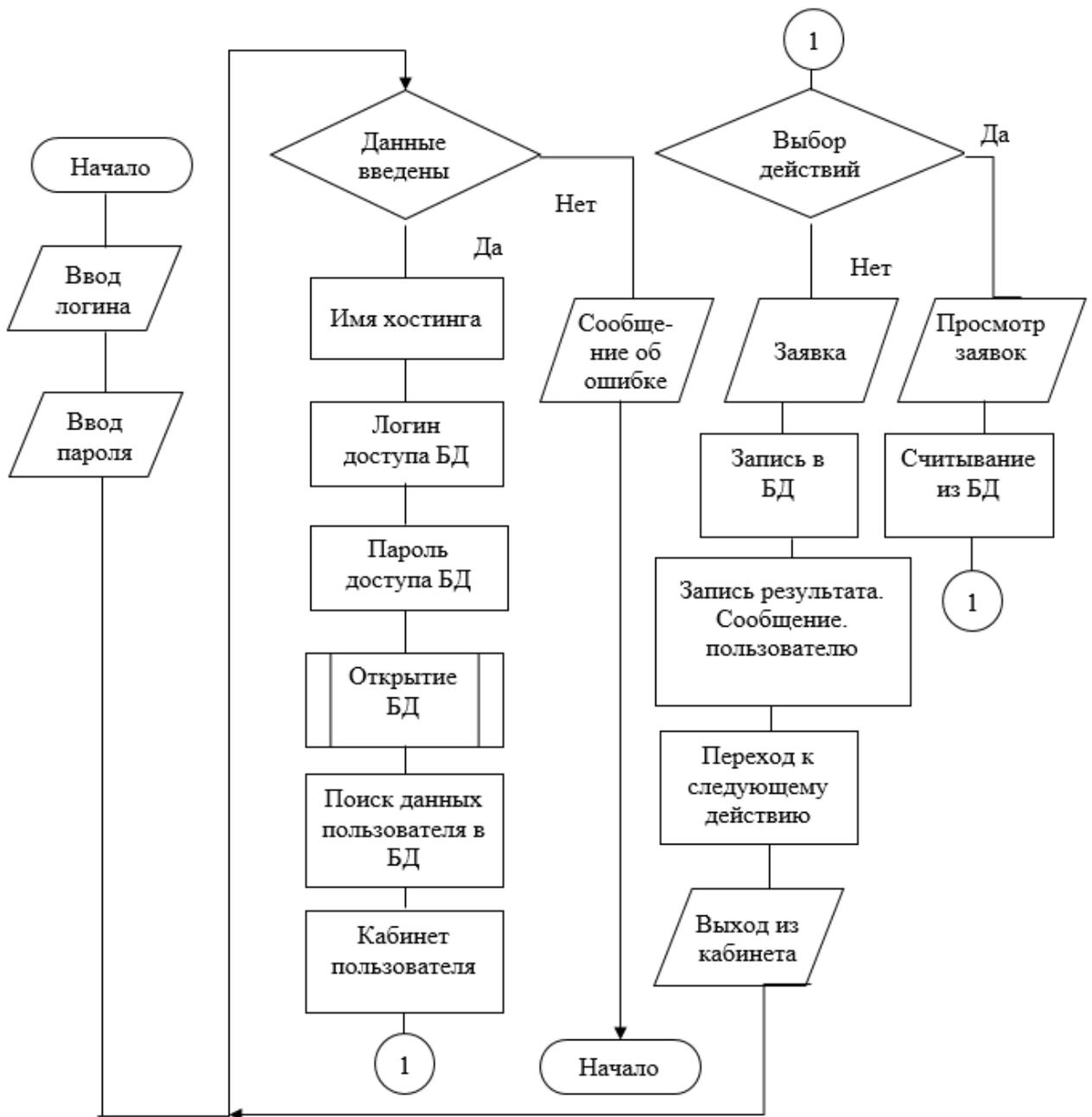


Рисунок 27 – Блок-схема алгоритма авторизации пользователей

Форма регистрации клиентов представлена на рисунке 28.

РЕГИСТРАЦИЯ ПОЛЬЗОВАТЕЛЕЙ

Регистрация пользователей подразумевает создание логина и пароля с целью входа в защищенные области сайта, в частности, в личный кабинет пользователя. Зарегистрированный пользователь может делать заявки, просматривать свои заявки, просматривать свои сообщения.

Наименования полей	Заносимые или обновляемые данные
Логин (карточка) *	<input type="text" value="IvanaIvana"/>
Пароль*	<input type="password" value="*****"/>
Ф И О*	<input type="text" value="Иванова Ирина Ивановна"/>
Адрес*	<input type="text" value="г.Тольяти, ул. Центральная 15"/>
Место работы	<input type="text" value="Университет"/>
Телефон*	<input type="text" value="9999-999-9999"/>
Электронный Адрес*	<input type="text" value="ivan@mail.ru"/>
Примечания	<input type="text"/>
<input type="button" value="ДОБАВИТЬ"/>	

Рисунок 28 – Форма регистрации клиентов

После регистрации клиент нажимает «ВХОД», тогда появляется форма авторизации – всплывающая форма (рисунок 29).

localhost
Этот сайт просит вас войти.

Имя пользователя

Пароль

Если вы зарегистрированный пользователь системы, то
ДОБРО ПОЖАЛОВАТЬ!

Для входа в систему зарегистрированных пользователей и начала работы клиньте по первой ссылке в таблице, расположенной ниже.
 Если вы не зарегистрированный пользователь системы, то вам необходимо сначала зарегистрироваться - вторая ссылка в таблице.

Регистрация подразумевает создание логина и пароля для дальнейшего входа в систему и занесения некоторых персональных данных. Регистрация займет определенное время. После заполнения форм регистрации, ваши данные будут автоматически отправлены на адрес электронной почты администратора системы, который идентифицирует ваши данные и подтвердит ваше право на регистрацию в системе. После подтверждения регистрации, на ваш адрес электронной почты будет отправлено подтверждающее сообщение с логином и паролем.

[Вход для Зарегистрированных Пользователей](#)
[Регистрация Новых Пользователей](#)
[Вход для администратора системы](#)

Рисунок 29 – Приглашение для входа или регистрации

После авторизации клиент входит в личный кабинет (рисунок 30).



Рисунок 30 – Окно личного кабинета клиента

Появляется список услуг со ссылками на заказ (рисунок 31).

Код услуги	Изображение	Наименование услуги	Описание/Характеристика	Длительность	Стоимость	Комментарий	ЗАКАЗАТЬ
1		Парикмахерские услуги	Диагностика, лечение, лечение кожи головы, укладка, покраска	1.50	150	ЛУЧШИЙ МАСТЕР	ОФОРМИТЬ ЗАЯВКУ
2		КОСМЕТОЛОГИЯ	ЧИСТКА ЛИЦА ПИЛИНГ ЛИЦА ГРНКАРБОКСИТЕРАПИЯ MEDICARE АППАРАТ REMODELING FACE	2.10	120	ЛУЧШИЙ МАСТЕР	ОФОРМИТЬ ЗАЯВКУ

Рисунок 31 – Список услуг со ссылками

Нажав на ссылку на услугу, пользователь получает форму для заказа в виде, представленном на рисунке 32.

Система сама заполняет в форме заказа дату, код пользователя системы, фамилию, имя, отчество клиента, код услуги и наименование услуги. Заказ сразу поступит в базу данных и становится доступным менеджеру салона для назначения конкретной даты посещения салона клиентом.



Наименования полей	Заносимые или обновляемые данные
Код заказа	1713966997
Дата заказа	2024-04-24
Код клиента	IvanaIvana
ФИО клиента	Иванова Ирина Ивановна
Код услуги	1
Наименование	Парикмахерские услуги
Стоимость	150
<input type="button" value="Добавить"/>	

Рисунок 32 – Форма заказа

Для обеспечения безопасности и надежности базы данных администратором системы своевременно проводится работы по созданию дампов базы данных с помощью программы phpMyAdmin.

Вход в систему администратора осуществляется по имени и паролю.

Только после авторизации администратор может приступить к работе со всей базой данных системы.

Заключение

В рамках выпускной квалификационной работы было проведено исследование и проведена разработка веб-ориентированной информационной системы, веб-приложения учета и мониторинга онлайн заказов на услуги салона красоты компании "САЛОН КРАСОТЫ".

Было проведено исследование бизнес-процесса онлайн заявок, сделан вывод о необходимости внедрения интернет технологий для онлайн заказов на услуги салона.

Был проведен выбор актуальных средств разработки веб-приложений, которые бы обеспечивали быструю разработку и внедрение приложения в деятельность компании.

Для достижения поставленной цели была проведена разработка базы данных веб-приложения в среде СУБД MySQL с использованием таких средств разработки, как стенд разработки веб-приложений Денвер, включающий язык разработки динамических веб-страниц PHP, а также программа создания и управления базой данных phpMyAdmin.

Были разработаны компоненты программного обеспечения, проведено тестирование этих компонент и работоспособности всей системы в целом. Система показала свою работоспособность и соответствие выполняемых функций заявленным в рамках проекта требованиям.

Задачи поставленные и решенные при выполнении ВКР:

- изучены источники по теме ВКР и объекту исследования;
- проведен анализ бизнес-процессов, связанных с онлайн-заказами услуг салонов;
- проведен анализ актуальных средств, используемых в разработках веб-приложений;
- обоснован выбор программных средств разработки для реализации функций веб-приложения;

- разработана структура базы данных веб-приложения, создана база данных веб-приложения в среде СУБД MySQL, используя прикладные инструменты разработки баз данных;
- разработан веб-интерфейс пользователей веб-ориентированной системы;
- проведено тестирование разработанного программного обеспечения с использованием локальных средств поддержки веб-приложений;
- проведена публикация веб-приложения на хостинге для демонстрации свойств и возможностей в соответствии с проектом.

Программное обеспечение создано на основе современных концепций программирования, что обеспечивает его гибкость, легкую настройку на изменение или дополнения структуры базы данных.

Практическая значимость проекта состоит в том, что систему можно использовать для деятельности салонов, что обеспечить принципиально новые преимущества для компании в обеспечении заказами посредством сети интернет.

Список используемой литературы и используемых источников

1. 11 типов современных баз данных: краткие описания, схемы и примеры БД [Электронный ресурс]. URL: <https://proglib.io/p/11-tipov-sovremennyh-baz-dannyh-kratkie-opisaniya-shemy-i-primery-bd-2020-01-07> (дата обращения: 12.05.2024).
2. База данных (БД): виды, типы, примеры использования и систем управления базами данных [Электронный ресурс]. URL: <https://kokos.com/blog/baza-dannyh/> (дата обращения: 12.05.2024).
3. Виды баз данных [Электронный ресурс]. URL: <https://selectel.ru/blog/databases-types/> (дата обращения: 12.05.2024).
4. Виды баз данных - обзор и сравнение популярных типов [Электронный ресурс]. URL: <https://www.corpsoft24.ru/about/blog/vidy-baz-dannykh/> (дата обращения: 12.05.2024).
5. Кормен Т. Алгоритмы: построение и анализ/ Т. Кормен, Ч. Лейзерсон, Р. Ривест. Диалектика (Вильямс), 2020. 450 с.
6. Моделирование данных: зачем нужно и как реализовать. [Электронный ресурс]. URL: <https://habr.com/ru/articles/554388/> (дата обращения: 12.05.2024).
7. Моделирование данных в СУБД: от концептуального проектирования к реализации. [Электронный ресурс]. URL: <https://appmaster.io/ru/blog/modelirovanie-dannykh-v-rdbms> (дата обращения: 12.05.2024).
8. Основы баз данных. [Электронный ресурс]. URL: <http://repo.ssau.ru/bitstream/Uchebnye-izdaniya/Osnovy-baz-dannyh-96560/1/> Крикунов М.М. Основы баз данных 2021.pdf (дата обращения: 12.05.2024).
9. Отличия реляционных и нереляционных баз данных [Электронный ресурс]. URL: <https://www.xelent.ru/blog/otlichiya-relyatsionnykh-i-nerelyatsionnykh-baz-dannykh/> (дата обращения: 12.05.2024).

10. Реляционные и нереляционные базы данных: какие выбрать? [Электронный ресурс]. URL: <https://habr.com/ru/companies/sberbank/articles/672022/> (дата обращения: 12.05.2024).
11. Смирнова Г.Н. Проектирование экономических информационных систем/ Г.Н. Смирнова, Ю.Ф. Тельнов. М.: МЭСИ, 2021. 223 с.
12. Томсон Л. Разработка Веб-приложений на PHP и MySQL/ Л.Томсон, Л. Веллинг. М.: Издательство DiaSoft, 2020. 560с.
13. Трофимов С.А. UML диаграммы в Rational Rose [Электронный ресурс]. URL: <https://knigogid.ru/books/1776717-uml-diagrammy-v-rational-rose/toread> (дата обращения: 12.05.2024).
14. Фаулер М. UML. Основы [Электронный ресурс]. URL: https://k0d.cc/storage/books/UML/uml_osnovy_3-e_izd.pdf (дата обращения: 12.05.2024).
15. Что такое База Данных (БД). [Электронный ресурс]. URL: <https://habr.com/ru/articles/555760/> (дата обращения: 12.05.2024).
16. Adam Trachtenberg, David Sklar, PHP Cookbook, O'Reilly Media, 2014, 813 pages .
17. Bryan Bibat, Kevin Wayne. HTML5 Shoot 'em Up in an Afternoon, 2018, 95 pages.
18. David Cross. Data Munging with PHP, Manning Publications Co., 2021, 304 pages.
19. John B. Schneider, Shira Lynn Broschat, Jess Dahmen. Algorithmic Problem Solving with PHP, Washington State University, Published, 2018, 360 pages .
20. Software Requirements [Электронный ресурс]. URL: <http://beervolume.com/oop/2020/software-requirements/>(дата обращения: 12.05.2024).