МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Отделение информационных технологий и вычислительной техники

**Курсовая работа по дисциплине «Технология разработки ПО»**

**тема: Разработка информационной системы учета гостиничных номеров.**

Выполнил: Болотненко Александр Дмитриевич

Проверил: Сафаров Рустам Ахтамович

Новосибирск, 2021 г.

Содержание

[Введение 2](#_Toc90396266)

[ГЛАВА 1. ИССЛЕДОВАТЕЛЬСКАЯ ЧАСТЬ 4](#_Toc90396267)

[1.1 История развития сети Интернет 4](#_Toc90396268)

[1.2 Анализ предметной области и обзор существующих решений 8](#_Toc90396269)

[1.3 Требования организации на разработку программного обеспечения 11](#_Toc90396270)

[1.4 Проектирование базы данных 14](#_Toc90396271)

[ГЛАВА 2. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ 15](#_Toc90396272)

[2.1 Выбор языка программирования 15](#_Toc90396273)

[2.2 Описание и разработка приложения 25](#_Toc90396274)

[ЗАКЛЮЧЕНИЕ 28](#_Toc90396275)

[СПИСОК ЛИТЕРАТУРЫ 30](#_Toc90396276)

[Приложение 31](#_Toc90396277)

# Введение

Интернет стал, наиболее эффективным средством рекламы и продвижения и является одним из важных элементов современной цивилизации. Интернет может удовлетворить все потребности современного человека: это покупки, заключение деловых отношений, поиск клиентов и так далее. Но многие заходят в Интернет читать, развлекаться, общаться, узнавать что-то новое для себя. Поэтому создание сайтов стало таким популярным.

Актуальность темы состоит в том, что гостинице для получения наибольшей выгоды требуются дополнительные способы распространения своих услуг. Сайт - это эффективная площадка для привлечения самой разной целевой аудитории, с помощью которой гостиница занимается продвижением своих услуг. С его помощью можно предоставлять потенциальным клиентам информацию о ценах, доступных услугах и обеспечить возможность заблаговременно заказать их через модуль онлайн-бронирования.

Цели курсового проекта:

– систематизация и закрепление полученных теоретических знаний и практических умений по общепрофессиональным и специальным дисциплинам, а также практики по эксплуатации программных средств информационных систем;

– разработать сайт для гостиничного комплекса (далее гостиницы);

– Для достижения поставленных целей необходимо выполнить следующие задачи:

– изучить и проанализировать имеющиеся материалы по вопросу деятельности организации;

– рассмотреть предназначение сайта гостиницы;

– проанализировать существующие языки для написания сайта;

– создать сайт гостиницы;

– рассчитать экономическую эффективность разработки.

Объект исследования: интернет-ресурс гостиницы.

Предмет - язык гипертекстовой разметки HTML как средство создания сайта.

При написании проекта использовались метод теоретического исследования (анализ).

Практическая значимость проекта заключается в том, что с помощью данного сайта можно просматривать ресурсы и информацию на сайте.

# ГЛАВА 1. ИССЛЕДОВАТЕЛЬСКАЯ ЧАСТЬ

## 1.1 История развития сети Интернет

Интернет - это глобальная система связанных между собой протоколом IP компьютерных сетей. С помощью Интернета в мире созданы Глобальное информационное пространство и Всемирная паутина WWW.

Сформирован Интернет из огромного количества сетей разнообразной топологии и архитектуры (правительственных, корпоративных, домашних и пр.). Перенаправлением пакетов данных и их сортировкой занимаются особые устройства - маршрутизаторы, а передача данных может вестись с помощью большинства современных каналов связи.

К созданию Интернета человечество шло долгие годы, изобретая новые и новые средства связи: изобретение телеграфа (1836г.); первый атлантический кабель для связи между континентами (1858г.); изобретение телефона (1876 г.).

Интернет состоит из взаимосвязанных сетей, эксплуатируемых университетами, государственными учреждениями, военными, корпорациями и другими организациями. Эти сети связаны друг с другом различными видами оборудования такими, как маршрутизаторы, мосты и коммутаторы. Маршрутизаторы решают, в каком направлении отослать сетевые данные, предварительно посылая пакеты по нужному адресу и затем "маршрутизируя" данные в направлении соответствующего компьютера, где эти пакеты собираются вновь. Мосты предназначены для соединения двух каких-либо сегментов кабельной проводки внутри сети; коммутаторы - это приспособления для замыкания и размыкания цепей.

Первоначальная сеть была создана в конце 1960-х годов и называлась ARPANET. Ее цель состояла в том, чтобы дать возможность подрядчикам, университетам и сотрудникам Министерства обороны, участвующим в исследованиях и разработках оборонного характера, поддерживать связь по компьютерным сетям и совместно использовать вычислительные ресурсы тех немногих на то время мощных компьютеров, которые находились в разных географических точках. В сентябре 1969 года в Калифорнийском университете Лос-Анджелеса (UCLA) была создана пакетная сеть с одинарным маршрутизатором. А вскоре уже заработали четыре маршрутизатора. Система ARPANET быстро разрасталась. К 1977 году она включала в себя 111 систем хостинга. Поскольку многие университеты и исследовательские центры, входящие м ARPANET позднее подсоединили свои местные сети к APANET, в конечном счете она стала стержневой сетью ARPA Internet, система многих сетей, использующих в качестве языка Протокол управления передачей/Инернет-протокол (TCP/IP) как основу архитектуры. ARPANET сыграл важную роль в развитии Интернета. В свое время это была самая большая, наиболее быстро действующая и населенная часть Интернета,

В 1984 году ARPANET разделилась на две сети: ARPANET и Сеть оборонных данных (DDN). DDN продолжает оставаться одной из составляющих сетей Интернета. (Открытая часть DDN называется MILNET). В 1990 году ARPANET прекратила работу, поскольку более эффективную работу Интернет на коммерческой основе смогла обеспечить сеть Национального научного фонда (NSF) и различные сети среднего уровня, финансируемые Национальным фондом науки. Продолжает работать военная сеть в Министерстве обороны.

В 1985 году Национальный фонд науки финансировал создание нескольких национальных суперкомпьютерных центров, которые были предназначены для использования исследователями в университетах по всей стране. К тому времени многие университеты штатов и регионов создали свои местные и региональные сети, а некоторые использовали систему TCP/IP. NSF финансировал создание сетей мощностью в 56 килобайтов/сек, объединившую пять суперкомпьютерных центров и предложил любым региональным и университетским компьютерным центрам, находившимся в пределах физической досягаемости от этой сети подсоединиться к ней. Это был зароды сети Интернет в нынешнем ее виде. Первоначально смысл подключения к ней состоял в том, что это давало дистанционный доступ к суперкомпьютеру.

Целый ряд университетов подсоединились к сети NSF, чтобы получить доступ к суперкомпьютерам. Кроме исследовательских задач выяснилось, что сеть может с успехом быть использована для целей электронной почти, компьютерной передачи файлов и новостей. Объем трафика в сети быстро возрастал. В ноябре 1987 года NSF заключил контракт с Merit Network Inc., которая совместно с IBM, MCI и штатом Мичиган должны были заняться усовершенствованием и эксплуатацией опорной сети NSF, первого уровня доступа к Интернету.

К этому времени основная цель опорной сети NSF состояла в том, чтобы обеспечить связь между растущими региональными сетями, созданы различными университетскими системами. Термин "Интернет" начал употребляться с 1983 года для обозначения концепции взаимокоммутируемых сетей.

В мае 1993 года NSF радикальным образом изменил архитектуру Интернета, поскольку правительство не хотело больше иметь дела с системами опорной сети. Вместо нее NSF выделило ряд «точек доступа в сеть» (NAP), в которых могли бы взаимодействовать друг с другом частные коммерческие опорные сети. В 1994 году NSF объявил о строительстве четырех NAP в Сан-Франциско, Нью-Йорке, Чикаго и Вашингтоне, округ Колумбия. Заказ NSF на четыре точки доступа в сеть был выполнен Ameritech, PacBell, Sprint и MFS Datanet. Дополнительная точка доступа, известная под названием MAE-West, была создана MFS Dananet на Западном побережье.

апреля 1995 года опорная сеть NSF была практически закрыта, а архитектура NAP превратилась в Интернет.

За два десятилетия своего существования Сеть Интернет претерпела кардинальные изменения. Она зарождалась в эпоху разделения времени, но сумела выжить во времена господства персональных компьютеров, одноранговых сетей, систем клиент-сервер и сетевых компьютеров. Она проектировалась до первых локальных вычислительных сетей (ЛВС), но впитала эту новую сетевую технологию, равно как и появившиеся позднее технологии коммутации ячеек и кадров. Она задумывалась для поддержки широкого спектра функций, от разделения файлов и удаленного входа до разделения ресурсов и совместной работы, породив электронную почту и, в более поздний период, - Всемирную паутину. Но важнее всего то, что Сеть, создававшаяся вначале как объект деятельности небольшого коллектива исследователей, выросла до коммерчески выгодного предприятия, в которое ежегодно вкладываются миллиарды долларов.

Для того, чтобы различные компьютеры в Интернете могли сообщаться друг с другом, используются протоколы, т.е. правила или условия коммуникации. Язык Интернета называется TCP/IP, что означает протокол Управления передачей/Интернет-протокол. Любой компьютер для того, чтобы общаться в Интернете, должен уметь "говорить" на языке TCP/IP. Этот стандарт является "открытым", что означает, что он не является фирменным продуктом какой-либо одной компании. Основная операционная система называется UNIX, но стандарт Интернета приспособлен для использования на всех наиболее распространенных системах, таких как UNIX, PC и Macintosh.

## 1.2 Анализ предметной области и обзор существующих решений

Гостиничный комплекс, который рассматривается в данной работе, будет содержать одну гостиницу. Гостиница дает возможность покупателям забронировать номера. Для того, чтобы продать номер клиенту, необходимы следующие данные: ФИО, e-mail, телефон, дату заселения, тип номера и количество взрослых. Гостиница хранит информацию о номерах в гостинице, которые заняты клиентами. В случае, если бронирование номеров гостиницы переносится на более позднее время или дату, изменения должны быть внесены в бюллетень покупателей, зарегистрированных в этот номер гостиницы, сообщены покупателю и должна быть предоплата за бронирование номера. База данных должна быть защищена от несанкционированных действий злоумышленников паролем на вход.

База данных должна выполнять следующие основные задачи:

– хранить сведения обо всех номерах гостиницы;

– хранить сведения о забронированных номерах клиентов;

– обеспечивать оформление номера на нужного покупателя.

База данных должна быть настроенной, то есть в ней должна присутствовать возможность изменения, дополнения, в том числе и удаления следующих параметров:

– клиента и (или) номера (при въезде или отъезде клиента).

Для осуществления процесса оформления номера необходимо ввести фамилию, имя, отчество по которому зарегистрирован клиент. После покупки номера клиентом необходима сохранность информации о клиенте. Должна быть возможность просматривать список зарегистрированных клиентов и покупателей, их данные, и т.д. (только для администрации).

Обзор существующих решений.

Сеть отелей «Седьмое небо» 7or.ru



Рисунок 1. Сайт отеля «Седьмое небо»

Достоинства:

– информативная главная страница, вся контактная информация аккуратно расположена в «шапке» сайта;

– гармоничное расположение элементов;

– удобная навигация;

– стилистически вписывающийся в дизайн слайдер.

– наличие формы для отзывов.

Недостатки:

– отпугивающий рекламный баннер при попадании на сайт;

– неаккуратное, выпадающее в виде списка, навигационное меню;

– отсутствие формы бронирования и предварительного заказа;

– неуместно всплывающие рекламные блоки собственных сервисов;

– отсутствие адаптивности.

Отель "Привал" otel-prival.ru.

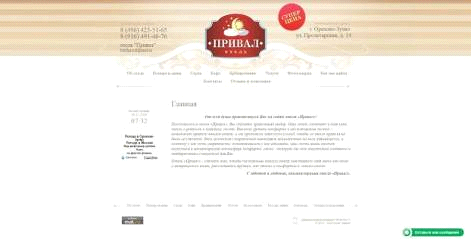


Рисунок 2. Сайт отеля «Привал»

Достоинства:

– лаконичная главная страница оставляет хорошее первое впечатление;

– информативность и удобство;

– все услуги выставлены в «шапке» сайта;

– наличие модуля для бронирования и отзывов;

Недостатки:

– отсутствие адаптивности;

– бесполезные виджеты с датой, временем и погодой, дублирование шапки сайта в подвале.

## 1.3 Требования организации на разработку программного обеспечения

«HOTEL✰✰✰✰» - это общество с ограниченной ответственностью; участники общества не отвечают по его обязательствам и несут риск убытков, связанных с деятельностью общества, в пределах стоимости принадлежащих им долей в уставном капитале общества.

Сайт гостиницы выполняет следующие функции: предоставление общей информации о гостинице, бронирование номеров, обратная связь.

Сайт ведет непосредственную работу с клиентами по вопросам предоставления номеров для временного проживания. Клиенты интернет-магазин - физические лица, для которых предусмотрены номера различных категорий.

Клиент заходит на сайт, где расположены номера гостиницы с выставленными на них ценами и характеристиками. Клиент рассматривает товар в области где показаны товары, после чего выбирает какой-либо определенный товар и нажимает на кнопку бронировать тем самым товар переходит в форму бронирования, далее идет ввод персональных данных покупателя чтобы компания «HOTEL✰✰✰✰» могла забронировать номер под конкретного гостя. Каждая бронь фиксируется в базе данных. Документом, подтверждающим покупку, является чек, который выписывается продавцом, в котором содержатся дата заказа, тип номера, количество номеров, количество взрослых/детей в номере.

Основанием для разработки является выпускная квалификационная работа.

Наименование работы: разработка сайта гостиницы.

Сайт гостиницы «HOTEL✰✰✰✰» предназначен для предоставления с целью облегчения получения пользователя актуальной информации о компании, упрощении обратной связи, коммуникаций с партнёрами и для привлечения новых клиентов.

Требования к функциональным характеристикам.

Сайт гостиницы «HOTEL✰✰✰✰» должен обеспечивать выполнение функций:

– предоставление информации о гостинице;

– предоставление информации об услугах, доступных номерах;

– предоставление контактной информации;

– возможность бронирования услуг;

– возможность обратной связи.

Пользователь при попадании на главную страницу сайта имеет возможность для просмотра описания, сведений, изображений и видеофрагментов о гостинице. Также он может забронировать номер или оставить отзыв в соответствующих формах. На сайте присутствуют отдельные страницы для фотогалереи, расположения дополнительной информации о местоположении и услугах.

Требования к организации входных данных.

Входные данные программы должны быть организованы в виде вводимого в специальную форму текста, соответствующего определенному шаблону. Данные, вводимые вручную, проверяются на корректность после попытки сохранения.

Требования к организации выходных данных.

Выходные данные программы должны быть организованы в виде таблиц базы данных. Доступ к таблицам зависит и от принадлежности пользователя к определенной группе пользователя с теми или иными правами.

Файлы указанного формата должны храниться в базе данных сайта, отформатированных согласно требованиям операционной системы. В программе необходимо предусмотреть резервное копирование данных; парольный доступ в систему.

Требования к надежности.

Разрабатываемый сайт должен иметь:

– возможность самовосстановления после сбоев;

– парольную защиту при запуске;

– возможность резервного копирования базы данных;

– исключение несанкционированного копирования персональных данных покупателей;

– предусмотреть контроль за вводимыми данными при работе пользователя с программой.

Требования к составу и параметрам технических средств.

Серверная часть:

– Веб-сервер Apache 1.3.18 и выше

– PHP 4.2.0 и выше (должен быть собран как модуль Apache)

– СУБД MySQL 4.1.14 и выше (предпочтительно: поддержка формата InnoDB).

– Компьютер с процессором Pentium IV 2 ГГц (рекомендуется от 3 ГГц)

– Оперативная память 1 Гб (рекомендуется от 2 Гб).

– Место на жестком диске от 1 Гб.

Клиентская часть:

– Любой из перечисленный ниже браузеров (указана минимальная версия) с включенным интерпретатором JavaScript:

– Internet Explorer 10 / Mozilla Firefox 50 / Opera 10.10.

– Компьютер с процессором Pentium IV 1 ГГц (рекомендуется от 1.5 ГГц).

– Оперативная память 256 Мб (рекомендуется от 512 Мб).

– Требования к информационной и программной совместимости.

– Программа должна работать на операционной системе Windows 7/8/8.1/10/11 и современных интернет-браузерах.

## 1.4 Проектирование базы данных

Проектирование базы данных.

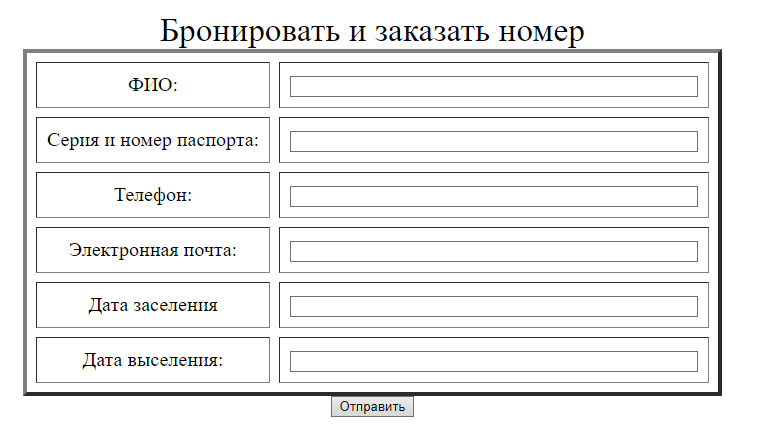


Рисунок 3 . Отправить информацию на базу данных.



Рисунок 4 . Структура таблицы homer.



Рисунок 5 . Результат таблицы получении информации homer.

# ГЛАВА 2. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ

## 2.1 Выбор языка программирования

Язык гипертекстовой разметки HTML.- это гипертекстовый язык разметки (HyperText Markup Language), который используется для создания документов в Интернет (веб-страниц). При помощи HTML создается веб-страница, содержащая все нужные элементы. На HTML странице можно разместить простой текст, выделить его жирным или курсивом, вставить ссылку, таблицу, нумерованный или ненумерованный список, картинки, разбить текст на абзацы и разделы, задавать разделам заголовки. Также на HTML страницу можно вставить форму с текстовыми полями, кнопками, выбором вариантов из списка, checkbox'ами и radio элементами. В HTML5 в страницу можно встраивать видео и аудио файлы, рисовать при помощи canvas, делать простые анимации при помощи новых тегов (marquee).

В 1986 году теоретики Международной организации по стандартизации приняли стандарт ISO-8879, посвященный описанию метаязыка SGML. Их восхитительная выдумка позволяла быстро и без особых хлопот строить системы структурной разметки самых разных видов документов - например, указывать границы и соподчинение их составных частей, выделять заголовки, подписи или математические формулы. Стандарт определял синтаксис записи управляющих элементов и их атрибутов, а также правила определения новых тегов и указания взаимоотношений между ними.

Размеченный таким образом текст могла интерпретировать любая программа, работающая на какой угодно компьютерной платформе с любым мыслимым устройством вывода. Таким образом, SGML вполне можно сравнить со скелетом, на который та или иная программа-интерпретатор будет наращивать подходящее к случаю "мясо" шрифтов, пробелов, перевода строк, пауз и интонаций.

Сам по себе язык SGML был мало востребован пока сообществу ученных по всему миру не понадобился инструмент способный быстро и легко передавать информацию. В 1991 была разработана система Gopher. Но она не поддерживала графику и гипертекстовые ссылки. Именно тогда на арену вышел Европейский институт физики элементарных частиц (CERN) в лице Тима Бернерса-Ли с проектом «Всемирная паутина».

Началась работа над созданием первого текстового браузера поддерживающего гипертекст. Новорожденный язык призванный обеспечить легкость и оперативность доставки информации был назван HTML. Язык в много повторял своего прародителя, кроме отличия в том что некоторые элементы структуры должны были отображаться особенным образом (жирными текстом, курсивом, с отступами…).

В сентябре 1993 года возглавляемая Марком Андрисеном группа программистов Национального центра супер компьютерных приложений выпустила первый графический браузер Mosaic. Программа начала распространятся со скоростью вируса. За первый год было установлено около двух миллионов копий программа. Программа поддерживала мышиный интерфейс и отображение картинок. Все это распространялось абсолютно бесплатно и помещалось на одну дискету.

Тем временем Тим Бернерс-Ли не почивал на лаврах, а продолжал развитие сети. В 1994 была создана организация которая была призвана стандартизировать гипертекстовую разметку. Организация была названа World Wide Web Consortium (w3c.org) и началась работа над создание спецификации для HTML 2.0. Начиная с версии 3.0 была реализована поддержка СSS которая была создана для разделения элементов отвечающих за отображение и структуру документа.

В начале 1994 года группа разработчиков браузера Mosaic покинула NCSA и совместно с Джеймсом Кларком из Silicon Graphics основала корпорацию Netscape Communications. Через полгода состоялся дебют - вышла первая версия коммерческого браузера Netscape для трех основных платформ. Менеджеры компании прекрасно понимали, что горизонты совсем недолго останутся такими безоблачными, и постарались подстелить соломки где только возможно. Чтобы отхватить как можно больший кусок потенциальной аудитории, привлечь максимальное количество новых пользователей, Netscape вводила в HTML все новые и новые усовершенствования, поддерживавшиеся, естественно, только одноименным браузером. Практически все новые теги были направлены на улучшение внешнего вида документа и расширение возможностей его форматирования.

Выпущенная летом 1996г. версия Internet Explorer 3.0, поддерживавшая практически все расширения Netscape и обладавшая симпатичным и дружественным интерфейсом, очень быстро утвердилась в качестве «второго главного браузера».

Четверные версии обоих браузеров вышли практически одновременно и не отличались друг от друга особой скоростью работы или иными параметрами. Но если Netscape нужно было покупать то Internet Explorer начал поставляться бесплатно в операционной системе Windows и стал фактически стандартом отрасли.

Формальный язык CSS.(Cascading Style Sheets) с английского переводится как каскадные таблицы стилей, - это язык, который используется как средство оформления web-страниц, а именно для работы с шрифтами, цветами, полями, таблицам, картинками, расположением элементов.

Основной целью, которая послужила для разработки языка CSS, являлось разделение разметки содержимого (которое написанного на HTML или другом языке программирования) и представления документа (написанного на CSS). Результат этого разделения помогает увеличить доступность документа, предоставить большую гибкость, а также уменьшить сложность и повторяемость в структурном содержимом, создает удобное описание дизайна и стилистики web-страницы и ее содержимого. Еще CSS дает возможность предоставить один документ в различных стилях или методах вывода (например, таких как экранное представление, чтение голосом, печать).

В 1990-х возникла необходимость стандартизировать web-инструменты, создать общие правила, с помощью которых программисты и web-дизайнеры могли бы создавать сайты. Как результат этой необходимости, появились языки HTML 4.01 и XHTML, а также стандарт CSS.

Первое упоминание о CSS было в 1994 году, когда Хоком Виум Ли предложил использовать CSS (Каскадные таблицы стилей) для стилистического оформления web-страниц. У него не сразу получилось продвинуть свою технологию, - только через пару лет ему удалось привлечь внимание к CSS. Итак, 17 декабря 1996 года опубликована первая спецификация (CSS1) и она была рекомендована к использованию Консорциумом Всемирной паутины (W3C).

После небольшого успеха положение дел у технологии CSS пошли куда лучше и 12 мая 1998 года (через 2 года) была принята рекомендация W3C для CSS2. Следующим этапом была CSS 2.1 - версия W3C от 8 сентября 2009 года, она была построена на базе CSS2, и была работой над исправлением существующих ошибок.

Сегодня актуальна версия CSS3, которая максимально расширена по сравнению с предыдущими версиями. CSS3 включают эффекты теней, а также скругленные углы у блоков. В CSS3 появилась возможность устанавливать изображения в качестве фона и использовать их как границы. Намного проще и удобнее стала работа с анимацией - теперь для ее создания достаточно возможностей самой CSS3, стало не нужно работать с языком JavaScript.

Плюсы и минусы использования CSS.

До того, как появился CSS, оформление web-страниц могло осуществляться непосредственно внутри содержимого документа, появление же технологии CSS дало возможность разделению содержания и представления документа. Благодаря этой возможности стало реальным простое применение единого стиля оформления для большинства подобных документов, и стало доступно быстрое изменение их оформления. Использование CSS в разработке сайтов имеет свои плюсы и минусы.

Достоинства CSS:

– Повышение совместимости с разными платформами за счет использования web-стандартов.

– Простота самого языка CSS вместе с принципом отделения оформления от содержания дает возможность сократить время на разработку и поддержку сайта.

– Имеется несколько вариантов дизайнов страницы для просмотра на разных устройствах. Например, дизайн на экране компьютера рассчитан на одну ширину, и будет полностью выводиться на экран, а на мобильных устройствах он будет подстраиваться к размерам экрана и некоторые элементы будут исключены от показа, также и при печати, будет печататься нужный текст, без лишнего (например, без шапки меню).

– Уменьшается время загрузки страниц web-сайта за счет переноса правил представления данных в отдельный CSS-файл. Благодаря этому браузер загружает только структуру документа, а также данные, хранимые на странице, а представление этих данных загружается браузером только один раз и может бать закеширован, - благодаря этому уменьшается трафик, время загрузки, а также нагрузка на сервер.

– Простота изменения дизайна. Один CSS управляет отображением множества HTML-страниц. Когда возникает необходимость изменить дизайн сайта, то незачем править каждую страницу. Для последующего изменения дизайна всего лишь нужно изменить CSS-файл, и как результат, изменение дизайна делается быстрее.

– CSS предоставляет дополнительные возможности форматирования, о которых при использовании только самих атрибутов даже и не приходилось мечтать.

Недостатки CSS:

– Различное отображение вёрстки в разных браузерах. Если браузеры устаревшие, то возможно, что одни и те же данные CSS по-разному ими интерпретируются.

– Необходимость исправлять не только один CSS-файл, но и теги HTML. Часто встречается на практике необходимость исправлять не только один CSS-файл, но и теги HTML, которые связаны с селекторами CSS. Иногда это значительно увеличивает время редактирования, а также и тестирования.

Итак, CSS - это сильнейший инструмент и один из основных составляющих практически любого web-сайта. Без CSS просто невозможно представить современную деятельность web-разработчиков. Сегодня CSS - это общепринятый стандарт разработки, который принимается всеми без исключения компаниями-разработчиками, что явно показывает его значимость и необходимость.

Язык программирования PHP.- это широко используемый язык сценариев общего назначения с открытым исходным кодом. Говоря проще, PHP это язык программирования, специально разработанный для написания web-приложений (сценариев), исполняющихся на Web-сервере. Аббревиатура PHP означает “Hypertext Preprocessor (Препроцессор Гипертекста)". Синтаксис языка берет начало из C, Java и Perl. PHP достаточно прост для изучения.

Преимуществом PHP является предоставление web-разработчикам возможности быстрого создания динамически генерируемых web-страниц. Важным преимуществом языка PHP перед такими языками, как языков Perl и C заключается в возможности создания HTML документов с внедренными командами PHP. Значительным отличием PHP от какого-либо кода, выполняющегося на стороне клиента, например, JavaScript, является то, что PHP-скрипты выполняются на стороне сервера. Вы даже можете сконфигурировать свой сервер таким образом, чтобы HTML-файлы обрабатывались процессором PHP, так что клиенты даже не смогут узнать, получают ли они обычный HTML-файл или результат выполнения скрипта. PHP позволяет создавать качественные Web-приложения за очень короткие сроки, получая продукты, легко модифицируемые и поддерживаемые в будущем. PHP прост для освоения, и вместе с тем способен удовлетворить запросы профессиональных программистов. Язык PHP постоянно совершенствуется, и ему наверняка обеспечено долгое доминирование в области языков web -программирования, по крайней мере, в ближайшее время. PHP: самый популярный язык для Интернета.(Hypertext PreProcessor, препроцессор гипертекста) - язык программирования, исполняемый на стороне веб-сервера, спроектированный Расмусом Лердорфом (Rasmus Lerdorf) в качестве инструмента создания динамических и интерактивных веб-сайтов.

Этот язык оказался достаточно гибким и мощным, поэтому приобрёл большую популярность и используется в проектах любого масштаба: от простого блога до крупнейших веб-приложений в Интернете:

Достоинства PHP:

– является свободным программным обеспечением, распространяемым под особой лицензией (PHP license);

– легок в освоении на всех этапах;

– поддерживается большим сообществом пользователей и разработчиков;

– имеет развитую поддержку баз данных;

– имеется огромное количество библиотек и расширений языка;

– может использоваться в изолированной среде;

– предлагает нативные средства организации веб-сессий, программный интерфейс расширений;

– является довольно полной заменой проприетарной среды ASP (Active Server Pages) от Microsoft;

– может быть развёрнут почти на любом сервере;

– портирован под большое количество аппаратных платформ и операционных систем.

Недостатки PHP:

– не подходит для создания десктопных приложений или системных компонентов;

– имеет слабые средства для работы с исключениями;

– глобальные параметры конфигурации влияют на базовый синтаксис языка, что затрудняет настройку сервера и разворачивание приложений;

– объекты передаются по значению, что смущает многих программистов, привыкших к передаче объектов по ссылке, как это делается в большинстве других языков;

– веб-приложения, написанные на PHP, зачастую имеют проблемы с безопасностью.

Язык программирования JavaScript.- это полноценный динамический язык программирования, который применяется к HTML документу, и может обеспечить динамическую интерактивность на веб-сайтах. JavaScript сам по себе довольно компактный, но очень гибкий, и разработчиками написано много инструментов поверх основного JavaScript языка, которые разблокируют огромное количество дополнительных функций с очень небольшим усилием. К ним относятся: Программные Интерфейсы приложения (API) встроенные в браузеры, обеспечивающие различные функциональные возможности, такие как динамическое создание HTML и установку CSS стилей, захват и манипуляция видеопотоком, работа с веб-камерой пользователя или генерация 3D графики и аудио сэмплов. Сторонние API позволяют разработчикам внедрять функциональность в свои сайты от других разработчиков, таких как Twitter или Facebook.

Достоинства JavaScript:

– Ни один современный браузер не обходится без поддержки JavaScript.

– С использованием написанных на JavaScript плагинов и скриптов справится даже не специалист.

– Полезные функциональные настройки.

– Постоянно совершенствующийся язык - сейчас разрабатывается бета-вариация проекта, JavaScript2.

– Взаимодействие с приложением может осуществляется даже через текстовые редакторы - Microsoft Office и Open Office.

– Перспектива использования языка в процессе обучения программированию и информатике.

Недостатки JavaScript:

– Пониженный уровень безопасности ввиду повсеместного и свободного доступа к исходным кодам популярных скриптов.

– Множество мелких раздражающих ошибок на каждом этапе работы. Большая часть из них легко исправляется, но их наличие позволяет считать этот язык менее профессиональным, сравнительно с другими.

– Повсеместное распространение. Своеобразным недостатком можно считать тот факт, что часть активно используемых программ (особенно приложений) перестанут существовать при отсутствии языка, поскольку целиком базируются на нем.

Язык программирования MySQL.- это одна из самых популярных и самых распространенных СУБД (система управления базами данных) в интернете. Она не предназначена для работы с большими объемами информации, но ее применение идеально для интернет сайтов, как небольших, так и достаточно крупных. MySQL отличатся хорошей скоростью работы, надежностью, гибкостью. Работа с ней, как правило, не вызывает больших трудностей. Поддержка сервера MySQL автоматически включается в поставку PHP. Немаловажным фактором является ее бесплатность.распространяется на условиях общей лицензии GNU (GPL, GNU Public License). Ранее для долговременного хранения информации мы работали с файлами: помещали в них некоторое количество строчек, а затем извлекали их для последующей работы. Задача длительного хранения информации очень часто встречается в программировании Web-приложений: подсчёт посетителей в счётчике, хранение сообщений в форуме, удалённое управление содержанием информации на сайте и т.д.

Между тем, профессиональные приёмы работы с файлами очень трудоёмки: необходимо заботится о помещении в них информации, о её сортировке, извлечении, при этом не нужно забывать, что все эти действия будут происходить на сервере хост-провайдера, где с очень большой вероятностью стоит один из вариантов Unix - следовательно, нужно так же заботится о правах доступа к файлам и их размещении. При этом объём кода значительно возрастает, и совершить ошибку в программе очень просто. Все эти проблемы решает использование базы данных. Базы данных сами заботятся о безопасности информации и её сортировке и позволяют извлекать и размещать информацию при помощи одной строчки. Код с использованием базы данных получается более компактным, и отлаживать его гораздо легче. Кроме того, не нужно забывать и о скорости - выборка информации из базы данных происходит значительно быстрее, чем из файлов.

Достоинства MySQL:

– Простота в работе - установить MySQL довольно просто. Дополнительные приложения, например GUI, позволяет довольно легко работать с БД.

– Богатый функционал - MySQL поддерживает большинство функционала SQL.

– Безопасность - большое количество функций обеспечивающих безопасность, которые поддерживается по умолчанию.

– Масштабируемость - MySQL легко работает с большими объемами данных и легко масштабируется.

– Скорость - упрощение некоторых стандартов позволяет MySQL значительно увеличить производительность.

Недостатки MySQL:

– Известные ограничения - по задумке в MySQL заложены некоторые ограничения функционала, которые иногда необходимы в особо требовательных приложениях.

– Проблемы с надежностью - из-за некоторых способов обработки данных MySQL (связи, транзакции, аудиты) иногда уступает другим СУБД по надежности.

– Медленная разработка - Хотя MySQL технически открытое ПО, существуют жалобы на процесс разработки. Стоит заметить, что существуют другие довольно успешные СУБД созданные на базе MySQL, например MariaDB.

## 2.2 Описание и разработка приложения

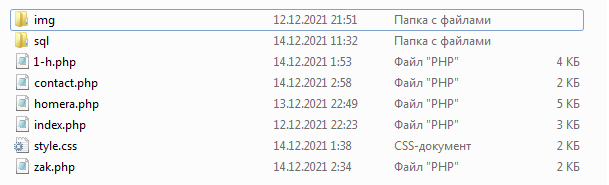


Рисунок 6. Полный файл.

Файл index.php. Первая страница, которую видит пользователь при заходе на сайт.

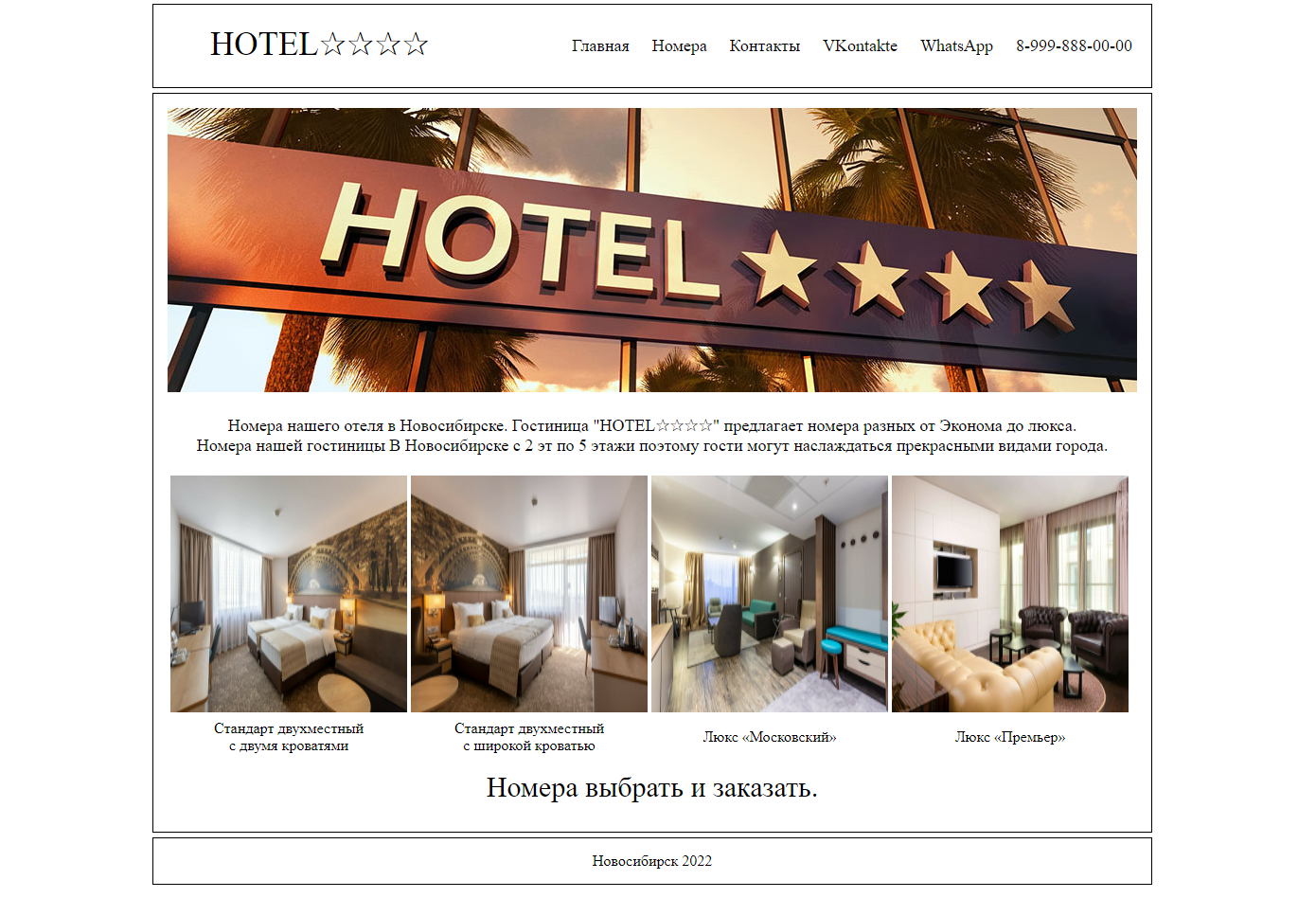


Рисунок 7. Главная страница сайта index.php.

Файл homera.php. На странице представлена информация о доступных номерах и их краткое описание.

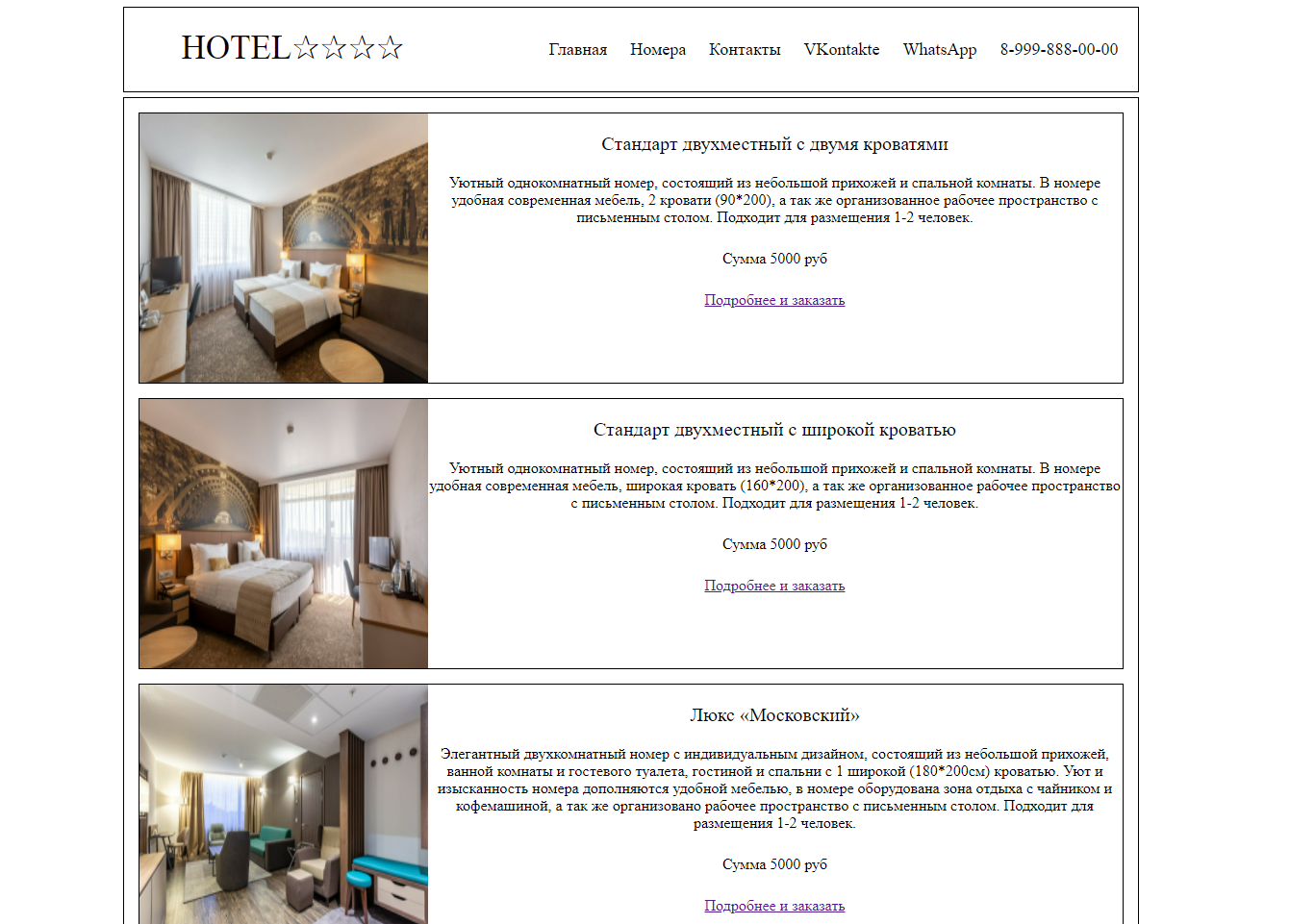


Рисунок 8. Страница «Номера и Цены» homera.php.

Файл contact.php. Содержит контактный телефон, адрес.

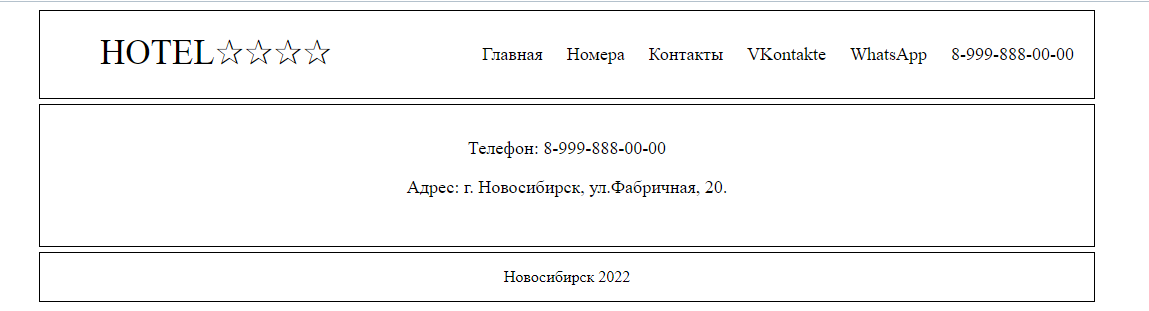


Рисунок 9. Страница «Контакты» contact.php.

Файл 1-h.php. На странице представлена информация о доступных номерах и их полное описание. Бронировать и заказать номер.

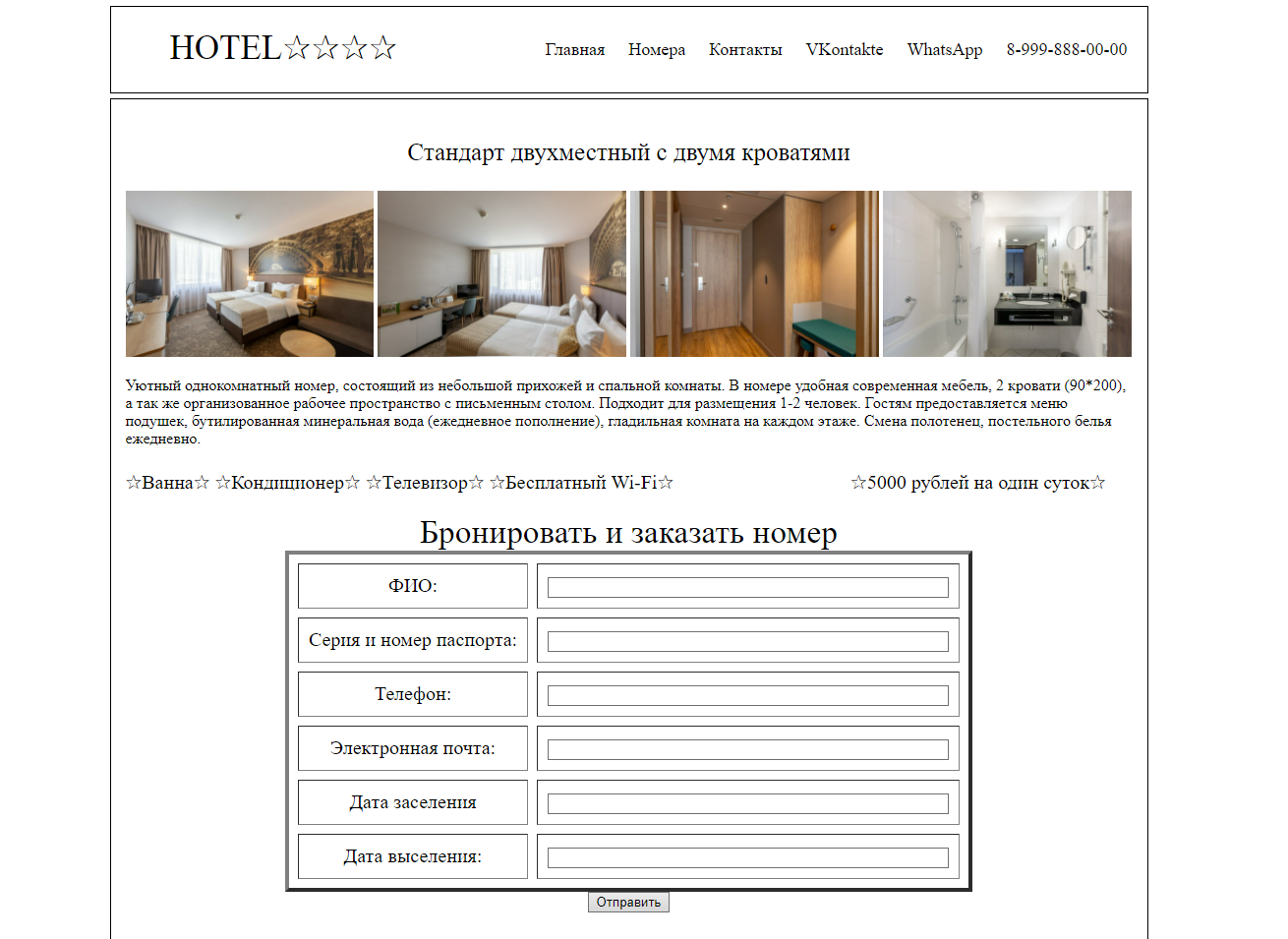


Рисунок 10. Страница «Подробнее и заказать» 1-h.php.

# ЗАКЛЮЧЕНИЕ

Результатом выполнения курсового проекта стал разработанный web-сайт гостиницы. Исходя из основных задач, была проведена детальная работа над структурой сайта, макетами web-страниц, навигацией по сайту. Макеты заполнены содержимым, в них внедрены графическое наполнение и дополнительные компоненты, обеспечивающие реализацию функциональных возможностей сайта. Протестирована функциональность сайта с точки зрения удобства навигации, проверены корректность работы кода в различных браузерах, орфография и пунктуация текстов, корректность гиперссылок.

Пользовательский интерфейс спроектирован максимально удобно и продуманно. Это касается как удобства расположения отдельных элементов и их использования, которое не требует какого-либо обучения или привыкания, интерфейс интуитивно понятен, так и подбора фона и цвета различных элементов, которые могут быть изменены по первому требованию генерального директора предприятия.

В процессе выполнения выпускной квалификационной работы были получены следующие результаты:

– изучена специфика работы сайта гостиницы и обоснована необходимость наличия сайта;

– проанализированы сайты аналогичных компаний;

– выбрана инструментальная среда для разработки сайта;

– спроектированы макет сайта и базы данных;

– реализованы разработанные схемы сайта и базы данных с помощью программного обеспечения;

– изучены языки программирования и технологии HTML, CSS, PHP, JavaScript, MySQL;

К недостаткам работы можно отнести следующие: отсутствие кнопки поиск по сайту, автоматического оповещения по электронной почте и возможности заказа дополнительных услуг. Данный недостаток является несущественным и не влияет на общую оценку работы.

Могут быть указаны следующие пути совершенствования:

– совершенствование системы бронирования;

– добавление более актуального материала;

– расширение таблиц базы данных;

– автоматизация оповещения клиента при помощи дополнительного сервера электронной почты

– установка более высокого уровня защиты для панели администратора.

Таким образом, проект можно считать успешно выполненным, так как большинство запланированных функций и возможностей были реализованы на сайте, и данная работа определенно вложит свою долю в развитие гостиницы, позволит ей увеличить свою прибыль и привлекать новых клиентов.

Вот веб сайт, ссылка «<http://217.71.129.139:4268/>»

Все файлы в гитхабе «<https://github.com/9I-DAMAGER/Website>».

# СПИСОК ЛИТЕРАТУРЫ

1. Гаевский А.Ю. 100% самоучитель. Создание Web-страниц и Web-сайтов. HTML и JavaScript / А.Ю. Гаевский, В.А. Романовский. - М.: Триумф, 2016. - 464c.
2. Дакетт Дж. HTML и CSS. Разработка и дизайн веб-сайтов (+ CD-ROM) / Дж. Дакетт. - М.: Эксмо, 2013. - 480c.
3. Дебольт В. HTML и CSS. Совместное использование / Дебольт Вирджиния. - М.: НТ Пресс, 2013. - 512c.
4. Дронов В. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов / В. Дронов. - СПб. : БХВ-Петербург, 2014. - 904c.
5. Дронов В. JavaScript в Web-дизайне / В. Дронов. - М.: СПб: БХВ, 2015. - 880c.
6. Кузнецов М.В. PHP. Практика создания Web-сайтов / Кузнецов М.В. и. - СПб.: БХВ-Петербург, 2012. - 716c.
7. Маркин А.В. Основы Web-программирования на PHP / А.В. Маркин, С.С. Шкарин. - Москва: ИЛ, 2012. - 256c.

# Приложение

Листинг кода главной страницы «index.php»

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>HOTEL✰✰✰✰</title>

<link rel="stylesheet" href="style.css" type="text/css"/>

</head>

<body>

<header>

<div class="container">

<a href="index.php" class="logo">HOTEL✰✰✰✰</a>

<nav>

<ul>

<li><a href="index.php">Главная</a></li>

<li><a href="homera.php">Номера</a></li>

<li><a href="contact.php">Контакты</a></li>

<li><a href="">VKontakte</a></li>

<li><a href="">WhatsApp</a></li>

<li><p>8-999-888-00-00</p></li>

</ul>

</nav>

</div>

</header>

<div class="main">

<div class="container">

<div class="row">

<div class="col-img"><img src="img/gos.jpg"></div>

</div>

<div class="row">

<div class="col-1">

<p>Номера нашего отеля в Новосибирске. Гостиница "HOTEL✰✰✰✰" предлагает номера разных от Эконома до люкса.

</br>Номера нашей гостиницы В Новосибирске с 2 эт по 5 этажи поэтому гости могут наслаждаться прекрасными видами города.</p>

</div>

<div class="col-1-2">

<table>

<tr>

<td><img class="col-1-22" src="img/1/1.jpg"/></td>

<td><img class="col-1-22" src="img/2/1.jpg"/></td>

<td><img class="col-1-22" src="img/3/1.jpg"/></td>

<td><img class="col-1-22" src="img/4/1.jpg"/></td>

</tr>

<tr>

<td id="col">Стандарт двухместный </br>с двумя кроватями</td>

<td id="col">Стандарт двухместный </br>с широкой кроватью</td>

<td id="col">Люкс «Московский»</td>

<td id="col">Люкс «Премьер»</td>

</tr>

</table>

</div>

<div class="zak"><a id="zak" href="homera.php">Номера выбрать и заказать.</a></div>

</div>

</div>

</div>

<footer>

<div class="container">

<div class="col-end">Новосибирск 2022</div>

</div>

</footer>

</body>

</html>

Листинг кода «Подробнее и заказать» страницы сайта «1-h.php».

<div class="col-zakk"><center>

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

<table border="4" cellspacing="9" cellpadding="10" style="font-size:15pt;">

<caption style="font-size:25pt;"> Бронировать и заказать номер </caption>

<TBODY>

<tr align="center">

<td><label for=name>ФИО:</label></td>

<td><input style="width:400px;" name=name></td>

</tr>

<tr align="center">

<td><label for=name> Серия и номер паспорта: </label></td>

<td><input style="width:400px;" name=pasport></td>

</tr>

<tr align="center">

<td><label for=name>Телефон:</label></td>

<td><input style="width:400px;" name=tel></td>

</tr>

<tr align="center">

<td><label for=name>Электронная почта:</label></td>

<td><input style="width:400px;" name=mail></td>

</tr>

<tr align="center">

<td><label for=data> Дата заселения </label></td>

<td><input style="width:400px;" name=dat1></td>

</tr>

<tr align="center">

<td><label for=data> Дата выселения: </label></td>

<td><input style="width:400px;" name=dat2></td>

</tr>

</table>

<input name="B1" value="Отправить" type="submit">

</form>

</div>

Листинг кода Отправиить информацию на базу данных «zak.php».

<?php

//Устанавливаем доступы к базе данных:

$host = 'localhost'; //имя хоста, на локальном компьютере это localhost

$user = 'root'; //имя пользователя, по умолчанию это root

$password = 'root'; //пароль, по умолчанию пустой

$db\_name = 'gos'; //имя базы данных

//Соединяемся с базой данных используя наши доступы:

$link = mysqli\_connect($host, $user, $password, $db\_name);

//Устанавливаем кодировку (не обязательно, но поможет избежать проблем):

mysqli\_query($link, "SET NAMES 'utf8'");

//Формируем тестовый запрос:

$query = "REPLACE homer SET Name = '".$\_POST['name']."',

Pasport = '".$\_POST['pasport']."',

Tel = '".$\_POST['tel']."',

Mail = '".$\_POST['mail']."',

Dat1 = '".$\_POST['dat1']."',

Dat2 = '".$\_POST['dat2']."'";

//Делаем запрос к БД, результат запроса пишем в $result:

$result = mysqli\_query($link, $query) or die(mysqli\_error($link));

//Проверяем что же нам отдала база данных, если null – то какие-то проблемы:

var\_dump($result);

?>

<!doctype html>

<html>

<head>

<title></title>

</head>

<body>

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

<input value="Главная страница" type="submit">

</form>

</body>

</html>