МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГАОУ ВО «СЕВЕРО–КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И ТЕЛЕКОММУНИКАЦИЙ

КАФЕДРА ПРИКЛАДНОЙ ИНФОРМАТИКИ

**КУРСОВАЯ РАБОТА**

по дисциплине

«ИНТЕРНЕТ–ТЕХНОЛОГИИ»

на тему:

«Разработка сервиса бронирования отелей»

**Выполнил:**

Куделя Максим Владимирович

Студент 3 курса группы ПИН–б–о–17–1

Направления подготовки

09.03.03 Прикладная информатика

очной формы обучения

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

**Руководитель работы:**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(ФИО, должность, кафедра)

Работа допущена к защите\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя) (дата)

Работа выполнена и

защищена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дата защиты\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Члены комиссии: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(должность) (подпись) (И.О. Фамилия)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ставрополь, 2020 г.

УТВЕРЖДАЮ

Зав. кафедрой ПИ

Азаров И.В.

**Институт Информационных Технологий и Телекоммуникаций**

**Кафедра Прикладной информатики**

**Направление подготовки 09.03.03 Прикладная информатика**

**Направленность (профиль) Прикладная информатика в экономике**

**ЗАДАНИЕ**

**на курсовую работу**

студента Куделя Максима Владимировича по дисциплине Интернет–технологии

1. Тема работы: разработка сервиса бронирования отелей
2. Цель: разработка сервиса бронирования отелей, формирование навыков научно–исследовательской работы, повышение уровня профессиональной подготовки, углубление знаний по учебной дисциплине.
3. Задачи:
   1. Спроектировать базу данных;
   2. Нарисовать макет дизайна сайта;
   3. Сверстать сайт;
   4. Разработать серверную часть.

4. Перечень подлежащих разработке вопросов:

а) по теоретической части Фреймворки и библиотеки JavaScript

б) по практической части разработка сервиса бронирования отелей

5. Исходные данные:

а) по литературным источникам: Weil A. Learn ASP.NET MVC/ Weil A.– lulu.com, 2016; Rees J. ASP.NET MVC 5: Your First Guide – from Zero Steps to Confident User,2017; Fagerberg J. Asp.net Mvc 5 – Building a Website With Visual Studio 2015 and C Sharp: The Tactical Guidebook, 2016.

б) по вариантам, разработанным преподавателем

6. Список рекомендуемой литературы:

7. Контрольные сроки представления отдельных разделов курсовой работы:

25 % –\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020 г.

50 % –\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2020г.

75 % –\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020г.

100 % –\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020г.

8. Срок защиты студентом курсовой работы «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020г.

Дата выдачи задания «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020г.

Руководитель курсовой работы

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(ученая степень, звание) (личная подпись) (инициалы, фамилия)*

Задание принял к исполнению студент очной формы обучения 3 курса группы ПИН–б–о–17–1 \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(личная подпись) (инициалы, фамилия)*

**Отзыв**

на курсовой проект студента 3 курса

Куделя Максима Владимировича

тема

Актуальность: Курсовая работа посвящена разработке веб – приложения. У любой современной компании существует [сайт](https://cetera.ru/webdevelopment/). Это один из элементов престижа, ведь именно в Интернете потенциальные клиенты будут в первую очередь искать информацию о фирме. И если у нее нет хотя бы одной страничка с прайсом, это покажется подозрительным – насколько же это неуспешная фирма, если не может даже небольшой веб-ресурс создать? Компания может донести информацию максимально быстро до огромного количества людей с помощью собственного сайта сделать. Веб-ресурс позволяет представить информацию о компании и ее товарах или услугах сжато и одновременно полноценно.

В первой главе рассмотрено понятие фреймворков и библиотек java script, проведено их сравнение, а также преимущества и недостатки. Были проанализированы и описаны наиболее популярные фреймворки и библиотеки.

Во второй главе был спроектирован и разработан сервис бронирования отелей.

Выводы, сделанные в Заключении, соответствуют целям, поставленным во введении.

Проанализирована литература в объеме 7 источников.

За время работы студент проявил себя как квалифицированный специалист в области программирования, способный к самостоятельному поиску литературных источников, анализу информации и синтезу новых решений.

Таким образом, работа выполнена на высоком уровне, соответствует требованиям, предъявляемым к курсовым проектам, и заслуживает оценки

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Научный руководитель:

"…….."………..2020г.

Содержание

[Введение 6](#_Toc40788749)

[1. Теоретическая часть 7](#_Toc40788750)

[1.1 О java-script 7](#_Toc40788751)

[1.2 Фреймворки и их преимущества 7](#_Toc40788752)

[1.3 Библиотеки и их отличия от фреймвокров 8](#_Toc40788753)

[1.4 Популярные фреймворки java-script 9](#_Toc40788754)

[1.5 Популярные библиотеки java-script 10](#_Toc40788754)

[2. Практическая часть 12](#_Toc40788755)

[2.1 Нарисуйте дизайн макет интернет магазина. 12](#_Toc40788756)

[2.2 Разработать клиентскую часть интернет магазина. 16](#_Toc40788757)

[2.3 Разработайте серверную часть интернет магазина. 32](#_Toc40788758)

[Заключение 33](#_Toc40788759)

[Список используемой литературы 34](#_Toc40788760)

### **Введение**

В настоящее время интернет является составной частью развлекательной сферы деятельности. Во время его использования мы видим миллионы интернет–магазинов, реклам. Следовательно, существует очень сильная конкуренция между предпринимателями. Очень важно правильно структурировать сайт, чтобы пользователю хотелось вернуться и совершить еще покупку. А так как информационные технологии развиваются очень быстро, то тяжело представить магазин без собственного сайта.

Объектом курсовой работы является веб–приложение «разработка сервиса бронирования отелей», предмет – инструменты разработки и поддержания работоспособности веб–приложения. Практическая значимость заключается в проектирование и разработке сайта на технологии ASP.NET.

Цели курсовой работы: разработка сайта для сервиса бронирования отелей, формирование навыков научно–исследовательской работы, повышения уровня профессиональной подготовки, углубление знаний по учебной дисциплине.

Для достижения поставленных целей были поставлены следующие задачи:

1. Спроектировать базу данных;
2. Нарисовать макет дизайна сайта;
3. Разработать клиентскую часть;
4. Разработать серверную часть.
5. **Теоретическая чаcть Фреймворки и библиотеки java-script**
   1. **О java-script**

JavaScript – это полноценный [динамический язык программирования](https://developer.mozilla.org/ru/docs/%D0%A1%D0%BB%D0%BE%D0%B2%D0%B0%D1%80%D1%8C/Dynamic_programming_language), который применяется к [HTML](https://developer.mozilla.org/ru/docs/%D0%A1%D0%BB%D0%BE%D0%B2%D0%B0%D1%80%D1%8C/HTML) документу, и может обеспечить динамическую интерактивность на веб-сайтах. Его разработал Brendan Eich, сооснователь проекта Mozilla, Mozilla Foundation и Mozilla Corporation.[6]

JavaScript существует уже более 20 лет, и это один из тех языков, которые никогда не перестают развиваться. В последнее время среда JavaScript стала просто огромной. Изначально JavaScript был создан, чтобы «сделать веб-страницы живыми».[3]

Программы на этом языке называются скриптами. Они могут встраиваться в HTML и выполняться автоматически при загрузке веб-страницы.

Скрипты распространяются и выполняются, как простой текст. Им не нужна специальная подготовка или компиляция для запуска.

Три сильные стороны JavaScript:

* Полная интеграция с HTML/CSS.
* Простые вещи делаются просто.
* Поддерживается всеми основными браузерами и включён по умолчанию.

JavaScript – это единственная браузерная технология, сочетающая в себе все эти три вещи. [6]

* 1. **Фреймворки и их преимущества**

Фреймворки представляют собой программные продукты по типу неких библиотек, наборов определенных кодов, служащих основой для создания уникальных механизмов и функционала сайта.

Это программная среда специального назначения, своеобразный каркас, используемый для того, чтобы существенно облегчить процесс объединения определенных компонентов при создании программ. Это основа, которая позволяет добавлять компоненты в зависимости от потребностей. База, на которой можно сформировать программу любого назначения достаточно быстро и без особых затруднений.

С помощью фреймворка можно создать не только публичную часть ресурса, но и спроектировать базу данных, построить модули новой системы на основе требуемых уникальных алгоритмов, в которые впоследствии гораздо проще вносить коррективы при необходимости, не «разбирая» и не «перенастраивая» затем всю систему. Также на фреймфорке разрабатывается административный интерфейс управления сайтом.

Таким образом, фреймворк универсален и более многозадачен. С помощью фреймворка можно достичь полностью индивидуального результата. Кроме того, составляя код из мелких «кирпичиков», можно полностью исключить лишнее и увеличить скорость работы и эффективность веб-проекта.[2]

* 1. **Библиотеки и их отличия от фреймворков**

Библиотека — это более простой компонент архитектуры программного обеспечения. Программная библиотека может быть использована просто как набор подсистем близкой функциональности, не влияя на архитектуру основного программного продукта и не накладывая на неё никаких ограничений.

Фреймворк же не просто даёт разработчику нужный функционал, но ещё и диктует правила построения архитектуры приложения, задавая на начальном этапе разработки поведение по умолчанию, формируя каркас, который нужно будет расширять и изменять согласно указанным требованиям. Фреймворк также может включать вспомогательные программы, язык сценариев и другое ПО, облегчающее разработку и объединение разных компонентов большого программного проекта.

Также, в отличие от библиотеки, которая объединяет в себе набор близкой функциональности, – фреймворк может содержать в себе большое число разных по тематике библиотек.[1]

* 1. **Популярные фреймворки java-script**

### **Angular**

Этот [фреймворк](https://angularjs.org/) от Google, ставший очень популярным в последние годы, идеален для разработчиков, которые с большим вниманием относятся к последним новинкам отрасли. Angular представляет из себя набор современных функций для разработки и проектирования, обеспечивающих высокую скорость разработки. Компания Google даже [создала](https://material.angularjs.org/latest/) специальный компонент Angular Material, позволяющий легко реализовать настоящий Material Design с помощью Angular.

Angular 2 [добрался](https://tproger.ru/news/angular-2-0-release/) до релиза в сентябре прошлого года. Позже Google [анонсировала](https://tproger.ru/news/angular-3-is-around-the-corner/) выпуск Angular 3, но затем [сообщила](https://tproger.ru/news/angular-4/) о том, что сразу займется четвертой версией фреймворка, которая в марте [стала доступна](https://tproger.ru/news/angular-4-0-0-released/) для скачивания.

### **Vue**

[Vue](https://vuejs.org/) — это простой в использовании и при этом очень мощный фреймворк. Начинающие по достоинству оценят документацию. Vue берёт всё лучшее от других фреймворков и соединяет в один удобный инструмент. Так же, как и React, Vue работает с однофайловыми компонентами и однонаправленными связями между ними, но работать с ним проще благодаря CLI-инструменту [vue-cli](https://github.com/vuejs/vue-cli" \t "_blank). Внутри Vue-компонентов можно использовать препроцессоры (Jade, Stylus и др.)

### **Ember**

[Ember](http://emberjs.com/) — современный JavaScript-фреймворк для амбициозных разработчиков и проектов. Будучи невероятно мощным, он очень прост в освоении.

Одной из вещей, за которые любят Ember, является шаблонизатор Handlebars, который, по словам разработчиков, намного элегантней директив Angular.

### **Backbone**

Очень легкий и стильный фреймворк с MV\*-структурой, предназначенный для создания SPA.

[Backbone](http://backbonejs.org/) имеет всего одну зависимость – это библиотека Underscore.

Фреймворк обладает богатой экосистемой, которая в сочетании с Mustache и Marionette позволяет создавать полноценные клиентские приложения.[4]

* 1. **Популярные библиотеки java-script**

**React**

React - это библиотека JavaScript, созданная разработчиками Facebook и Instagram. Согласно опросу Stack Overflow Survey 2017, React был признан самой популярной технологией среди разработчиков. React также имеет честь быть самым популярным проектом JavaScript, согласно количеству звезд на GitHub.

Так почему React получает все внимание? С помощью React можно создать интерактивный интерфейс с использованием декларативного подхода*,* в котором вы можете контролировать состояние приложения, говоря: «Представление должно выглядеть так». Он использует компонентную модель, в которой компоненты являются повторно используемыми элементами пользовательского интерфейса, и каждый компонент имеет свое собственное состояние.

React использует виртуальную модель DOM, так что вам не нужно беспокоиться о прямом манипулировании с DOM. Другие примечательные особенности React включают однонаправленный поток данных, дополнительный синтаксис JSX и инструмент командной строки для создания проекта React с нуля.[7]

### [**JQuery**](https://jquery.com/)

jQuery - это библиотека, которая сделала JavaScript более доступным а DOM-манипуляцией проще, чем раньше. Плавная кривая обучения jQuery и простой синтаксис породили на стороне клиента новое поколение новых разработчиков. Несколько лет назад jQuery считался прочным решением для создания надежных веб-сайтов с поддержкой кросс-браузерности. Основные функции jQuery, такие как манипулирование DOM на основе селекторов CSS, обработка событий и создание вызовов AJAX, подпитывали его популярность.

Однако все изменилось, и среда JavaScript постоянно развивается. Некоторые функции jQuery были включены в новую спецификацию ECMAScript. Более того, новые библиотеки и фреймворки, используемые сегодня, имеют собственный способ связывания DOM, и поэтому простые методы манипуляции с DOM больше не требуются. [7]

### [**D3: Документы, управляемые данными**](https://d3js.org/)

D3 (или D3.js) - мощная библиотека JavaScript для создания интерактивных визуализаций с использованием веб-стандартов, таких как SVG, HTML и CSS. В отличие от других библиотек визуализации, D3 предлагает лучший контроль над окончательным визуальным результатом.

D3 работает, связывая данные с DOM и затем преобразуя их в документ. Она также имеет собственную экосистему, которая состоит из плагинов и библиотек, которые расширяют ее базовые функциональные возможности.[7]

1. **Практическая часть**
   1. **Разработка дизайна ­**– **макета сайта бронирования отелей.**

Для проектирования макета сайта будем использовать онлайн ­– сервис «Figma». Нарисуем макеты страниц сайта: «Главная», «Город», «Отель».

Страница «Главная» состоит из хэдера, на котором будут отображаться название сайта, а также кнопки регистрации и авторизации, основного раздела и футера. Результат представлен на рисунке 2.1.

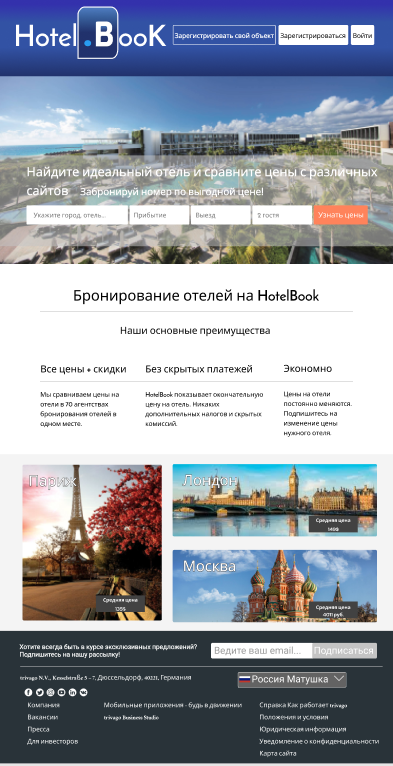


Рисунок 2.1 – Страница «Главная»

Страница «Город» также содержит хэдер и футер, а также главную область с отелями, которые находятся в этом городе. Результат представлен на рисунке 2.2.

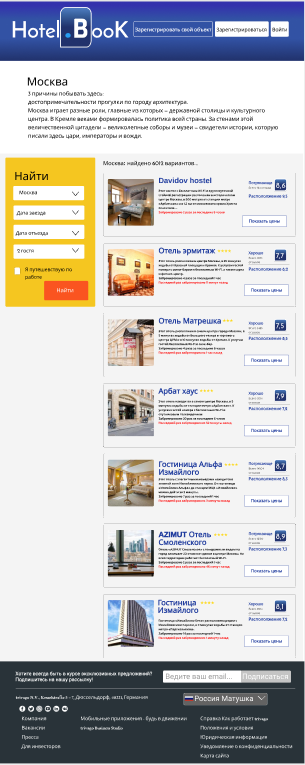


Рисунок 2.2 – Страница «Город»

Страница «Город» также содержит хэдер и футер. В основной части размещены: информация об отеле, номера в виде таблицы, список предоставляемых услуг. Результат показан на рисунках 2.3-2.4.



Рисунок 2.3 – Страница «Город»

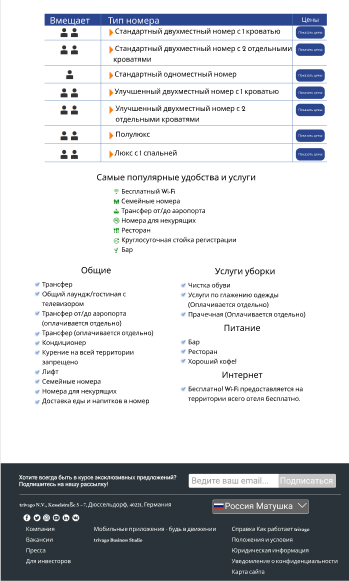


Рисунок 2.4 – Страница «Город»

* 1. **Разработать клиентскую часть сайта**

Для создания сайта будем использовать платформу ASP.NET.

Создадим хедер для нашего сайта. Так как хедер будет фиксированным и присутствовать на каждой странице, определим его в layout page (Листинг 1).

Листинг – 1 разметка хедера.

*<body >*

*<div>*

*<div id="head">*

*<div id="left">*

*<img id="Logo" src="../../Content/img/Logo.png">*

*</div>*

*<div id="right">*

*<div class="wrapper">*

*<input type="checkbox" id="callback">*

*<label class="btn" for="callback">Регистрация/Вход</label>*

*<form class="popup">*

*<span>Введите ваши данные</span>*

*<input type="text" name="Name" placeholder="Имя">*

*<input type="text" name="Pass" placeholder="Пароль">*

*<button class="default" type="submit" formaction="/Home/AddUser" formmethod="post">Регистрация</button>*

*<button class="default" type="submit" formaction="/Home/LogUser" formmethod="post">Вход</button>*

*<label class="close" for="callback">+</label>*

*</form>*

*</div>*

*<div class="wrapper">*

*@if (User.Identity.IsAuthenticated)*

*{*

*<a href="/Home/UserPriofile" class="btn">@ViewBag.UserName</a>*

*}*

*else*

*{*

*<a class="btn" onclick="Info()">Не вошел</a>*

*}*

*</div>*

*</div>*

*</div>*

*@RenderBody()*

*</div>*

*</body>*

Дальше подключим стили (Листинг 2).

Листинг 2 – css стили для хедера.

*div#head {*

*padding-top: 30px;*

*margin-bottom: 0px;*

*background: linear-gradient(to bottom, rgba(10,39,107,1) 100%, rgba(11,21,168,1) 0%, rgba(12,36,165,1) 10%,rgba(15,68,159,1) 32%, rgba(15,68,159,1) 62%);*

*width: 100%;*

*height: 100px;*

*font-family: Reem Kufi;*

*font-style: normal;*

*font-weight: normal;*

*font-size: 288px;*

*line-height: 432px;*

*font: 50px Arial;*

*color: #ffffff;*

*}*

*#left{*

*width: 50%;*

*float: left;*

*}*

*#right{*

*margin-left: 70%;*

*margin-top: 18px;*

*}*

В хедере будет присутствовать всплывающая форма регистрации и авторизации. Для нее используются следующие стили css (Листинг 3):

Листинг 3 – css стили формы авторизации.

.*wrapper {*

*max-width: 500px;*

*max-height: 50px;*

*margin: 40px auto;*

*text-align: center;*

*position: relative;*

*}*

*.btn {*

*font-size: small;*

*font-family: Arial;*

*display: block;*

*margin: 0 auto;*

*width: 150px;*

*border: 2px solid #fff;*

*padding: 1.5%;*

*margin: 0 2%;*

*color: #fff;*

*font-family: arial;*

*cursor: pointer;*

*transition: .3s all;*

*float: left;*

*}*

*.btn:hover {*

*background-color: rgba(207, 207, 207, 0.527);*

*color: #FFF;*

*}*

*.popup {*

*display: block;*

*width: 260px;*

*background: rgba(10,39,107,1);*

*border-top: rgba(255, 255, 255, 0.69) solid 0.5px;*

*padding: 30px;*

*text-transform: uppercase;*

*font-family: arial;*

*color: white;*

*font-size: 14px;*

*position: absolute;*

*left: 50%;*

*transform: translate(-50%);*

*top: -999px;*

*transition: .5s all linear;*

*}*

*.popup input{*

*width: 100%;*

*border: 0;*

*margin: 10px 0;*

*height: 35px;*

*padding: 5px;*

*outline: none;*

*}*

*.submit{*

*background-color: #fff;*

*cursor: pointer;*

*}*

*#callback{*

*display: none;*

*}*

*#callback:checked ~ .popup{*

*top: 60px;*

*}*

*.close{*

*display: block;*

*position: absolute;*

*transform: rotate(45deg);*

*font-size: 24px;*

*top: 5px;*

*right: 10px;*

*cursor: pointer;*

*}*

При авторизации пользователя его логин будет выводиться в виде текста на кнопке профиля. если же пользователь не авторизовался, на кнопке будет присутствовать текст «Не вошел».

В результате получим заголовок, показанный на рисунке 2.5.



Рисунок 2.5 – Хедер

Теперь создадим футер (Листинг 4).

Листинг 4 – html разметка футера.

*<div id="footer">*

*<div id="footer\_head">*

*<div id="left">*

*<p id="w">Хотите всегда быть в курсе эксклюзивных предложений? Подпишитесь на нашу рассылку!</p>*

*</div>*

*<div id="right">*

*<input class="default" placeholder="Введите свою почту">*

*<button class="default">Подтвердить</button>*

*</div>*

*</div>*

*<hr>*

*<div id="footer\_body">*

*<div class="center">*

*<div id="left">*

*<p id="w">trivago N.V., Kesselstraße 5 – 7, Дюссельдорф, 40221, Германия</p>*

*<img src="../../Content/img/Symbols.jpg" width="150" height="30px">*

*</div>*

*<div id="right">*

*<p id="w">Выбрать язык:</p>*

*<select class="default">*

*<option>Россия Матушка</option>*

*<option>English</option>*

*</select>*

*</div>*

*</div>*

*<div id="footer\_end">*

*<div id="k1">*

*<p id="w">*

*Компания*

*Вакансии*

*Пресса*

*Для инвесторов*

*</p>*

*</div>*

*<div id="k2">*

*<p id="w">*

*Мобильные приложения - будь в движении trivago Business Studio*

*</p>*

*</div>*

*<div id="k3">*

*<p id="w">*

*СправкаКак работает trivago*

*Положения и условия*

*Юридическая информация*

*Уведомление о конфиденциальности*

*Карта сайта*

*</p>*

*</div>*

*</div>*

*</div>*

*</div>*

Применим футеру уже готовые стили css, а так же добавим новые(Листинг 5).

Листинг 5 – css стили футера.

*div.center{*

*margin: 0px;*

*padding: 0px;*

*margin-left: 8%;*

*margin-right: 8%;*

*height: 100px;*

*text-align: center;*

*}*

*div#footer{*

*padding-bottom: 0%;*

*background-color: #293339;*

*height: 300px;*

*}*

*div#footer\_head{*

*margin-left: 7.5%;*

*margin-right: 7.5%;*

*padding: 10px;*

*}*

*div#footer\_end{*

*margin-left: 8%;*

*margin-right: 8%;*

*float: left;*

*}*

*div#k1{*

*width : 14%;*

*float: left;*

*}*

*div#k2{*

*padding-left: 10%;*

*width : 20%;*

*float: left;*

*}*

*div#k3{*

*padding-left: 16%;*

*width : 40%;*

*float: left;*

*}*

В результате получим футер, показанный на рисунке 2.6.

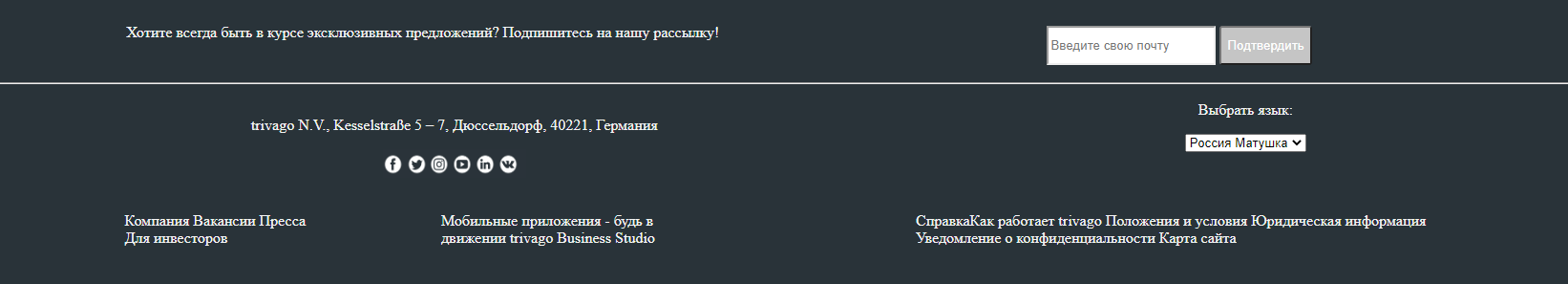


Рисунок 2.6 – Футер

Теперь создадим главную страницу сайта. Она будет состоять из заголовка (листинг 5) и основной части (Листинг 6).

Листинг 5 – html разметка заголовка главной страницы сайта.

*<div id="seach\_main">*

*<div id="seach\_second">*

*<p id="w\_b">Найдите идеальный отель и сравните цены с различных сайтов Забронируй номер по выгодной цене!</p>*

*<div class="center">*

*<input class="default" id="rad" placeholder="Укажите город, отель...">*

*<input class="default" id="rad" placeholder="Прибытие">*

*<input class="default" id="rad" placeholder="Выезд">*

*<input class="default" id="rad" placeholder="Число гостей">*

*<button class="default" id="orange">Найти</button>*

*</div>*

*</div>*

*</div>*

Листинг 6 – html разметка главной страницы сайта.

*<div id="body">*

*<div id="info">*

*<div>*

*<h3 class="center">*

*Бронирование отелей на HotelBook*

*Наши основные преимущества*

*</h3>*

*</div>*

*<hr>*

*<ul class="main">*

*<li class="main">Все цены + скидки Мы сравниваем цены на отели в 70 агентствах бронирования отелей в одном месте.</li>*

*<li class="main">Без скрытых платежей HotelBook показывает окончательную цену на отель. Никаких дополнительных налогов и скрытых комиссий.</li>*

*<li class="main">Экономно Цены на отели постоянно меняются. Подпишитесь на изменение цены нужного отеля.</li>*

*<li class="helper"></li>*

*</ul>*

*</div>*

*<div id="body\_main">*

*<a class="obj" href="/Home/Moscow" id="obj1">*

*<div class="shadow">*

*<h2 id="w">Париж</h2>*

*</div>*

*</a>*

*<a class="obj" href="/Home/Moscow" id="obj2">*

*<div class="shadow">*

*<h2 id="w">Москва</h2>*

*</div>*

*</a>*

*<a class="obj" href="/Home/Moscow" id="obj3">*

*<div class="shadow">*

*<h2 id="w">Лондон</h2>*

*</div>*

*</a>*

*</div>*

*</div>*

Применим css стили (Листинг 6.1 – 6.2).

Листинг 6.1 – css стили заголовка для главной страницы.

*div#seach\_main{*

*background-image: url("img/HeadBackMain.jpg");*

*height: 200px;*

*width: 800px;*

*background-size: 100% 300px;*

*margin-left: auto;*

*margin-right: auto;*

*margin-bottom: 0%;*

*padding-top: 100px;*

*padding-right: 0%;*

*}*

*div#seach\_second{*

*position: center;*

*background-color: rgba(180, 180, 180, 0.623) ;*

*height: 150px;*

*width: 800px;*

*}*

Листинг 6.2 – css стили для главной страницы.

*div#info{*

*margin-left: 13%;*

*margin-right: 13%;*

*height: wrap content;*

*text-align: center;*

*}*

*div#body\_main{*

*background-color: #f0efef;*

*height: 450px;*

*width: 100%;*

*}*

*.hotel\_page\_body{*

*margin: 0 10%;*

*}*

*.obj{*

*display: block;*

*background-size: 100% 100%;*

*margin: 4.5% 4.5%;*

*height: 300px;*

*width: 23%;*

*float: left;*

*}*

*.obj:hover{*

*background-color: rgba(180, 180, 180, 0.623) ;*

*}*

*a#obj1{*

*background-image: url("img/Paris.jpg");*

*}*

*a#obj2{*

*background-image: url("img/Moscow.jpg");*

*}*

*a#obj3{*

*background-image: url("img/London.jpg");*

*background-size: 100% 100%;*

*}*

*.shadow{*

*position: center;*

*padding: 0% 5%;*

*background-color: rgba(126, 125, 125, 0.623) ;*

*height: 50px;*

*width: wrap content;*

*}*

*div.column{*

*margin: 0%;*

*margin-left: 15%;*

*padding: 0px;*

*margin-left: 100px;*

*width: 100px;*

*float: left;*

*}*

*ul.main{*

*font: 14px Verdana, Geneva, sans-serif;*

*text-align: center;*

*}*

*li.main {*

*width: 18%;*

*height: 100px;*

*margin: 0% 5%;*

*display: inline-block;*

*text-align: center;*

*padding-bottom: 20px;*

*margin-top: 0;*

*}*

*ul li.helper {*

*width: 100%;*

*height: 0;*

*visibility: hidden;*

*}*

Результат показан на рисунке 2.6.

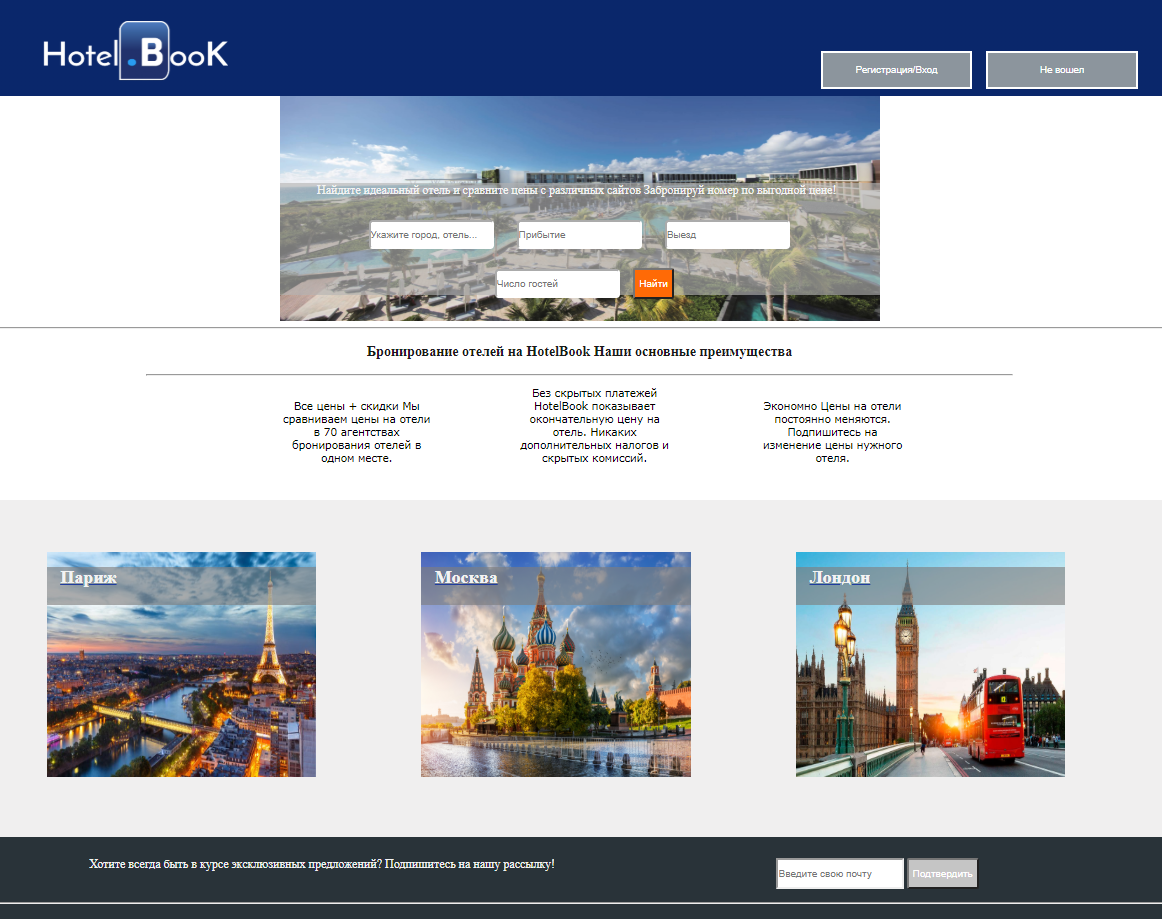


Рисунок 2.6 Главная страница сайта.

Теперь добавим разметку города (Листинг 7).

Листинг 7 – html разметка страницы городов.

*<body>*

*<div calss="body">*

*<div class="body-head">*

*<a href="Index"><h2>Главная </a><a href="Moscow">Москва </a></h2>*

*<hr>*

*<p>*

*3 причины побывать здесь:*

*достопримечательности прогулки по городу архитектура.*

*Москва играет разные роли, главные из которых — державной столицы и культурного центра.*

*В Кремле веками формировалась политика всей страны. За стенами этой величественной*

*цитадели — великолепные соборы и музеи — свидетели истории, которую писали здесь цари,*

*императоры и вожди.*

*</p>*

*<hr>*

*</div>*

*<div class="body-main">*

*<div class="left-main">*

*<div class="seach">*

*<h2 id="inp">Найти</h2>*

*<p class="small">Место, город, отель</p>*

*<input class="default" id="rad" placeholder="Город">*

*<p class="small">Дата заезда</p>*

*<input class="default" id="rad" placeholder="Дата заезда">*

*<p class="small">Дата отъезда</p>*

*<input class="default" id="rad" placeholder="Дата отъезда">*

*<p class="small">Число готей, тип комнаты</p>*

*<input class="default" id="rad" placeholder="Число гостей">*

*<button class="orange" id="org"> Подтвердить</button>*

*</div>*

*</div>*

*<div class="right-main">*

*<div class="hotel">*

*<a href="HotelM1" class="obj" id="obj\_img"></a>*

*<div class="obj" id="obj\_des">*

*<a href="HotelM1"><h2 style="color:#203C9E;">Davidov hostel</h2></a>*

*<p>*

*Этот хостел с бесплатным Wi-Fi и круглосуточной стойкой регистрации расположен в*

*историческом центре Москвы, в 500 метрах от станции метро «Арбатская» и в 1,2 км*

*от московского храма Христа Спасителя....*

*</p>*

*<p style="color:red;">Забронировано 2 раза за последние 6 часов</p>*

*</div>*

*<div class="obj" id="obj\_info">*

*<img src="../../Content/img/Оценки1.jpg" width="100%">*

*<a href="HotelM1"><button class="hotel">Показать цены</button></a>*

*</div>*

*</div>*

*<div class="hotel">*

*<a href="HotelM2" class="obj" id="obj\_img\_2"></a>*

*<div class="obj" id="obj\_des">*

*<a href="HotelM2"><h2 style="color:#203C9E;">Отель эрмитаж</h2></a>*

*<p>*

*Этот отель расположен в центре Москвы, в 20 минутах ходьбы от Красной площади и Кремля. К услугам гостей номера с мини-баром и бесплатным Wi-Fi, а также сауна и фитнес-центр.*

*</p>*

*<p style="color:rgb(23, 87, 42);">Вы можете оформить бронирование без кредитной карты</p>*

*</div>*

*<div class="obj" id="obj\_info">*

*<img src="../../Content/img/Оценки1.jpg" width="100%">*

*<a href="HotelM2"><button class="hotel">Показать цены</button></a>*

*</div>*

*</div>*

*<div class="hotel">*

*<a href="HotelM3" class="obj" id="obj\_img\_3"></a>*

*<div class="obj" id="obj\_des">*

*<a href="HotelM3"><h2 style="color:#203C9E;">Отель Матрешка</h2></a>*

*<p>*

*Этот отель расположен в самом центре города Москвы, в 5 минутах ходьбы от Большого театра и торгового центра ЦУМ и в 10 минутах ходьбы от Кремля. К услугам гостей бесплатный Wi-Fi и снэк-бар.*

*</p>*

*<p style="color:red;">Забронировано 3 раза за последние 13 часов</p>*

*</div>*

*<div class="obj" id="obj\_info">*

*<img src="../../Content/img/Оценки1.jpg" width="100%">*

*<a href="HotelM3"><button class="hotel">Показать цены</button></a>*

*</div>*

*</div>*

*</div>*

*</div>*

*</div>*

Применим css стили (Листинг 8).

Листинг 8 – css стили страницы городов.

*.body{*

*background-color: rgb(175, 175, 175);*

*}*

*.body-head{*

*background-color: #ffffff;*

*margin: 0% 4%;*

*}*

*.body-main{*

*background-color: #ffffff;*

*padding-top: 1%;*

*height: wrap content;*

*}*

*/\* LEFT POSITION \*/*

*/\* ------------------------------------------------------------- \*/*

*.left-main{*

*width: 16%;*

*margin-left: 3%;*

*padding: 0% 0%;*

*float: left;*

*}*

*.seach{*

*background-color: rgb(255, 201, 24);*

*padding: 5%;*

*Border-radius: 5px;*

*height: 350px;*

*}*

*.map{*

*background-image: url("img/map2.jpg");*

*background-size: 100% 100%;*

*height:150px;*

*width:300px;*

*}*

*/\* RIGHT POSITION \*/*

*/\* ------------------------------------------------------------- \*/*

*.right-main{*

*margin-top: 0.3%;*

*padding-bottom: 3%;*

*height: wrap content;*

*width: 85%;*

*}*

*.hotel{*

*margin-left: 24%;*

*padding: 2% 2%;*

*width: 80%;*

*height: 200px;*

*border: #a3a3a3 solid 1px;*

*}*

*.obj{*

*margin: 0 0;*

*height: 190px;*

*}*

*#obj\_img{*

*background-image: url("img/hotel2.jpg");*

*background-size: 100% 100%;*

*}*

*#obj\_img\_2{*

*background-image: url("img/hotel1.jpg");*

*background-size: 100% 100%;*

*}*

*#obj\_img\_3{*

*background-image: url("img/hotel3.jpg");*

*background-size: 100% 100%;*

*}*

*#obj\_des{*

*margin: 0 2%;*

*width: 45%;*

*}*

*#obj\_info{*

*width: 20%;*

*height: 150px;*

*}*

*.obj:hover{*

*background-color: rgb(255, 255, 255) ;*

*}*

Результат показан на рисунке 2.7.

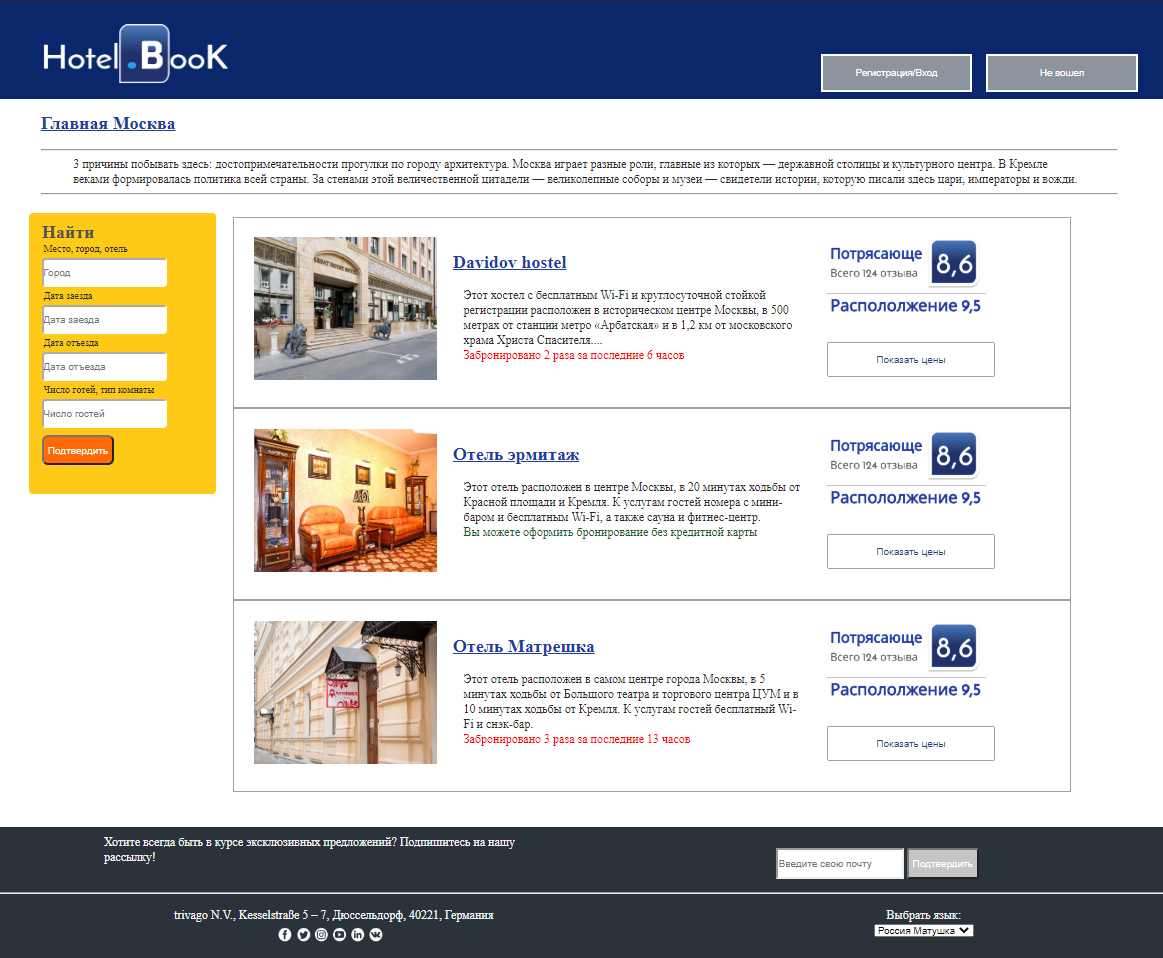


Рисунок 2.7 Страница города Москва.

Теперь добавим страницу отеля на сайт. Данные о номерах, которые предоставляет отель, будут выбираться из базы данных, после чего выводиться в таблицу. разметку отеля (Листинг 9).

Листинг 9 – html разметка отеля.

*<div class="body-main">*

*<div class="left-main">*

*<div class="seach">*

*<h2 id="inp">Найти</h2>*

*<p class="small">Место, город, отель</p>*

*<input class="default" id="rad" placeholder="Город">*

*<p class="small">Дата заезда</p>*

*<input class="default" id="rad" placeholder="Дата заезда">*

*<p class="small">Дата отъезда</p>*

*<input class="default" id="rad" placeholder="Дата отъезда">*

*<p class="small">Число готей, тип комнаты</p>*

*<input class="default" id="rad" placeholder="Число гостей">*

*<button class="orange" id="org"> Подтвердить</button>*

*</div>*

*</div>*

*<div class="right-main" id="h">*

*<div class="hotel\_page">*

*<div class="hotel\_page\_head">*

*<div class="left" style="width:50%; float:left; margin-left:5%;">*

*<h2 style="color: #203C9E">Арбат хаус</h2>*

*<p>*

*Этот отель находится в самом центре Москвы, в 5 минутах ходьбы от станции метро «Арбатская». К услугам гостей номера с бесплатным Wi-Fi и спутниковым телевидением.*

*Забронировано 20 раз за последние 6 часов*

*</p>*

*<p style="color: red">Последний раз забронировано 52 минуты назад</p>*

*</div>*

*<div class="right"><img src="../../../Content/img/Оценки1.jpg" style="width: 200px;margin-top:18px ;margin-left:15%; margin-bottom:100px ;"> </div>*

*<hr>*

*</div>*

*<div class="hotel\_page\_body" style="background-color: #d1d1d1ad; height: 250px; border: solid 1px #b1b1b1f8;">*

*<div class="image image"></div>*

*</div>*

*<hr>*

*<div class="hotel\_page\_info">*

*<pi>*

*Москва — остановитесь в самом центре – Великолепное расположение — показать на карте*

*Один из лучших вариантов, выбранных нами в городе Москва.*

*Этот отель находится в самом центре Москвы, в 5 минутах ходьбы от станции метро «Арбатская». К услугам гостей номера с бесплатным Wi-Fi и спутниковым телевидением.*

*Самые популярные удобства и услуги*

*Бесплатный Wi-Fi Семейные номера Трансфер от/до аэропорта Номера для некурящих Ресторан Круглосуточная стойка регистрации Бар*

*Банкомат: Нужны наличные? Здесь есть банкомат.*

*</pi>*

*</div>*

*</div>*

*<div class="info">*

*<div class="info\_head">*

*</div>*

*</div>*

*</div>*

*<hr>*

*<div class="table">*

*<table>*

*<tr>*

*<th>Вмещает</th>*

*<th>Тип номера</th>*

*<th>Цена</th>*

*<th>Забронировать</th>*

*</tr>*

*@foreach (Rooms r in Model)*

*{*

*if (r.HotelId == 1)*

*{*

*<tr>*

*@if (@r.RoomType == "Двухместный ")*

*{*

*<td><img src="../../../Content/img/people2.jpg" width="50" height="25"></td>*

*}*

*else*

*{*

*<td><img src="../../../Content/img/people1.jpg" width="20" height="25"></td>*

*}*

*<td>Описание: @r.RoomInfo</td>*

*<td>Цена: @r.Price</td>*

*@if (User.Identity.IsAuthenticated)*

*{*

*<td><button class="tab" onclick="AddReservation('@r.RoomId', '@ViewBag.UserName')"><pi>Выбрать</pi></button></td>*

*}*

*else*

*{*

*<td><button class="tab" onclick="Info()"><pi>Выбрать</pi></button></td>*

*}*

*</tr>*

*}*

*}*

*</table>*

*</div>*

*<hr>*

*<div id="body\_main" style="background-color: #FFF; padding-bottom: 330px;">*

*<div class="obj" id="head\_">*

*<h2 style="text-align: center; margin-left: -60%">Самые популярные удобства и услуги</h2>*

*<ul>*

*<li id="id0"><p>Бесплатный Wi-Fi</p></li>*

*<li id="id1"><p>Семейные номера</p></li>*

*<li id="id2"><p>Трансфер от/до аэропорта </p></li>*

*<li id="id3"><p>Номера для некурящих </p></li>*

*<li id="id4"><p>Ресторан</p></li>*

*<li id="id5"><p>Круглосуточная стойка регистрации</p></li>*

*<li id="id6"><p>Бар</p></li>*

*</ul>*

*</div>*

*<div class="obj" id="obj1">*

*<h2 style="text-align: center; margin-left: -35%">Самые популярные удобства и услуги</h2>*

*<ul>*

*<li id="img"><p>Трансфер</p></li>*

*<li id="img"><p>Общий лаундж/гостиная с телевизором</p></li>*

*<li id="img"><p>Трансфер от/до аэропорта (оплачивается отдельно)</p></li>*

*<li id="img"><p>Трансфер (оплачивается отдельно)</p></li>*

*<li id="img"><p>Кондиционер</p></li>*

*<li id="img"><p>Курение на всей территории запрещено</p></li>*

*<li id="img"><p>Лифт</p></li>*

*<li id="img"><p>Семейные номера</p></li>*

*<li id="img"><p>Номера для некурящих</p></li>*

*<li id="img"><p>Доставка еды и напитков в номер</p></li>*

*</ul>*

*</div>*

*<div class="obj" id="obj2">*

*<h2 style="text-align: center; margin-left: -35%">Услуги уборки</h2>*

*<ul>*

*<li id="img"><p>Чистка обуви</p></li>*

*<li id="img"><p>Услуги по глажению одежды (Оплачивается отдельно)</p></li>*

*<li id="img"><p>Прачечная (Оплачивается отдельно)</p></li>*

*</ul>*

*<h2 style="text-align: center; margin-left: -35%">Питание</h2>*

*<ul>*

*<li id="img"><p>Бар</p></li>*

*<li id="img"><p>Ресторан</p></li>*

*<li id="img"><p>Хороший кофе!</p></li>*

*</ul>*

*<h2 style="text-align: center; margin-left: -35%">Интеренет</h2>*

*<ul>*

*<li id="img"><p>Бесплатно! Wi-Fi предоставляется на территории всего отеля бесплатно.</p></li>*

*</ul>*

*</div>*

*</div>*

*</div>*

*</div>*

Применим css стили (Листинг 10).

Листинг 10 – css стили для страницы отеля.

*.hotel\_page {*

*margin-left: 24%;*

*margin-bottom: 24%;*

*padding: 2% 2%;*

*width: 80%;*

*height: 800px;*

*}*

*.hotel\_page\_head {*

*padding: 0;*

*}*

*.hotel\_page\_body {*

*width: 80%;*

*height: 200px;*

*}*

*.hotel\_page\_info {*

*margin-bottom: 10%;*

*}*

*.image {*

*width: 400px;*

*height: 250px;*

*-webkit-background-size: cover;*

*background-size: 100% 100%;*

*margin: 0 20%;*

*background-image: url("img/Hotel2.jpg");*

*animation-delay: 6s;*

*}*

*.image\_01 {*

*background-image: url("img/Hotel2.jpg");*

*animation-delay: 6s;*

*}*

*/\*TABLE!!!!!!!!!!!!!!!!! \*/*

*/\* -------------------------------------------------------------------------------- \*/*

*.table {*

*margin: 5% 10%;*

*width: 80%;*

*height: 400px;*

*}*

*/\* Table \*/*

*/\* -------------------------------------------------------------------- \*/*

*table, td, th {*

*border: 1px solid #2B468F;*

*text-align: center;*

*}*

*table {*

*border-collapse: collapse;*

*width: 100%;*

*}*

*th, td {*

*padding: 15px;*

*}*

*th {*

*background-color: #2B468F;*

*color: #FFFFFF;*

*}*

Результат показан на рисунках 2.8-2.9.

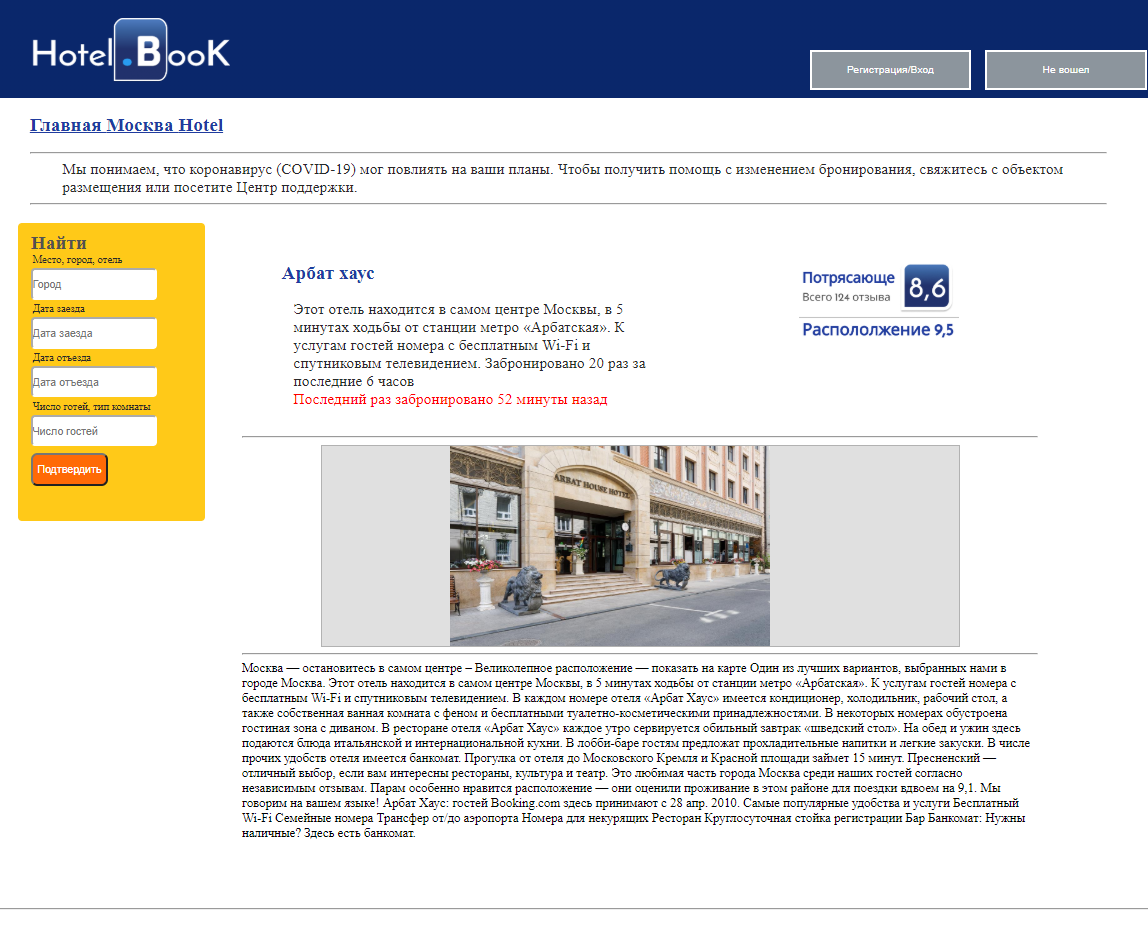


Рисунок 2.8 Страница отеля Арбат Хаус.



Рисунок 2.9 Страница отеля «Арбат Хаус».

Страница «Профиль пользователя» будет доступна только для авторизированных пользователей. В ней будут выводиться все бронирования пользователя, а также будет присутствовать выйти из профиля. Добавим html разметку данной страницы (Листинг 11).

Листинг 11 – html разметка страницы «Профиль пользователя».

*<body>*

*<div class="userForm">*

*<p id="w">ВАШИ ЗАКАЗЫ:</p>*

*@foreach (Reservation r in Model)*

*{*

*if (r.UserName == @ViewBag.UserName)*

*{*

*<div id="w">*

*Дата бронирования: @r.sDate*

*</div>*

*<div id="w">*

*город: @r.City, @r.Hotel, Тип: @r.RoomType, Цена: @r.Price*

*<button class="default" onclick="DelReservation(@r.IdReservation)">Удалить</button>*

*</div>*

*}*

*}*

*<a id="center" href="/Home/Index" vlink="white" alink="white" class="btn">На главную</a>*

*<a id="center" href="/Home/Index" class="btn" onclick="LogOut()">Выйти из системы</a>*

*</div>*

*</body>*

Данные о резервированиях пользователя будут выбираться из базы данных и выводиться на страницу. Добавим css стили для этой формы (Листинг 11).

Листинг 11 – css стили страницы «Профиль пользователя».

*.userForm {*

*position: fixed;*

*top: 20%;*

*left: 35%;*

*padding: 20px;*

*width: 30%;*

*background-color: rgba(10,39,107,1);*

*border-radius: 20px;*

*}*

*#center {*

*margin-top: 20px;*

*margin-left: 20%;*

*text-align: center;*

*}*

Результат показан на Рисунке 2.10.

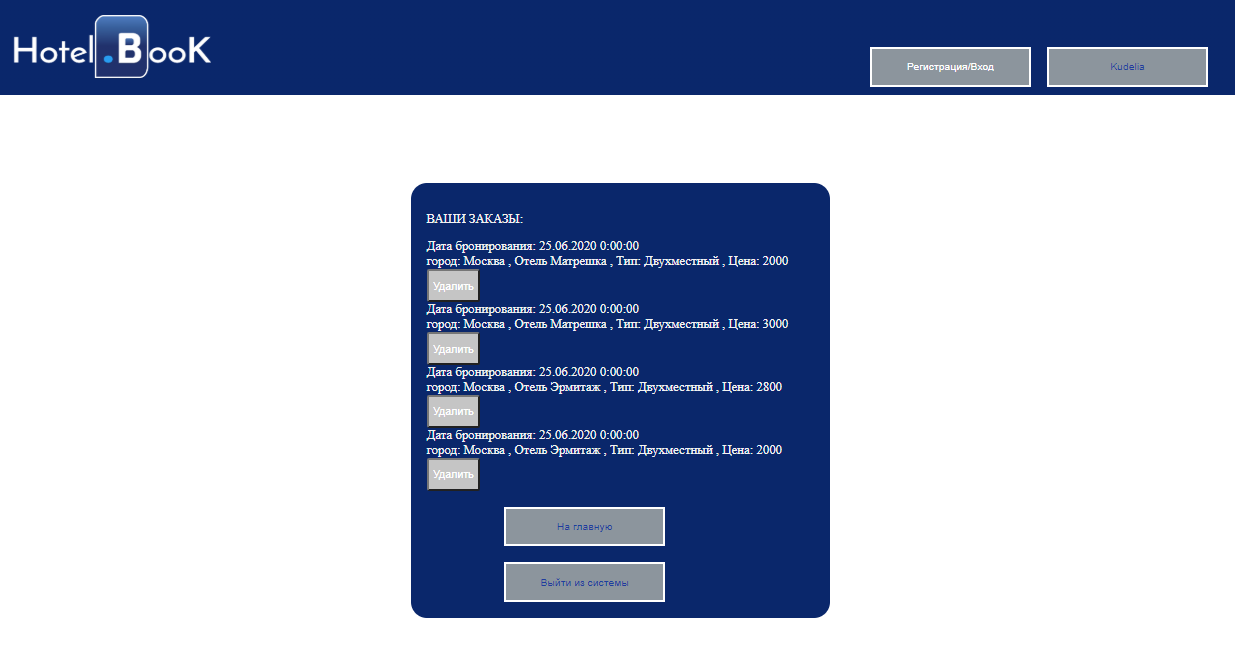


Рисунок 2.10 Страница «Профиль пользователя».

* 1. **Разработайте серверную часть интернет магазина.**

Теперь напишем серверную часть сайта. Для этого создадим новый контроллер и определим в нем методы Index(), Moscow(), Hotel1(), Hotel2(), Hotel3(), UserProfile(), которые будут возвращать наши представления.

Листинг 12 – код методов

*public ActionResult Index()*

*{*

*ViewBag.UserName = System.Web.HttpContext.Current.User.Identity.Name;*

*return View();*

*}*

*public ActionResult Moscow()*

*{*

*ViewBag.UserName = System.Web.HttpContext.Current.User.Identity.Name;*

*return View();*

*}*

*public ActionResult UserPriofile()*

*{*

*ViewBag.UserName = System.Web.HttpContext.Current.User.Identity.Name;*

*return View(ct.GetReservations());*

*}*

*public ActionResult HotelM1()*

*{*

*ViewBag.UserName = System.Web.HttpContext.Current.User.Identity.Name;*

*return View(ct.GetRooms());*

*}*

*public ActionResult HotelM2()*

*{*

*ViewBag.UserName = System.Web.HttpContext.Current.User.Identity.Name;*

*return View(ct.GetRooms());*

*}*

*public ActionResult HotelM3()*

*{*

*ViewBag.UserName = System.Web.HttpContext.Current.User.Identity.Name;*

*return View(ct.GetRooms());*

*}*

Для выборки данных из базы данных создадим новые классы (Листинг 13)

Листинг 13 – классы, используемые для передачи данных в представление.

*public class Reservation*

*{*

*public int IdReservation { get; set; }*

*public string RoomType { get; set; }*

*public string Hotel { get; set; }*

*public string City { get; set; }*

*public Int32 Price { get; set; }*

*public string UserName { get; set; }*

*public DateTime sDate { get; set; }*

*}*

*public class Rooms*

*{*

*public int RoomId { get; set; }*

*public int HotelId { get; set; }*

*public string RoomType { get; set; }*

*public string RoomInfo { get; set; }*

*public Int32 Price { get; set; }*

*public string Hotel { get; set; }*

*public string City { get; set; }*

*}*

В параметры View() методов отелей и UserProfile передадим список объектов, выбранных из базы данных, которые нам нужно отобразить в данных представлениях. Реализация методов, совершающих выборку данных из базы данных, и возвращающих их в виде списка объектов предоставлена в листинге 14.

Листинг 14 – методы, возвращающие данные из базы данных в виде списка объектов.

*public List<Rooms> GetRooms()*

*{*

*List<Rooms> r = new List<Rooms>();*

*using (SqlConnection conn = new SqlConnection(ConfigurationManager.ConnectionStrings["Database"].ConnectionString))*

*{*

*conn.Open();*

*SqlCommand command = new SqlCommand("SELECT Rooms.IdRoom, Rooms.IdHotel, Rooms.Type, Rooms.Info, Rooms.Price, Hotels.Name, Hotels.City" +*

*" FROM Rooms, Hotels" +*

*" WHERE Rooms.IdHotel = Hotels.IdHotel", conn);*

*using (var reader = command.ExecuteReader())*

*{*

*while (reader.Read())*

*{*

*r.Add(new Rooms() { RoomId = reader.GetInt32(0), HotelId = reader.GetInt32(1), RoomType = reader.GetString(2), RoomInfo = reader.GetString(3),*

*Price = reader.GetInt32(4), Hotel = reader.GetString(5), City = reader.GetString(6)});*

*}*

*}*

*}*

*return r;*

*}*

*public List<Reservation> GetReservations()*

*{*

*List<Reservation> r = new List<Reservation>();*

*using (SqlConnection conn = new SqlConnection(ConfigurationManager.ConnectionStrings["Database"].ConnectionString))*

*{*

*conn.Open();*

*SqlCommand command = new SqlCommand("SELECT Reservations.IdReservation, Rooms.Type, Hotels.Name, Hotels.City, Rooms.Price, Reservations.UserName, Reservations.sDate" +*

*" FROM Reservations, Rooms, Hotels" +*

*" WHERE Reservations.IdRoom = Rooms.IdRoom AND Rooms.IdHotel = Hotels.IdHotel", conn);*

*using (var reader = command.ExecuteReader())*

*{*

*while (reader.Read())*

*{*

*r.Add(new Reservation() { IdReservation = reader.GetInt32(0), RoomType = reader.GetString(1), Hotel = reader.GetString(2), City = reader.GetString(3), Price = reader.GetInt32(4), UserName = reader.GetString(5), sDate = reader.GetDateTime(6)});*

*}*

*}*

*}*

*return r;}*

Для авторизации и регистрации пользователей добавим методы AddUser(), LogUser(). Оба в качестве параметра получают логин и пароль. Первый метод проводит проверку на наличие такого пользователя в базе данных: если такого пользователя нет, то добавляет пользователя (Листинг 15).

Листинг 15 метод AddUser().

*[HttpPost]*

*public ActionResult AddUser(MyUser user)*

*{*

*if (!ct.CheckUser(user))*

*{*

*ct.AddUser(user);*

*}*

*return RedirectToAction("Index");*

*}*

Листинг 16 методы реализующие проверку и добавление пользователя в базу данных.

*public bool CheckUser(MyUser user)*

*{*

*using (SqlConnection conn = new SqlConnection(ConfigurationManager.ConnectionStrings["Database"].ConnectionString))*

*{*

*conn.Open();*

*SqlCommand command = new SqlCommand("select \* from Users", conn);*

*using (var reader = command.ExecuteReader())*

*{*

*while (reader.Read())*

*{*

*string NameD = reader.GetString(0);*

*string PassD = reader.GetString(1);*

*if (user.Name == NameD && user.Pass == PassD)*

*return true;*

*}*

*}*

*}*

*return false;*

*}*

*public void AddUser(MyUser user)*

*{*

*using (SqlConnection conn = new SqlConnection(ConfigurationManager.ConnectionStrings["Database"].ConnectionString))*

*{*

*conn.Open();*

*String query = "INSERT INTO Users (Name, Pass) VALUES (@username, @password)";*

*using (SqlCommand command = new SqlCommand(query, conn))*

*{*

*command.Parameters.AddWithValue("@username", user.Name);*

*command.Parameters.AddWithValue("@password", user.Pass);*

*command.ExecuteNonQuery();*

*}*

*}*

*}*

Второй метод, после проверки пользователя, записывает имя пользователя в куки (Листинг 17).

Листинг 17 метод LogUser().

*[HttpPost]*

*public ActionResult LogUser(MyUser user)*

*{*

*if (ct.CheckUser(user))*

*FormsAuthentication.SetAuthCookie(user.Name, false);*

*return RedirectToAction("Index");*

*}*

Для выхода пользователя из систем добавим метод LogOut() (Листинг 18).

Листинг 18 метод LogOut().

*[HttpPost]*

*public ActionResult LogOut()*

*{*

*FormsAuthentication.SignOut();*

*return RedirectToAction("Index");*

*}*

Добавление новых бронирований осуществляется через метод AddReservation() (Листинг 18), где в параметре передается объект, содержащий номер бронируемой комнаты, и пользователя, который совершил эту операцию.

Листинг 18 - метод AddReservation ().

*[HttpPost]*

*Public void AddReservation(Room id)*

*{*

*ct.AddReservation(id);*

*}*

Внутри вызывается метод контекста, который и совершает добавление новых данных в базу данных (Листинг 19).

Листинг 19 – метод добавления данных в базу данных.

*using (SqlConnection conn = new SqlConnection(ConfigurationManager.ConnectionStrings["Database"].ConnectionString))*

*{*

*conn.Open();*

*String query = "INSERT INTO Reservations(UserName, IdRoom, sDate, fDate) values(@username, @id, gestate(), getdate())";*

*using (SqlCommand command = new SqlCommand(query, conn))*

*{*

*command.Parameters.AddWithValue("@username", id.UserName);*

*command.Parameters.AddWithValue("@id", id.RoomId);*

*command.ExecuteNonQuery();*

*}*

*}*

Удаление бронирований осуществляется через метод DelReservation() (Листинг 20), где в параметре передается id бронирования.

Листинг 20 - метод DelReservation ().

*[HttpPost]*

*public void DelReservation(Reserv id)*

*{*

*ct.DelReservation(id);*

*}*

Метод удаления данных из базы данных предоставлен ниже (Листинг 21).

Листинг 21 – метод удаления данных из базы данных.

*using (SqlConnection conn = new SqlConnection(ConfigurationManager.ConnectionStrings["Database"].ConnectionString))*

*{*

*conn.Open();*

*SqlCommand command = new SqlCommand("DELETE FROM Reservations WHERE IdReservation = @id", conn);*

*command.Parameters.AddWithValue("@id", id.ReservId);*

*command.ExecuteNonQuery();*

*}*

**Заключение**

При выполнении теоретической части были рассмотрены популярные фреймворки и библиотеки java-script. Приведены их основные преимущества и недостатки.

Практическая часть выполнена при помощи Visual Studio, в частности с использованием платформы ASP.NET. Помимо этого, были закреплены знания в создании приложений с использованием языков программирования C#, java-script, а также html и css.

В результате выполнения курсовой работы было разработано веб – приложение «Сервис бронирования отелей». Сайт состоит из главной страницы, страницы города, страниц отелей и профиля пользователя. Добавлены методы, позволяющая бронировать номера, отменять бронирование.

**Список используемой литературы**

1. Фреймворки в веб разработке – Веб Креатор: сайт. – <https://web-creator.ru/articles/about_frameworks> (дата обращения: 05.06.2020). – Текст: электронный.
2. Что такое фреймворк и какие у него преимущества: сайт. – <https://depix.ru/articles/chto_takoe_freymvork_i_kakie_u_nego_preimuschestva_pered_korobochnymi_cms> (дата обращения: 06.06.2020). – Текст: электронный.
3. Основные библиотеки и фреймворки JavaScript: сайт. – <https://code.tutsplus.com/ru/articles/essential-javascript-libraries-and-frameworks-you-should-know-about--cms-29540> (дата обращения: 04.06.2020). – Текст: электронный.
4. Tproger: сайт. – <https://tproger.ru/digest/top-javascript-frontend-frameworks/> (дата обращения: 04.06.2020). – Текст: электронный.
5. Изучение веб-разработки – MDN – Mozilla: сайт. – <https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/JavaScript_basics#:~:text=JavaScript%20(%22JS%22%20%D0%B4%D0%BB%D1%8F%20%D0%BA%D1%80%D0%B0%D1%82%D0%BA%D0%BE%D1%81%D1%82%D0%B8,Mozilla%20Foundation%20%D0%B8%20Mozilla%20Corporation.> (дата обращения: 06.06.2020). – Текст: электронный.
6. Аутентификация форм в ASP.NET MVC 5 – METANIT.COM: сайт. – <https://metanit.com/sharp/mvc5/12.5.php> (дата обращения: 06.06.2020). – Текст: электронный.
7. Envato tuts+: сайт. – <https://code.tutsplus.com/ru/articles/essential-javascript-libraries-and-frameworks-you-should-know-about--cms-29540> (дата обращения: 05.06.2020). – Текст: электронный.