Федеральное государственное автономное образовательное учреждение высшего образования

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**

**МЕХАНИКИ И ОПТИКИ**

**Академия методов и техники управления («ЛИМТУ»)**

Кафедра компьютерного проектирования и дизайна

Выпускная аттестационная работа по специализации

«Проектирование и дизайн веб-сайтов»

**На тему: Разработка сайта-визитки фирмы**

Руководитель: /Перепелица Ф.А./ /*\_\_\_\_\_\_\_\_*/

Исполнитель: /Коптяева А.Д./ /\_\_\_\_\_\_\_\_/

Санкт-Петербург

**2017**

**Содержание**

[Введение 3](#_Toc489959707)

[Ход выполнения работы 4](#_Toc489959708)

[1. Дизайн макета сайта 4](#_Toc489959709)

[2. Технологический раздел 9](#_Toc489959710)

[2.1 Средства разработки 9](#_Toc489959711)

[2.2 Поэтапная разработка сайта 16](#_Toc489959712)

[2.2.1 «Шапка» сайта 16](#_Toc489959713)

[2.2.2 Раздел «Ресторан» 17](#_Toc489959714)

[2.2.3 Раздел «Спецпредложения» 17](#_Toc489959715)

[2.2.4 Раздел «Меню» 20](#_Toc489959716)

[2.2.5 Раздел «Отзывы» 22](#_Toc489959717)

[2.2.6 Раздел «Контакты» 24](#_Toc489959718)

[2.2.7 Раздел «Координаты» 25](#_Toc489959719)

[2.2.8 Раздел «Подвал» 29](#_Toc489959720)

[3. Оптимизация и продвижение готового сайта 30](#_Toc489959721)

[3.1 Оптимизация контента 30](#_Toc489959722)

[3.2 Продвижение сайта 32](#_Toc489959723)

[Заключение 36](#_Toc489959724)

## **Введение**

Цель работы*:* разработка и создание сайта-визитки фирмы.

Задачи работы*:*

* создать сайт выбранной фирмы по макету;
* использовать изученные технологии;
* страница должна включать навигационные элементы, охватывающие все разделы сайта;
* сайт должен быть в одном дизайнерском стиле;
* сайт должен быть простым, ясным и максимально логически выстроен;
* сайт должен иметь интерактивные элементы.

В качестве фирмы выбран ресторан стейков. Такая тема выбрана потому, что создавать сайт ресторана - это интересно и «вкусно».

В конце выполненной работы должен быть готовый сайт, где будут использованы изученные технологии, на странице будут присутствовать динамические эффекты и интерактивные элементы.

В данной работе задействованы следующие программы: Sublime Text 3, CodeLobster, Adobe Photoshop.

В работе использовались следующие языки: HTML, CSS, LESS, Bootstrap, JavaScript, jQuery, PHP.

## **Ход выполнения работы**

## **Дизайн макета сайта**

Основываясь на поставленных задачах, разработаем макет будущего веб-сайта. Решено, что сайт будет представлять собой лендинг ресторана Steak House. Следовательно, на одной странице необходимо использовать все изученные технологии.

Основная структура сайта – «шапка» сайта, навигация, основной контент и «подвал» сайта. В «шапке» сайта должен быть логотип, основной заголовок, так же возможно размещение навигации. Навигация может располагаться сбоку от основного контента, входить в «шапку» или идти после нее. Она должна быть удобной и понятной. Основной контент должен рассказывать про главные аспекты компании, быть легким и запоминающимся. В конце сайта идет «подвал». Там возможно расположение контактов, адреса, так же год создания сайта, его автора и др.

Основа дизайна макета была взята из интернета и изменена под текущее задание. Макет создавался в Adobe Photoshop. Ниже представлены основные разделы макета страницы. Цельный макет прилагается в архиве в формате PSD.



Рисунок 1 – «Шапка» сайта



Рисунок 2 – Раздел «Ресторан»

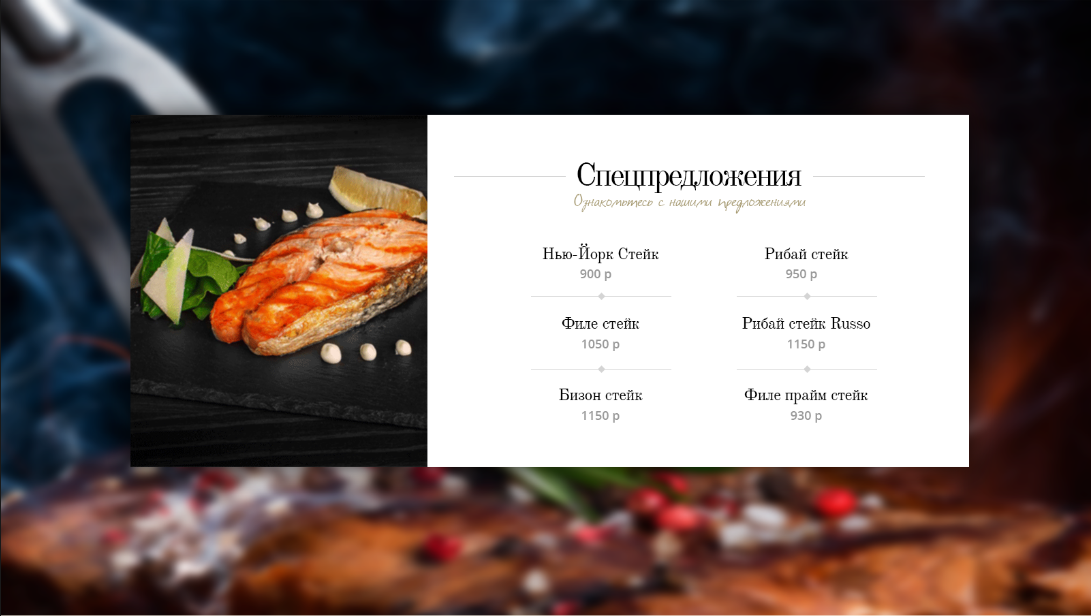


Рисунок 3 – Раздел «Спец предложения»



Рисунок 4 – Раздел «Меню»

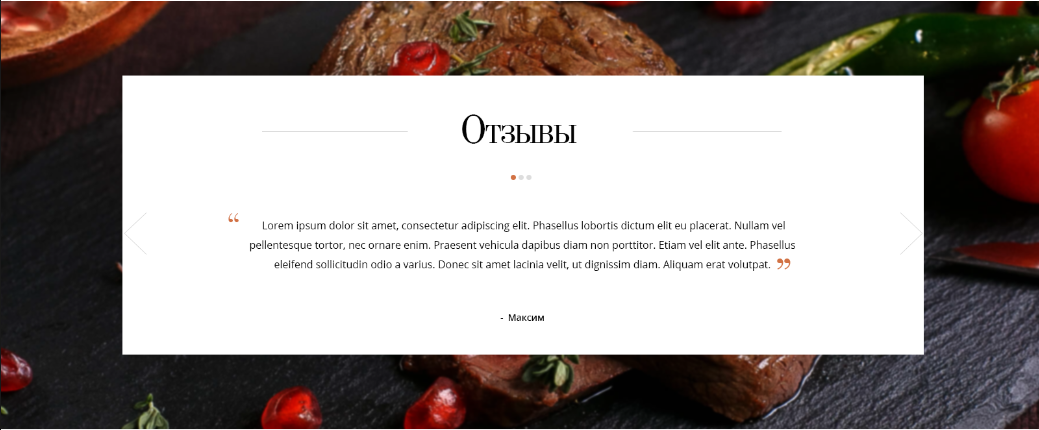


Рисунок 5 – Раздел «Отзывы»

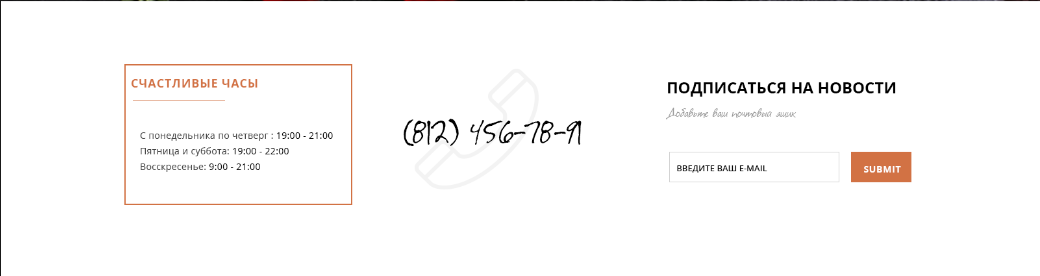


Рисунок 6 – Раздел «Контакты»

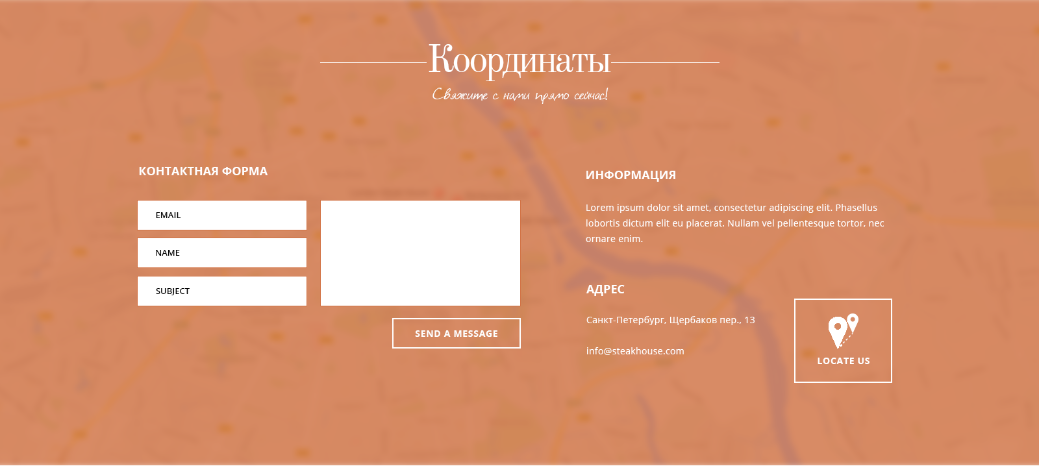


Рисунок 7 – Раздел «Координаты»



Рисунок 8 – «Подвал» сайта

## **Технологический раздел**

### **2.1 Средства разработки**

В данной работе использовались все изученные технологии: HTML, CSS, LESS, Bootstrap, JavaScript, jQuery, PHP.

*HTML* (Hyper Text Markup Language — язык гипертекстовой разметки) является компьютерным языком программирования, предназначенным для разработки веб-страниц и является основополагающим для разработки веб-сайтов. HTML предназначен для структурирования и оформления информации на странице. Для прочтения страниц HTML используются программы браузеры. Для того, чтобы страница отображалась, так как задумывалось надо придерживаться правилам написания HTML кода, которые описаны в спецификации языка HTML. Спецификацию языка HTML разрабатывает World Wide Web Consortium (W3C).

*CSS* (Cascading Style Sheets) ‐ это язык, позволяющий оформить внешний вид веб‐страницы и отчасти управляющий структурированием HTML документов. CSS обладает большими возможностями по оформлению элементов веб‐страниц, чем HTML. СSS используется для оформления внешнего вида веб‐страниц, написанных с помощью языков разметки HTML и XHTML, а также применятся к любым XML‐документам. Основной целью разработки CSS являлось разделение описания логической структуры веб‐страницы (которое производится с помощью HTML или других языков разметки) от описания внешнего вида этой веб‐страницы (которое производится с помощью языка CSS). То есть сам по себе существовать CSS не может, он используется только в связке, с каким‐либо языком разметки страницы, например, HTML. HTML используется для структурирования содержимого страницы. CSS используется для форматирования, этого структурированного содержимого.

Основные преимущества CSS:

* CSS позволяет задавать такие параметры, которые нельзя задать языком HTML. Например, убрать подчеркивание у ссылок, установить межстрочное расстояние и т.п.;
* CSS позволяет значительно сократить размер кода и сделать его читабельным;
* CSS позволяет легко изменять внешний вид страниц. Представьте, вы сделали сайт из 30 страниц, на которых все заголовки 36 размера. Для изменения размера заголовка, указанного в CSS, вам будет достаточно поменять только одну цифру;
* более точный контроль над внешним видом страниц;
* различные представления для разных носителей информации (экран, печать, и т. д.);
* сложная и проработанная техника дизайна.

*LESS* — это динамический язык стилей, который разработал Алексис Селлиер (Alexis Sellier). Он создан под влиянием языка стилей Sass, и, в свою очередь, оказал влияние на его новый синтаксис «SCSS», в котором также использован синтаксис, являющийся расширением CSS. LESS — это продукт с открытым исходным кодом. Его первая версия была написана на языке Ruby, однако в последующих версиях было решено отказаться от использования этого языка программирования в пользу JavaScript. LESS может работать на стороне клиента или на стороне сервера под управлением Node.js или Rhino. LESS можно использовать как на стороне клиента, так и на стороне сервера. Использование на стороне клиента это самый простой способ использования, он хорош для изучения и использования в небольших проектах. Но при использовании LESS в больших проектах рекомендуется использовать компиляцию на стороне сервера. LESS обеспечивает следующие расширения CSS: [переменные](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%BD%D0%B0%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), вложенные блоки, [миксины](https://ru.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BA%D1%81%D0%B8%D0%BD" \o "Миксин), [операторы](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) и [функции](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)).

*JavaScript* — объектно‐ориентированный сценарный язык программирования. Является реализацией языка ECMAScript. JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб‐страницам. Для работы с JavaScript во все современные браузеры встроен интерпретатор. Интерпретатор перед выполнением преобразует JavaScript в машинный код, оптимизирует, а затем выполняет. JavaScript можно использовать, так же и на сервере при условии, что на нем установлен интерпретатор, например, Node.js.

*jQuery* — [библиотека JavaScript](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_JavaScript), фокусирующаяся на взаимодействии [JavaScript](https://ru.wikipedia.org/wiki/JavaScript" \o "JavaScript) и [HTML](https://ru.wikipedia.org/wiki/HTML). Библиотека jQuery помогает легко получать доступ к любому элементу [DOM](https://ru.wikipedia.org/wiki/Document_Object_Model), обращаться к атрибутам и содержимому элементов [DOM](https://ru.wikipedia.org/wiki/Document_Object_Model), манипулировать ими. Также библиотека jQuery предоставляет удобный [API](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9) для работы с [AJAX](https://ru.wikipedia.org/wiki/AJAX).

Преимущества jQuery и почему это библиотека так распространена:

* самый простой синтаксис и самый компактный код;
* простое и понятное обращение и манипулирование элементами веб-страницы;
* множество готовых плагинов для решения разных задач;
* доступная документация;
* библиотека постоянно развивается, добавляя новые возможности и упрощая работу с кодом.

*Bootstrap*(также известен как Twitter Bootstrap) — свободный набор инструментов для создания [сайтов](https://ru.wikipedia.org/wiki/%D0%A1%D0%B0%D0%B9%D1%82) и [веб-приложений](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5). Включает в себя [HTML](https://ru.wikipedia.org/wiki/HTML)- и [CSS](https://ru.wikipedia.org/wiki/CSS)-шаблоны оформления для [типографики](https://ru.wikipedia.org/wiki/%D0%A2%D0%B8%D0%BF%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0), веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая [JavaScript](https://ru.wikipedia.org/wiki/JavaScript)-расширения. Bootstrap на данный момент является самым распространённым и уважаемым фреймворком для проектирования веб приложений. Он пользуется популярностью не только у разработчиков сайтов, но и у заказчиков, которые, убедившись в его надёжности, отдают предпочтение данному фреймворку при создании новых проектов. Скорость веб‐разработки значительно увеличивается при использовании Bootstrap, так как большинство элементов, которые требуются при создании веб приложений, уже готовы и адаптированы для разных браузеров и типов устройств и остаётся лишь задать нужный класс.

*PHP* (Hypertext Preprocessor) — [скриптовый язык](https://ru.wikipedia.org/wiki/%D0%A1%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством [хостинг-провайдеров](https://ru.wikipedia.org/wiki/%D0%A5%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3) и является одним из лидеров среди языков, применяющихся для создания [динамических веб-сайтов](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%BD%D0%B0%D0%BC%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D1%81%D0%B0%D0%B9%D1%82). В области веб-программирования, в частности серверной части, PHP — один из популярных [сценарных языков](https://ru.wikipedia.org/wiki/%D0%A1%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) (наряду с [JSP](https://ru.wikipedia.org/wiki/JSP), [Perl](https://ru.wikipedia.org/wiki/Perl" \o "Perl) и языками, используемыми в [ASP.NET](https://ru.wikipedia.org/wiki/ASP.NET)). Популярность в области построения [веб-сайтов](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B0%D0%B9%D1%82) определяется наличием большого набора встроенных средств для разработки веб-приложений. Основные из них:

* автоматическое извлечение [POST](https://ru.wikipedia.org/wiki/HTTP#POST) и [GET](https://ru.wikipedia.org/wiki/HTTP#GET)-параметров, а также переменных окружения веб-сервера в предопределённые массивы;
* взаимодействие с большим количеством различных СУБД ([MySQL](https://ru.wikipedia.org/wiki/MySQL" \o "MySQL), MSSQL, [SQLite](https://ru.wikipedia.org/wiki/SQLite), [PostgreSQL](https://ru.wikipedia.org/wiki/PostgreSQL" \o "PostgreSQL), [Oracle](https://ru.wikipedia.org/wiki/Oracle_(%D0%A1%D0%A3%D0%91%D0%94)) и др.);
* автоматизированная отправка [HTTP-заголовков](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B3%D0%BE%D0%BB%D0%BE%D0%B2%D0%BA%D0%B8_HTTP);
* работа с HTTP-авторизацией;
* работа с [cookies](https://ru.wikipedia.org/wiki/Cookies" \o "Cookies) и сессиями;
* работа с локальными и удалёнными файлами, [сокетами](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D0%BA%D0%B5%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D1%8B%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81));
* обработка файлов, загружаемых на сервер.

В качестве основного языка использовался HTML 5. На данный момент это последняя версия и она имеет ряд преимуществ:

* поддержка геолокации;
* воспроизведение видеороликов;
* воспроизведение аудиофайлов;
* локальное хранилище — позволяет сайтам сохранять информацию на локальном компьютере и обращаться к ней позже;
* фоновые вычисления — стандартный способ запуска JavaScript в браузере в фоновом режиме;
* оффлайновые приложения — страницы, которые могут работать при отключении Интернета;
* рисование с помощью тега <canvas>;
* новые элементы форм: для даты, времени, поиска, чисел, выбора цвета и др.

Так же упростилась структура основного кода.

Стили страниц написаны на языке LESS. Он довольно прост в написании и похож на CSS. Данный файл можно подключить к HTML-документу вместе со скриптом, который скомпилирует написанные стили в CSS при открытии страницы в браузере. Однако, такой метод задерживает открывание страниц, поэтому файл LESS был скомпилирован в CSS при помощи программы Prepros. Программа проста в использовании, интуитивно понятна и компилирует в сжатый файл CSS, исключая все пробелы и переносы. Данный файл подключаем к HTML-документу в тег <head>.

Для выполнения интерактивных элементов на сайте созданы два файла. Один написан с помощью языка JavaScript. Данный файл подключаем в конец сайта, так как для правильной работы скрипта необходимо что бы страница сначала загружалась полностью, а потом уже начинал работать скрипт. Ссылку на файл вставляем в тег <script>, его, в свою очередь, подключаем перед закрывающим тегом <body>.

Второй файл написан с помощью библиотеки jQuery. Для работы данной библиотеки с официального сайта jQuery скачали нужную нам библиотеку. Я выбрала версию 1.10.1. Полученный файл подключаем к HTML-странице в верху в теге <head>. Сам же файл с функциями jQuery вставляем в конце документа перед закрывающим тегом <body>. Это так же необходимо для правильной работы скрипта.

На HTML-странице использовался фреймворк Bootstrap. Фреймворк скачан с официального сайта. В тег <head> подключаем полученные файлы: bootstrap.min.css, bootstrap-theme.min.css и bootstrap/js/bootstrap.min.js.

Для написания кода использовался *Sublime Text 3*. Это кроссплатформенный проприетарный текстовый редактор. Он поддерживает большое количество языков программирования и имеет возможность подсветки синтаксиса для CSS, HTML, Java, JavaScript, PHP и др. В дополнение к тем языкам программирования, которые включены по умолчанию, пользователи имеют возможность загружать плагины для поддержки других языков. Кроме того, можно устанавливать дополнительные расширения для более удобной и быстрой работы с кодом.

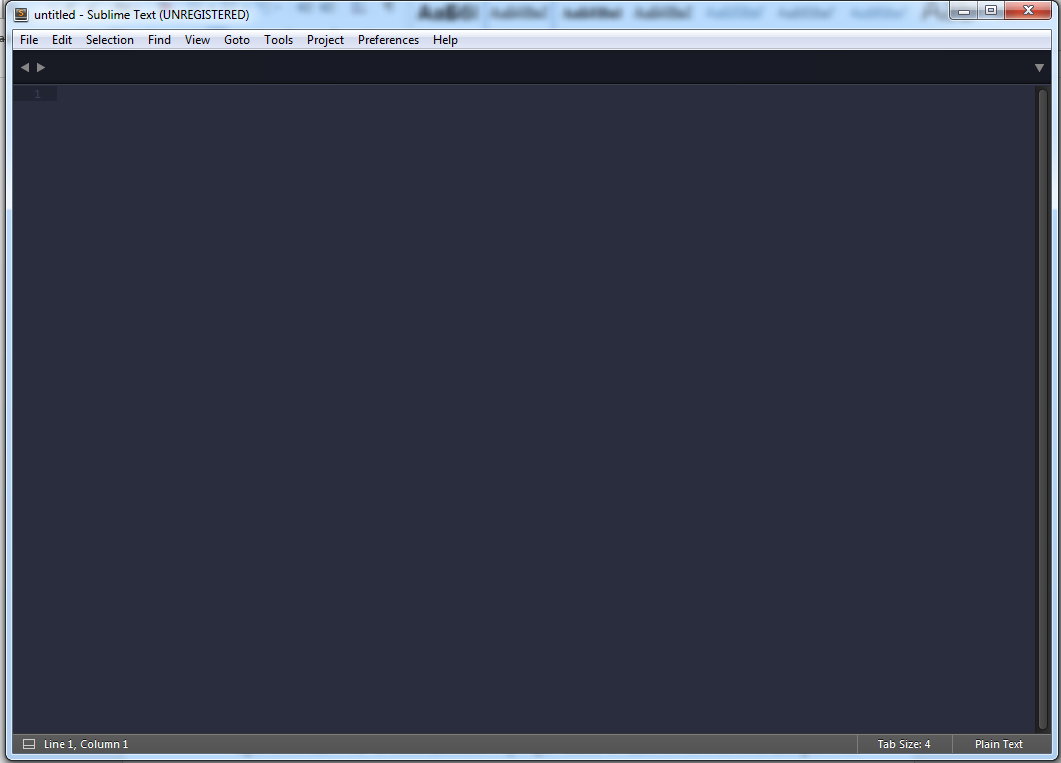


Рисунок 9 – Окно программы Sublime Text 3

В HTML-документе имеются формы, которые необходимо обрабатывать при помощи PHP и базы данных MySQL. Был создан пользователь для работы с БД с именем user и паролем 1234, так же была создана база данных под названием steak\_house. Для обращения к указанной БД написан код PHP, который сохранен в отдельном файле. Этот файл при помощи функции include\_once подключается в нужные нам в дальнейшем PHP-документы.

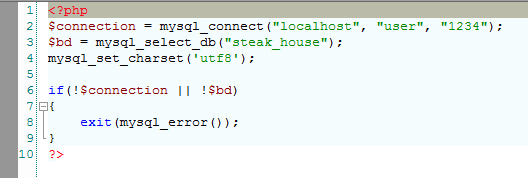


Рисунок 10 – PHP-код подключения к созданной БД

Сами файлы PHP создавались в CodeLobster:

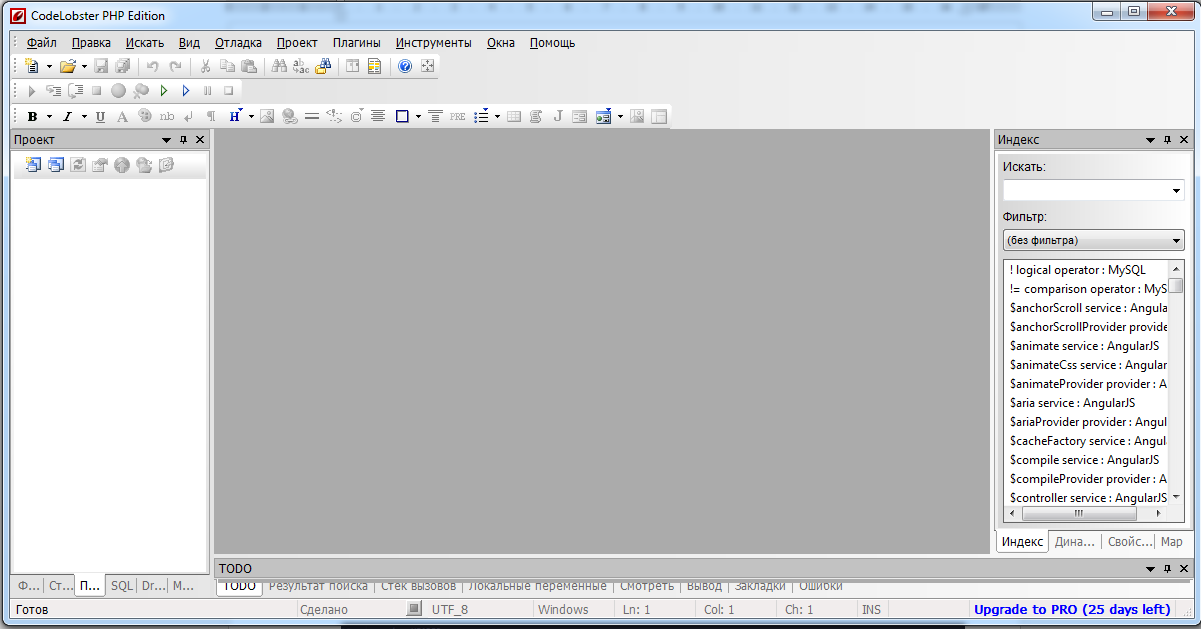


Рисунок 11 – Окно программы CodeLobster

### **2.2 Поэтапная разработка сайта**

### **2.2.1 «Шапка» сайта**

Сайт разрабатывался поэтапно в соответствии с представленными частями макета (см. рисунок 1 – 8).

На рисунке 1 представлена «шапка» сайта. Она имеет логотип фирмы, основной заголовок сайта и слоган. Так же внизу добавлена навигация по сайту.

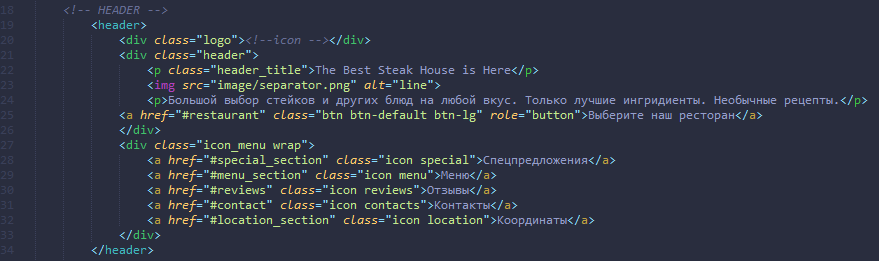


Рисунок 12 – HTML-код «шапки» сайта

Навигация создана при помощи «якорей». Это закладка с уникальным именем на определенном месте веб-страницы, предназначенная для создания перехода к ней по ссылке. Каждый элемент навигации перенаправляет пользователя к соответствующему разделу сайта. Здесь же добавлен динамический элемент: при наведении мышкой на элементы навигации они теряют прозрачность.

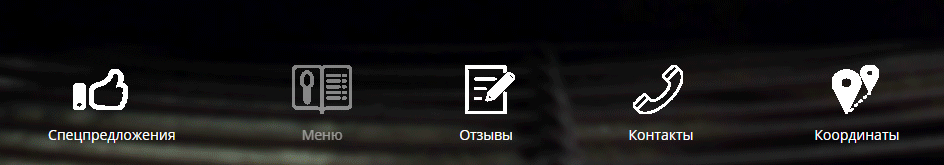


Рисунок 13 – Наведение курсором мышки на элемент навигации

Кнопка, расположенная в центре «шапки», так же является «якорем», она перенаправляет пользователя вниз, к следующему разделу и изменяется при наведении на нее курсором мышки. Кнопка создана при помощи фреймворка Bootstrap, оформление сделано при помощи собственных стилей.



Рисунок 14 – Изменение фона и цвета текста кнопки при наведении на нее курсором мышки

### **2.2.2 Раздел «Ресторан»**

Раздел «Ресторан» приведен на рисунке 2. В данном разделе приводится цитата шеф-повара, рассказывающая про ресторан.

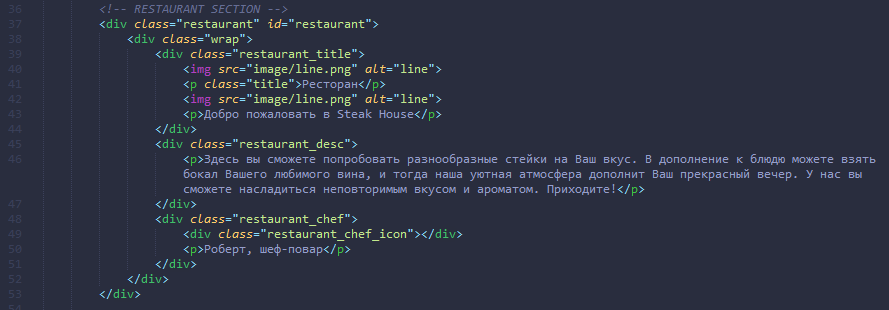


Рисунок 15 - HTML-код раздела «Ресторан»

### **2.2.3 Раздел «Спецпредложения»**

Раздел представляет собой перечень наименований продукции, которая предлагается по сниженной цене (см. рисунок 3).



Рисунок 16 – HTML-код раздела «Спецпредложения»

Каждая продукция сопровождается иллюстрацией, на которой показан продукт. Для выделения, что на данный текст можно нажать применяется динамический эффект при наведении мышкой: у текста появляется подчеркивание.

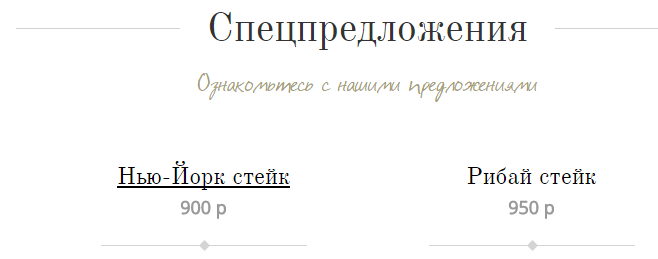


Рисунок 17 – Изменение стиля текста при наведении на него курсором мышки

При нажатии на название продукта изображение, находящееся слева, изменяется в соответствии с выбранным продуктом. Данная технология написана на JavaScript. Здесь используется функция onclick, функция создания нового элемента – createElement и вставка этого элемента в указанный тег с помощью метода appendChild. Таким образом, при загрузке страницы скрипт сохраняет в переменные все указанные на странице продукты, создает элемент image и далее при нажатии на наименование подключает адрес нужного изображения и добавляет его в левую область, заменяя изначальное изображение. Данный скрипт сохранен в файл с разрешением js и названием «image». Он находится в представленном архиве в папке «js».



Рисунок 18 - JavaScript-код

### 

### **2.2.4 Раздел «Меню»**

Раздел «Меню» изображен на рисунке 4. Изначально на сайте меню свернуто. В зависимости от того, какое именно меню пользователь хочет просмотреть, открывается меню стеков, меню салатов или меню напитков. Чтобы подчеркнуть, что меню разворачивается, текст под изображением продуктов при наведении на него мышкой меняет цвет с черного на оранжевый.



Рисунок 19 – Динамическое изменение цвета текста при наведении на него курсором мышки

Открытие меню происходит при нажатии на соответствующее изображение или текст под ним. Меню написано с использованием jQuery. При помощи данной библиотеки можно быстро и просто обратиться к нужному элементу в дереве DOM, а подходящие методы и функции позволяют создать необходимый эффект. Здесь применяется функция click, функция slideUp и slideToggle, оператор if, фильтр visible. Таким образом, при нажатии на выбранное меню, скрипт при помощи оператора if и фильтра visible проверяет показаны ли другие два меню, если да, то он их прячет с помощью функции slideUp и разворачивает выбранное меню при помощи функции slideToggle. Если два других меню не открыты, то скрипт разворачивает выбранное меню с помощью функции slideToggle. Так же, последняя функция позволяет свернуть выбранное меню повторным на него нажатием. Итак, все меню сворачивается и разворачивается.

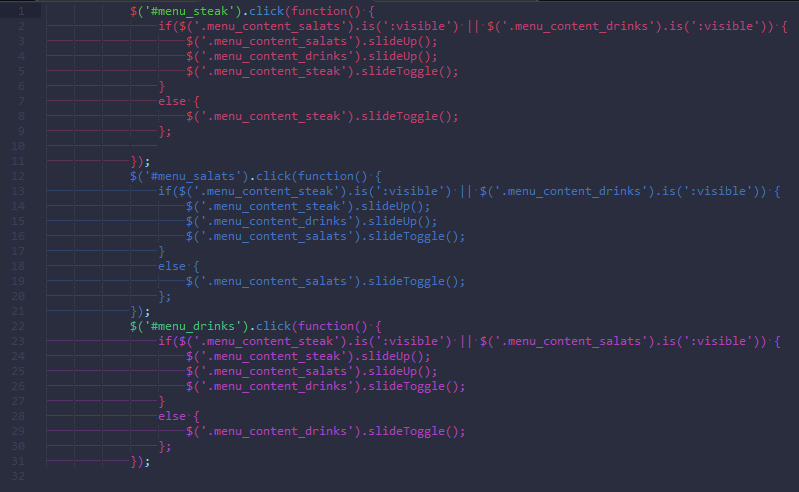


Рисунок 20 – jQuery-код

Само меню для точного отображения и правильного выравнивания написано при помощи модульной сетки Bootstrap. Благодаря такой сетке довольно просто создать красиво выровненное меню. Создается первая строка с классом col-md-12, в которую помещаются две строки. Первая используется для названий блюд в двух частях меню – левого и правого, используя две ячейки одинакового размера, вторая со сдвигом вправо. Вторая строка применяется для описания блюд и указания цены, так же для двух частей меню. Здесь используется четыре ячейки, две последние - со сдвигом вправо. И так далее создаются пять строк для каждого меню.

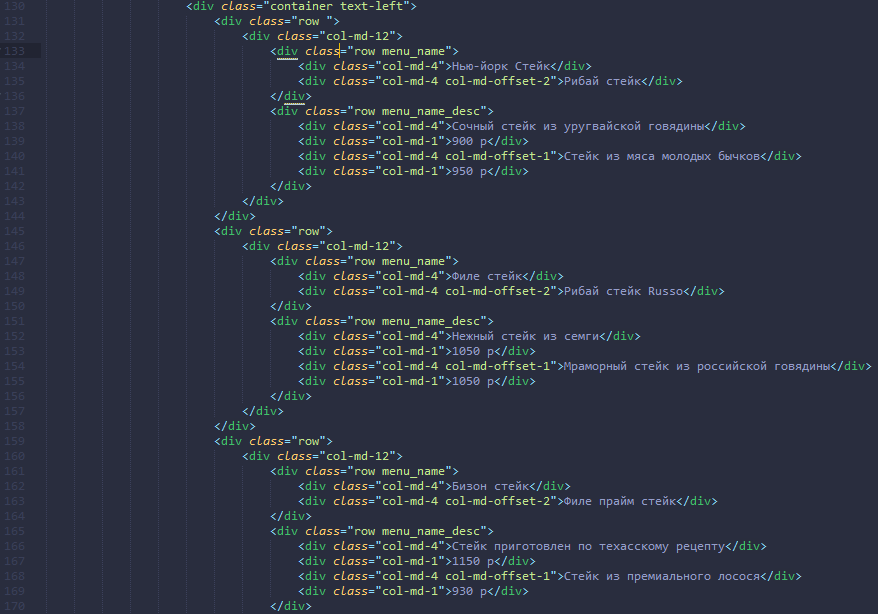


Рисунок 21 – Часть HTML-кода с использованием модульной сетки Bootstrap

### **2.2.5 Раздел «Отзывы»**

На рисунке 5 представлен раздел «Отзывы». На сайте представлено три отзыва, которые в виде слайдов автоматически сменяют друг друга. Слайд с отзывами написан при помощи Bootstrap. Это обосновано тем, что данный фреймворк уже имеет написанный код для такого эффекта. Вставляем его в HTML-документ. Исходный код, благодаря установленным классам, имеет стили, которые прописаны в файле фреймворка. Эти стили были изменены на более подходящие к данной веб-странице. Кроме того, были изменены боковые стрелки, позволяющие переключаться между отзывами.

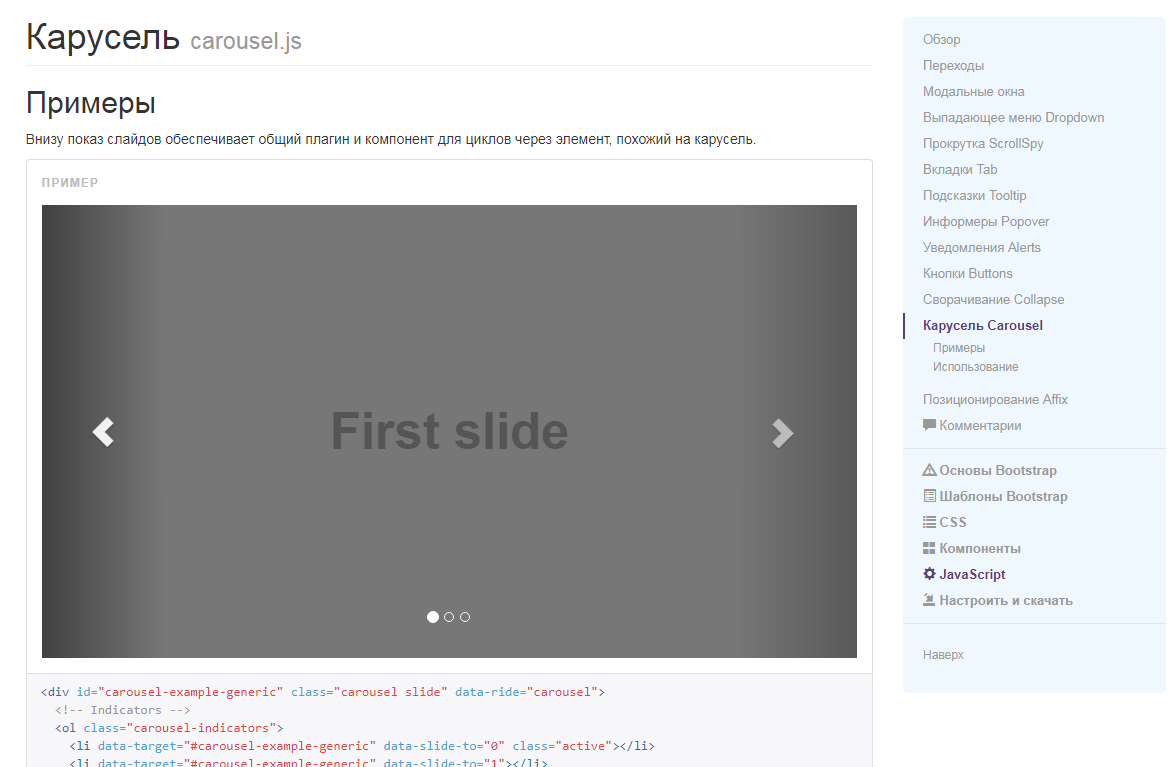


Рисунок 22 - Окно сайта Bootstrap, где рассказано как применять «карусель»

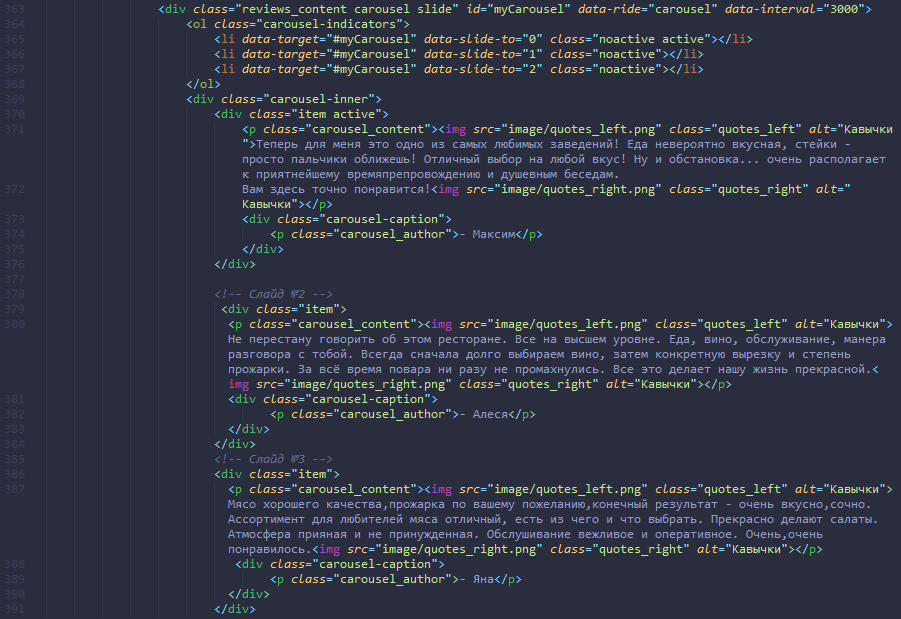


Рисунок 23 – часть HTML-кода, где прописан код для слайдера

### **2.2.6 Раздел «Контакты»**

Раздел «Контакты» (см. рисунок 6), который включает в себя номер телефона ресторана, блок с информацией «Счастливые часы» и формой для подписки на новости. Форма предлагает пользователю написать свой e-mail и получать на него последние новости ресторана. Для обработки формы используется PHP. В базе данных создается таблица sing\_up, в которую будут вносится указанные e-mail при нажатии на кнопку на веб-странице. При корректной обработке формы после нажатия на кнопку в соседней вкладке откроется новое окно. В нем будет сообщено, что указанный адрес почты добавлен в новостную рассылку. Код записан в файл sing-up.php, который приложен в архиве в папке php.

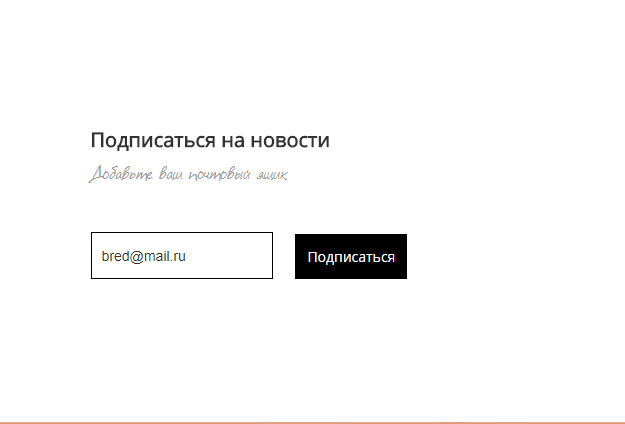


Рисунок 24 – Введение e-mail в форму и динамический эффект кнопки при наведении на нее мышкой

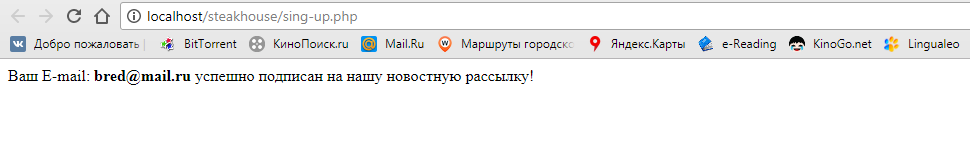


Рисунок 25 – Новое окно, открывающее при правильной работе PHP

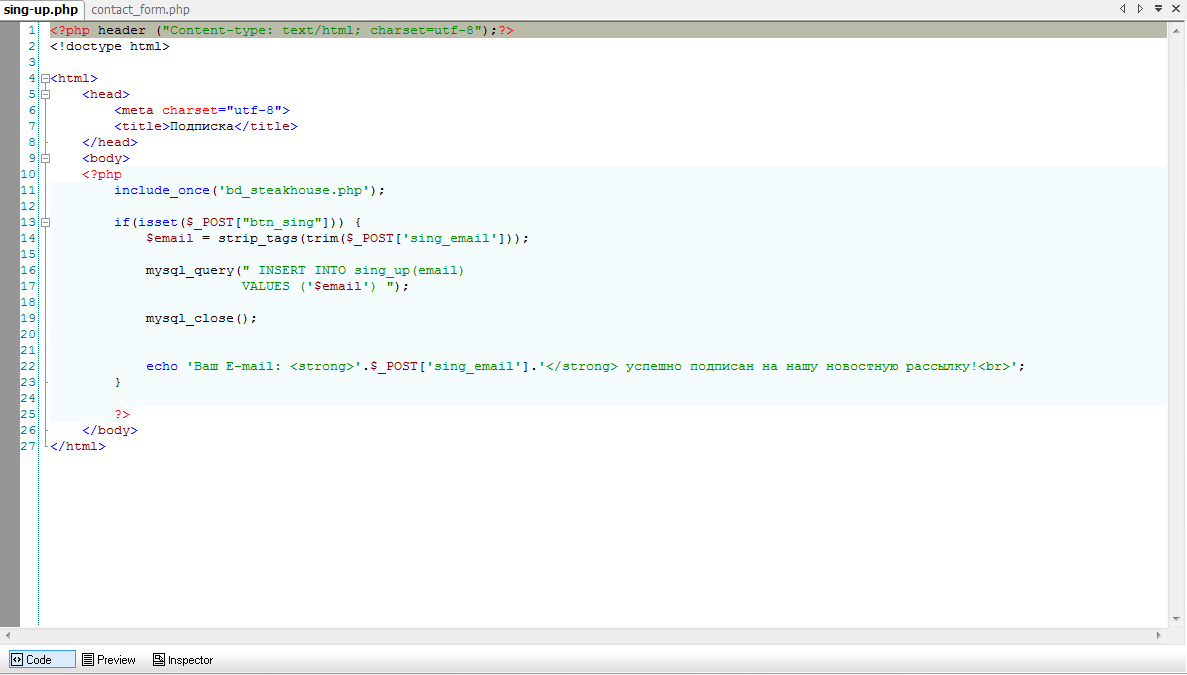


Рисунок 26 – PHP-код

Сама кнопка имеет динамический эффект при наведении: изменяет фоновый цвет и цвет текста (см. рисунок 24). Кнопка создана при помощи фреймфорка Bootstrap, оформление сделано при помощи собственных стилей.

### **2.2.7 Раздел «Координаты»**

Раздел «Координаты» представлен на рисунке 7. Здесь имеется контактная форма для отправки различных сообщений или отзывов, информация о времени работы, адрес, e-mail ресторана и ссылка на Google карты с точными координатами местонахождения ресторана.

Для работы формы так же применяется PHP. В той же базе данных создается таблица contact\_form. В нее записываются данные из формы: e-mail, имя, тема сообщения и само сообщение. Обрабатывается форма при нажатии на кнопку. При правильной работе PHP в соседней вкладке откроется окно, где будет написано, что сообщение отправлено.

Данный код записан в файл contact\_form.php, который приложен в архиве в папке php.

Кнопка формы оформлена через фреймворк Bootstrap с оформлением при помощи собственных стилей.

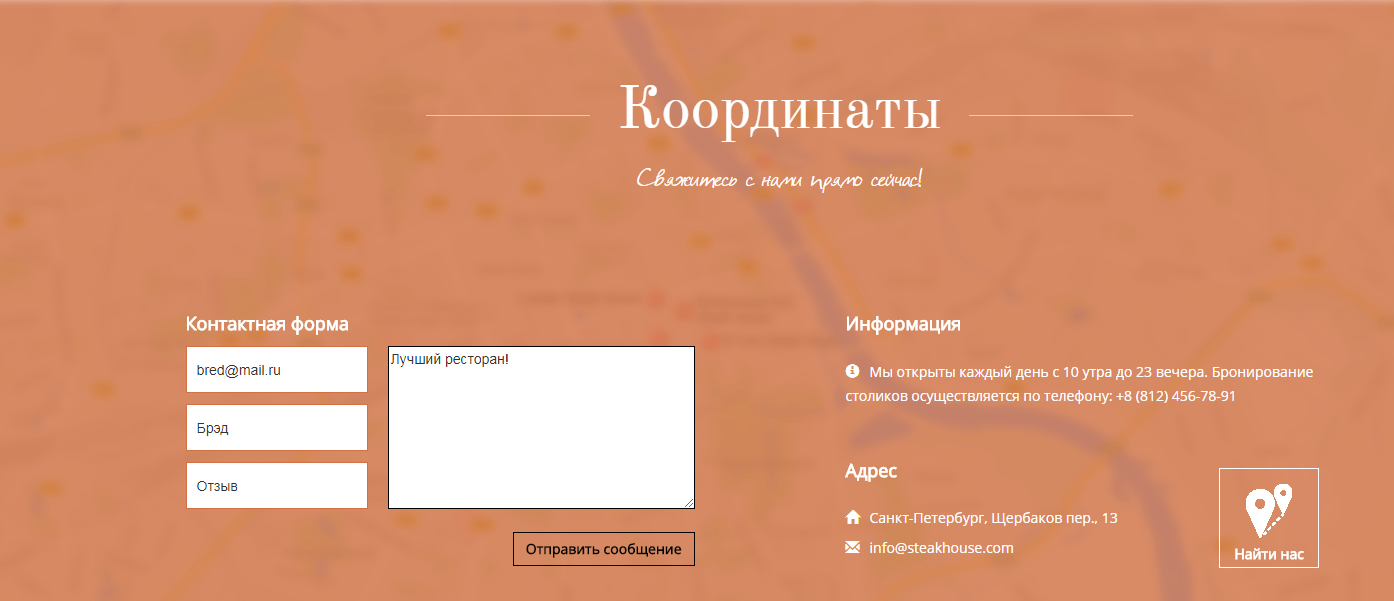


Рисунок 27 – Заполненная форма и динамический эффект кнопки

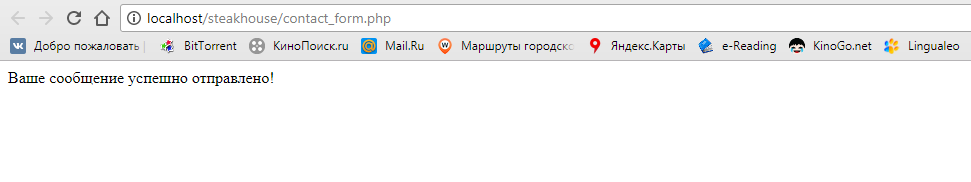


Рисунок 28 – Новое окно, открывающее при правильной работе PHP



Рисунок 29 – PHP-код

Для красивого отображения важной информации, адреса и почты ресторана используются иконки из Bootstrap.

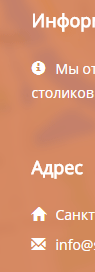


Рисунок 30 – Иконки на веб-странице

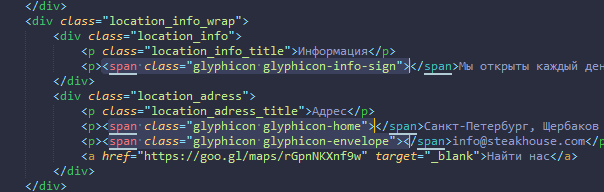


Рисунок 31 – HTML-код иконок

Ссылка на Google карты сделала при помощи гиперссылки с адресом, взятым из Google Карты. Страница с картой открывается в соседнем окне.

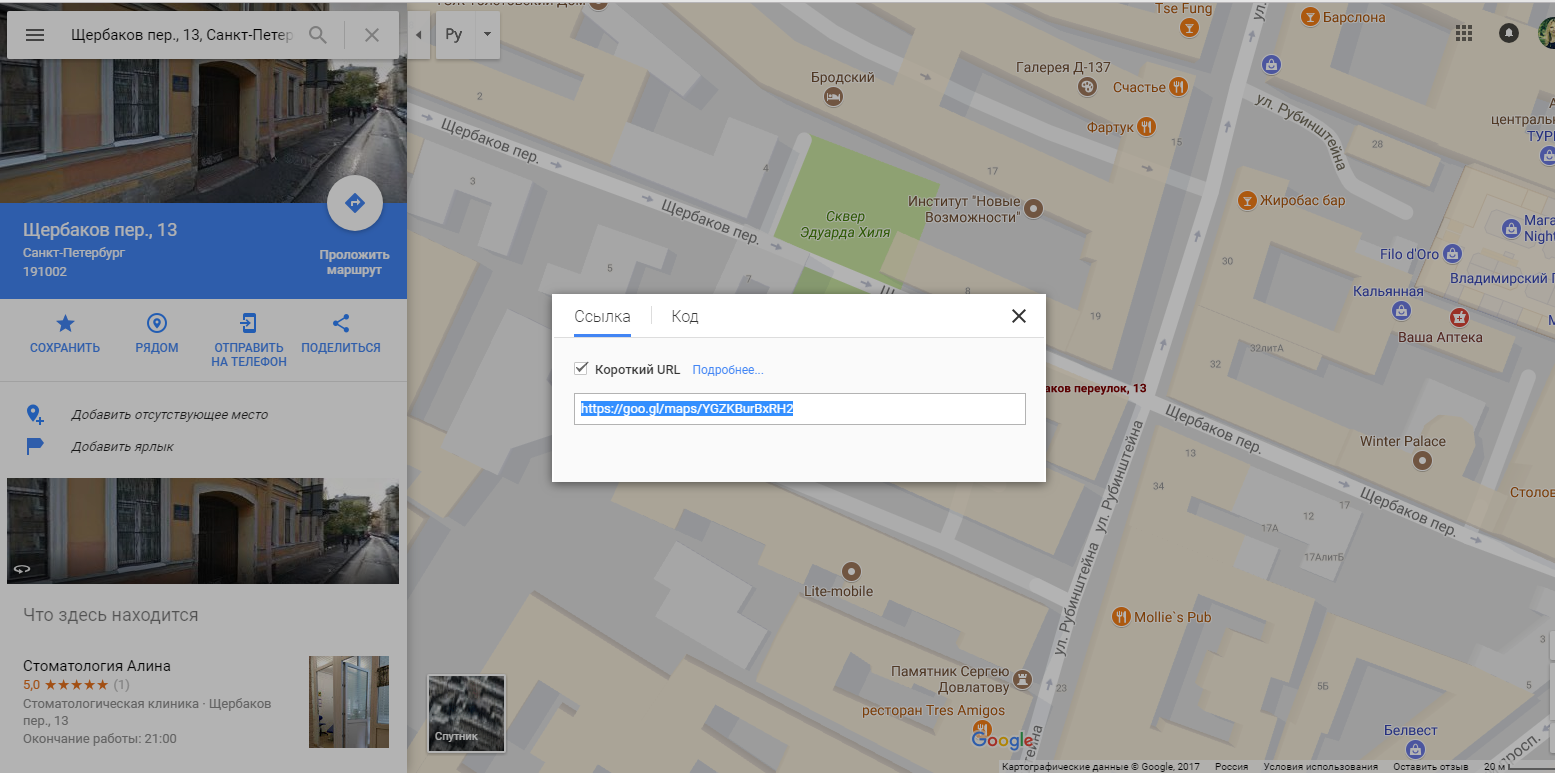


Рисунок 32 – Адрес карты для вставки в HTML-документ

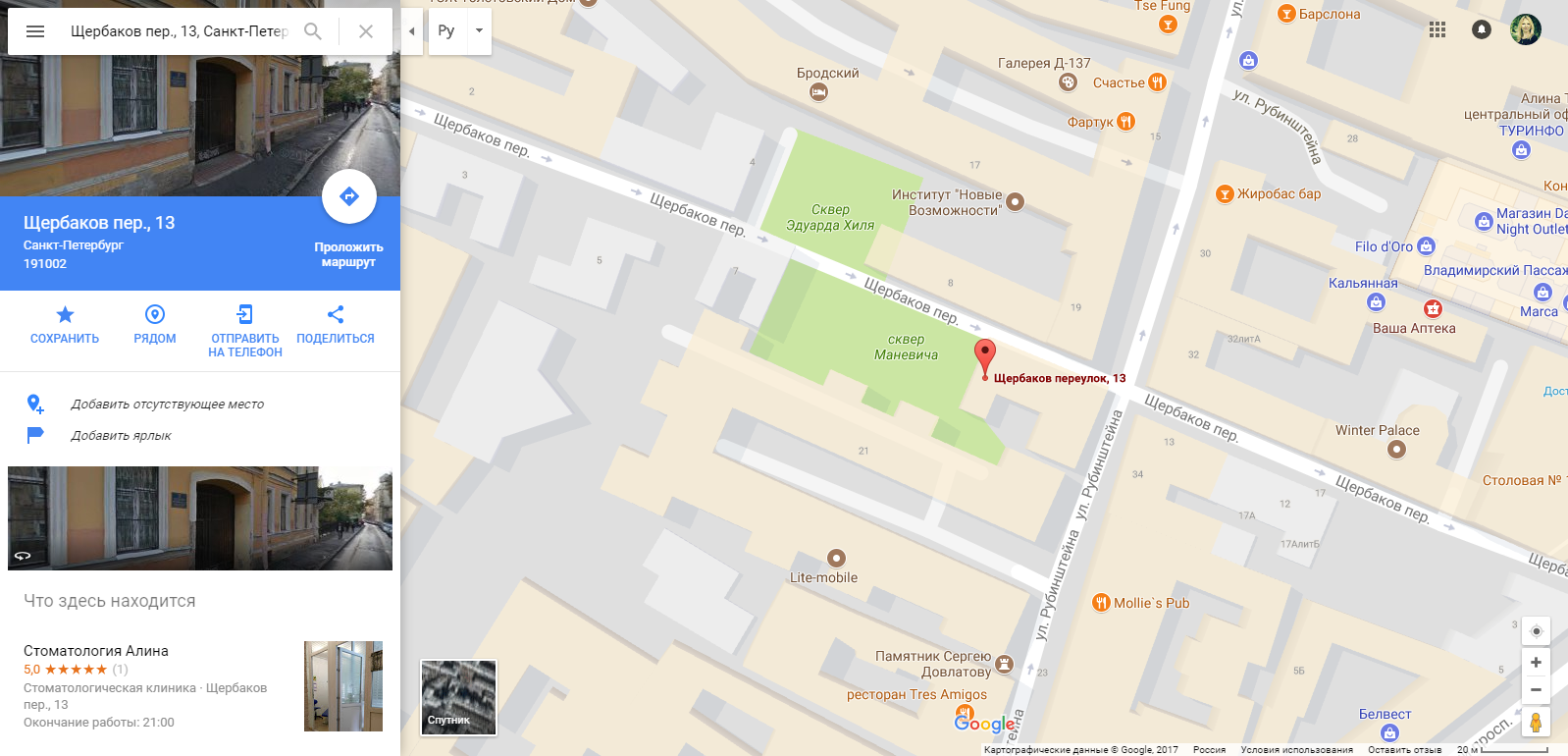


Рисунок 33 – Открывшаяся карта в соседнем окне

Сама ссылка сделана при помощи иконки, взятой из макета страницы и текста. Она имеет динамический эффект при наведении на нее мышкой: изменяет цвет на черный.

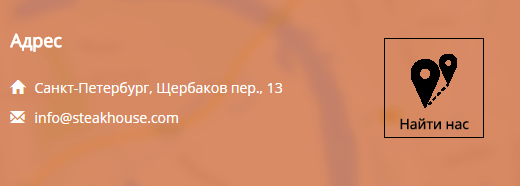


Рисунок 34– Динамический эффект ссылки на карту

### **2.2.8 Раздел «Подвал»**

Последним разделом сайта является «подвал». В него включена навигация, которая повторяет навигацию в начале страницы. Ссылки в навигации так же выполнены при помощи «якорей». Они имеют динамический эффект: при наведении цвет ссылок изменяется на оранжевый.

Так же в конце страницы указан год создания.

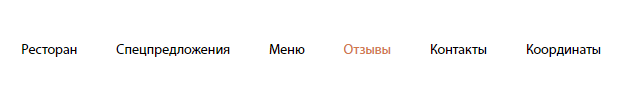


Рисунок 35 – Динамический эффект ссылок при наведении на них курсором мыши

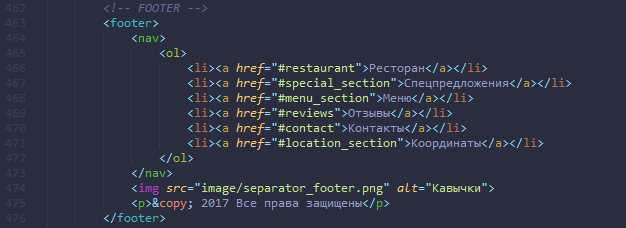


Рисунок 36 – HTML-код «подвала» сайта

## **Оптимизация и продвижение готового сайта**

### **Оптимизация контента**

Поисковая оптимизация контента очень важный шаг, который поможет получать максимальное количество посетителей из поисков, затратив время лишь однажды. Оптимизация так же производится в несколько этапов. Для начала необходимо определиться с поисковыми запросами, на которые будем ориентироваться при оптимизации сайта.

После этого с помощью Яндекс.Директа можно узнать самые популярные запросы. Перефразируя их, и введя в графу для поиска, можно получить информацию об уровне конкуренции по разным вариантам одного и того же запроса. Вполне естественно, что проще всего оптимизировать страницы и добиться первого места (или, по крайней мере, попасть в первую десятку) по менее популярным запросам.

После того, как все необходимые исследования проведены и отобраны несколько поисковых запросов, на которые будем ориентироваться, можно приступать к оптимизации страниц самого сайта.

На положение страницы при выдаче результатов влияет несколько её элементов:

* Title;
* Keywords;
* Description;
* заголовок на самой странице;
* количество ключевых слов, их расположение и частота.

В title лучше всего ввести сам поисковый запрос, или несколько запросов. При этом можно вводить как прямой запрос, так и ответ на него. Title нельзя делать длиннее 256 символов.

В keywords нужно через запятую вписать все слова, содержащиеся в title, и в заголовке на странице. Добавьте сюда слова, которые часто встречаются на странице, даже если они не имеют прямого отношения к поисковому запросу. Слова перечисляются в порядке убывания степени их важности для данной страницы.

Description играет роль не столько для поисковика (хотя тот тоже мимо него не пройдёт), сколько для потенциального посетителя странички. Когда человек, при просмотре результатов поиска, выбирает сайт для посещения, перед его глазами только title и description, поэтому, именно в описании нужно убедить человека, что ему нужно именно на этот сайт.

Самое сложное - это подобрать нужное количество ключевых слов. Здесь нет универсального рецепта, до сих пор существуют много мнений по поводу их оптимального количества. Пару ключевых слов нужно разместить в первых двух-трёх предложениях текста, а затем распределить их равномерно по всему тексту. Не стоит концентрировать ключевые слова в конце страницы, писать несколько штук подряд. Если ключевое слово является ссылкой, то это может повысить релевантность страницы, на которой размещена данная ссылка по некоторым запросам.

Таким образом, основываясь на изложенных основных этапах оптимизации контента проведем оптимизацию данного сайта.

В заголовке сайта, title, укажем название ресторана: «Ресторан Steak House». Для большей узнаваемости добавим иконку сайта, что бы она показывалась рядом с названием при открытии сайта в браузере:



Рисунок 37 – Отображение иконки и названия сайта в браузере

Для того, чтобы отобрать необходимые запросы воспользуемся сервисом Яндекс.Вордстат. В поле «Ключевые слова и словосочетания» вводим слово «ресторан стейков», нажимаем кнопку «Поиск» и отбираем слова, которые больше относятся к контенту нашей страницы. Получится вот такой список ключевых запросов: [стейк ресторан](https://wordstat.yandex.ru/#!/?words=%D1%81%D1%82%D0%B5%D0%B9%D0%BA%20%D1%80%D0%B5%D1%81%D1%82%D0%BE%D1%80%D0%B0%D0%BD), ресторан стейк хаус, ресторан со стейками, ресторан стейк хаус меню, бизон стейк ресторан, ресторан стейк хаус спб, стейковый ресторан.

Исходя из этого и уже имеющегося контента на странице запишем в keywords следующие ключевые слова: ресторан Steak House, ресторан стейков, ресторан стейков меню, ресторан стейков спб, уютный ресторан, меню стейков.

Далее необходимо добавить описание для страницы. Как было сказано при запросе в поисковых системах пользователь увидит заголовок и описание, благодаря которому он будет выбирать тот или иной сайт. Поэтому описание должно быть привлекающим внимание. Исходя из всей концепции сайта можно добавить следующее описание: «Стейк Хаус в Санкт-Петербурге предлагает большой выбор стейков по самым необычным рецептам и недорогим ценам».

На протяжении всей страницы в тексте встречаются необходимые ключевые слова.

### **Продвижение сайта**

Для правильного продвижения сайта необходимо верно спланировать рекламную кампанию в интернете. Одним из важных моментов является постановка целей и задач кампании. Здесь же необходимо определить целевую аудиторию сайта.

Далее основываясь на поставленных задачах необходимо определить набор инструментов. Например, медийная реклама больше подходит для продвижения бренда, в то время как контекстная реклама поможет быстро поднять посещаемость и продажи сайта.

Основным объединяющим звеном между различными инструментами является «message» компании, то есть то, что Вы хотите донести до своей целевой аудитории. Он выражается в самом послании, стиле его оформления и т.д. Формулировка сообщения, «message» кампании – также крайне важный этап. Сообщение должно быть простым, понятным, но не тривиальным.

Следующим этапом является подбор площадки для проведения кампании. На данном этапе необходимо найти в сети Интернет все возможные целевые площадки для проведения Вашей кампании, а далее отобрать наиболее подходящие по цене, качественным характеристикам и предоставляемым рекламным возможностям.

И наконец, расчет бюджета рекламной кампании.

Проведем рекламную кампанию для данного сайта.

На сайте представлен новый ресторан, который недавно вышел на рынок. Следовательно, целями кампании будут:

* увеличить продажи;
* увеличить аудиторию.

Задачами же кампании будут:

* обеспечить сайт компании максимально целевым трафиком;
* быстрый эффект;
* долгосрочный эффект;
* обеспечить информационное присутствие компании в Интернете.

Далее подберем инструменты, которые помогут провести рекламную кампанию.

Для увеличения продаж:

* контекстная реклама;
* SEO-оптимизация;
* электронная рассылка.

Для обеспечения информационного присутствия компании в Интернете:

* вирусный маркетинг (использовать осторожно);
* медийная реклама (одновременно и вклад в создание бренда);
* PR-акции, распространение статей и отзывов о компании в интернете, SMO продвижение.

Следующим этапом необходимо разработать «message» кампании. Для этого необходимо просмотреть рынок и проанализировать конкурентов. В большинстве своем конкуренты основываются на одних и тех же блюдах и довольно высоких ценах. Наша рекламная кампания может сделать акцент на новых интересных блюдах с секретными ингредиентами, различными акциями и сравнительно невысокими ценами.

Таким образом, можно анонсировать в рекламной кампании:

* среднерыночные цены;
* креативный подход к изготовлению блюд;
* натуральные и свежие ингредиенты;
* акции, бонусы, карты постоянных посетителей.

Подберем подходящие площадки для продвижения. Для этого через поисковик ищем сайты, анализируем их посещаемость и стоимость размещения рекламы.

После анализа рынка были выбраны следующие площадки:

* <https://saintpetersburg.zagranitsa.com/>;
* <https://www.restoclub.ru/>;
* <https://www.restorating.ru/>;
* <http://www.the-village.ru/>;
* <http://andychef.ru/>.

Где люди выбирают себе рестораны? Чаще всего заходят на сайты, где собраны практически все рестораны города. Поэтому, вполне логично, разместить на таких сайтах рекламу нового ресторана. Кроме того, на некоторых из них можно разместить PR-статью.

Таким образом, посчитаем необходимый бюджет для рекламной кампании:

* баннеры – разместим их на вышеперечисленных сайтах – 10 тыс.;
* медийная реклама – так же разместим на указанных сайтах – около 250 тыс.;
* контекстная реклама – бюджет 200 тыс.;
* SEO-продвижение – около 30 тыс.;
* PR-статьи, размещение их на сайтах – около 7 тыс.

Итого, получается примерно 500 тыс. Такой бюджет необходим для проведения рекламной кампании в интернете.

## **Заключение**

В ходе проделанной работы был создан полнофункциональный веб-сайт, полностью готовый к применению. С его помощью посетители могут ознакомиться со всей необходимой информацией об ресторане, подписаться на его новости и оставлять отзывы.

Разработанный сайт удовлетворяет всем поставленным целям и задачам. Создан дизайн макета сайта. Сам сайт разработан при помощи всех необходимых инструментов, указанных в задачах работы. Он включает в себя интерактивные элементы и динамические эффекты.

Так же была проведена оптимизация сайта и рассчитан бюджет рекламной кампании по продвижению сайта в интернете.