МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ   
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных технологий

Кафедра Информационных систем и технологий

Специальность 1-40 01 01 «Программное обеспечение информационных технологий»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА КУРСОВОГО ПРОЕКТА**

по дисциплине «Компьютерные языки разметки»

Тема «Web-сайт гостиницы»

**Исполнитель**

студент(ка) 1 курса 5 группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н. А. Песецкий

подпись, дата

**Руководитель**

ассистент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н. В. Ржеутская

должность, учен. степень, ученое звание подпись, дата

Допущен(а) к защите\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

дата, подпись

Курсовой проект защищен с оценкой

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н. В. Ржеутская

подпись дата инициалы и фамилия

Минск 2022

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования   
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий   
Кафедра информационных систем и технологий

Утверждаю

Заведующий кафедрой

\_\_\_\_\_\_\_\_\_\_\_ В.В. Смелов

подпись инициалы и фамилия

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

**ЗАДАНИЕ**

**на курсовое проектирование**

по дисциплине«Компьютерные языки разметки»

Курс 1 Группа: 5 Специальность: 1-40 01 01 «Программное обеспечение информационных технологий»

Студент: Песецкий Никита Андреевич

**1. Тема:** Web-сайт гостиницы

**2. Срок выполнения курсового проекта** : с 14 февраля 2022 г. по 10 мая 2022 г.

**3. Технические требования :**

3.1 Прототип веб-сайта должен быть разработан с использованием графических редакторов Figma/Adobe XD/Sketch.

3.2 Для хранения данных должен быть использован XML-формат.

3.3 Разметка содержания сайта должна быть выполнена с применением HTML5 и XML.

3.4 Для описания внешнего вида веб-страниц использовать Sass/CSS3.

3.5 Веб-сайт должен содержать:

– семантические теги HTML5;

– графические элементы в форме SVG;

– несколько веб-страниц;

3.6 Верстка сайта должна быть адаптивной и кроссбраузерной;

3.7 Для тестирования использовать Linter.

3.8 Проект и пояснения к проекту должны быть размещены на GitHub.

**4. Содержание пояснительной записки**

1. Титульный лист;

2. Задание на курсовое проектирование;

3. Введение;

4. Постановка задачи;

5. Проектирование веб-сайта

6. Реализация структуры веб-сайта

7. Тестирование веб-сайта

8. Заключение

9. Список использованных литературных источников

10. Приложения (полный исходный текст программы разработанного приложения с подробными комментариями)

**5. Форма представления на GitHub выполненного курсового проекта:**

– Теоретическая часть пояснительной записки курсового проекта должна быть представлена в формате MS Word.

– Оформление записки должно быть согласно правилам.

– Листинги представляются в приложении.

#### Календарный план

|  |  |  |
| --- | --- | --- |
| № п/п | Наименование этапов курсового проекта | Срок выполнения этапов проекта |
| 1 | Задание на курсовое проектирование. Титульный лист | 14.02.2022 |
| 2 | Введение. Постановка задачи | 22.02.2022 |
| 3 | Проектирование веб-сайта | 28.02.2022 |
| 4 | Реализация структуры веб-сайта на HTML5 и внешнего оформления на Sass/СSS3 | 09.03.2022 |
| 5 | Тестирование веб-сайта | 30.03.2022 |
| 6 | Заключение | 22.04.2022 |
| 7 | Список использованных литературных источников. Приложения | 29.04.2022 |
| 6 | Подготовка и оформление пояснительной записки курсового проекта | 3.05.2022 |
| 7 | Сдача и защита курсового проекта | 10.05.2022 |

**5. Дата выдачи задания** «14» февраля 2022 г.

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н. В. Ржеутская

(подпись)

Задание принял к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата и подпись студента)

**Содержание**

**[Введение](#_Toc98805133)** [5](#_Toc98805133)

**[1. Постановка задачи](#_Toc98805134)** [6](#_Toc98805134)

**[1.1. Обзор аналогичных решений](#_Toc98805135)** [6](#_Toc98805135)

**[1.2. Техническое задание](#_Toc98805136)** [8](#_Toc98805136)

**[1.2.1. Формирование требований к программному продукту](#_Toc98805137)** [8](#_Toc98805137)

**[1.2.2. Формулировка задач программного продукта](#_Toc98805138)** [8](#_Toc98805138)

**[1.3. Выбор средств реализации программного продуктаю](#_Toc98805139)** [8](#_Toc98805139)

**[1.4. Вывод](#_Toc98805140)** [8](#_Toc98805140)

**[2. Проектирование страниц веб-сайта](#_Toc98805141)** [9](#_Toc98805141)

**[2.1. Выбор способа верстки](#_Toc98805142)** [9](#_Toc98805142)

**2. Выбор стилевого оформления.**

**2.3 Выбор шрифтового оформления.**

**2.4 Разработка логотипа.**

**2.5 Разработка пользовательских элементов.**

**2.6 Разработка спецэффектов.**

**2.7 Вывод.**

**3 Программа реализации проекта**

**3.1. Структура HTML-документа**

**3.2. Добавление таблиц стилей Sass и CSS**

**3.3. Использование стандартов XML (SVG)**

**3.4.Вывод**

**4.Тестирование веб-сайта**

**4.1. Адаптивный дизайн веб-сайта.**

**4.2. Кроссбраузерность веб-сайта.**

**4.3. Руководство пользователя.**

**4.4.Вывод**

**Введение**

В нашем современном мире имеется огромный спрос пользователей на кроссплатформенные сайты, которые могли бы адаптивно работать как на компьютере(ноутбуке), так и на мобильном телефоне, но также необходимо учитывать необходимость поддержания кроссбраузерности сайта для возможности получения доступа к сайту из любого браузера и любого устройства. Также в настоящее время существует спрос на сайты журнальных изданий, для того, чтобы быстро публиковать самые свежие материалы, и пользователи имели возможность читать публикации, в момент их появления, не выходя из дома.

В связи с актуальностью данной работы целью курсового проекта является разработка структуры кроссбраузерного веб-сайта на HTML5 с подключением Sass/CSS3.

К задачам курсового проекта относится обзор аналогичных решений, характеристику решаемой задачи, выбор средств реализации; проектирование страниц веб-сайта, включающее выбор способа верстки, стилевого оформления, шрифтового оформления, разработку логотипа, пользовательских элементов, спецэффектов; реализация структуры веб-сайта, включающая листинги структуры HTML-документа, таблиц стилей SASS и CSS3, использования XML (SVG); тестирование веб-сайта, включающее тестирование адаптивного дизайна и кроссбраузерности, руководство пользователя; формирование вывода по каждому разделу; заключение, включающее вывод по проделанной работе.

Веб-сайт представляет собой набор веб-страниц, связанных между собой посредством ссылок. Макет веб-страницы состоят из прототипов, построенных на HTML5, включающих семантические теги, и стилей для элементов, реализованных на SASS и CSS3, и т.д Также веб-страницы могут включать в себя различные медиафайлы, для создания и размещения которых будет использоваться XML (SVG).

**1. Постановка задачи**

**1.1. Обзор аналогичных решений**

В сети интернет существует множество сайтов гостиннц, отелей и хостелов. Большинство из них имеют сайты, ведь рекламировать и предагать свои услуги гараздо легче через интернет, нежели на офлайн точке или через рекламу по телевидинью, радио или газете.

Смотря на многие сайты гостиниц можно заметить, что весь контент размещен по центру, для удобного просмотра. Примером такого дизайна может стать веб-сайт «HOTEL MINSK» (рисунок 1.1, 1.2).

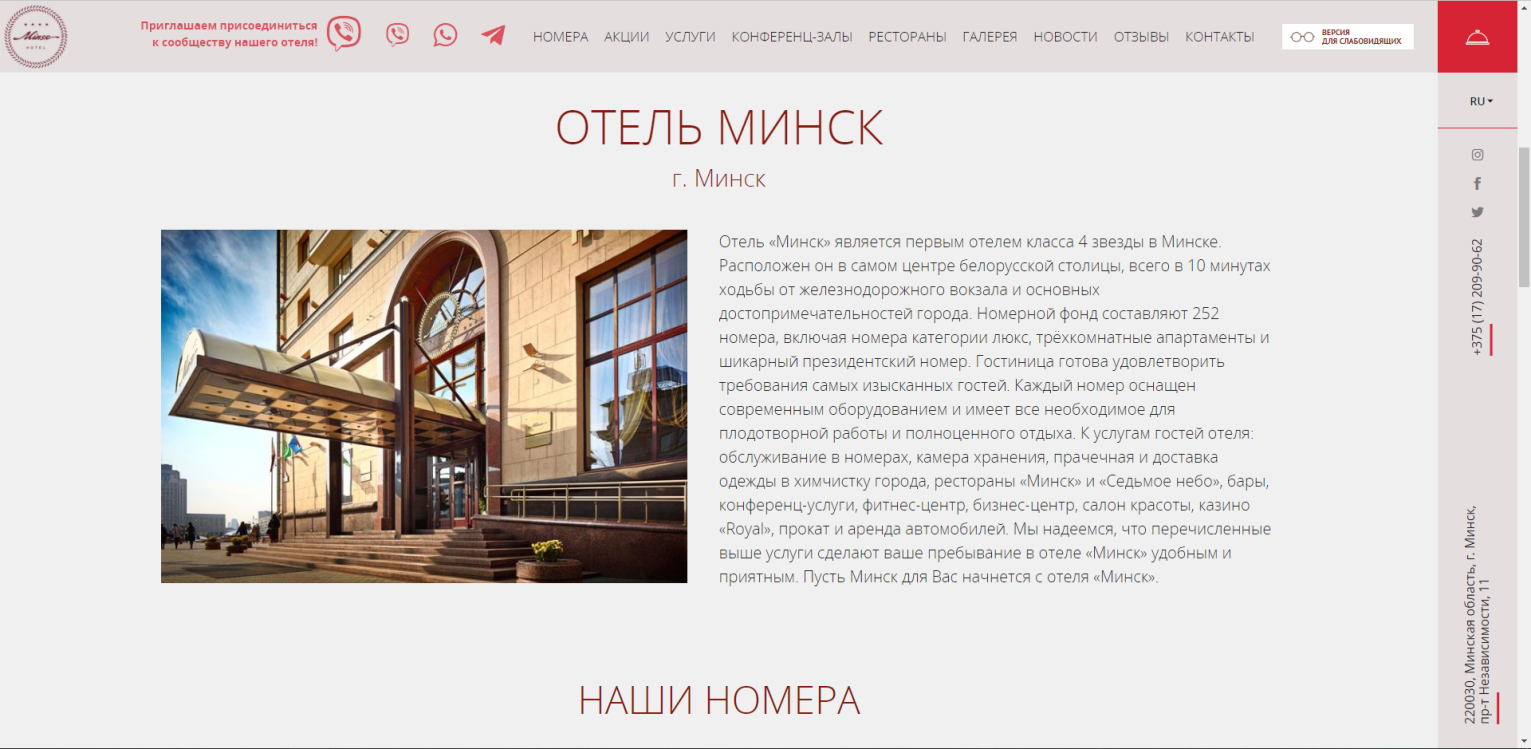
Рисунок 1.1

Рисунок 1.2

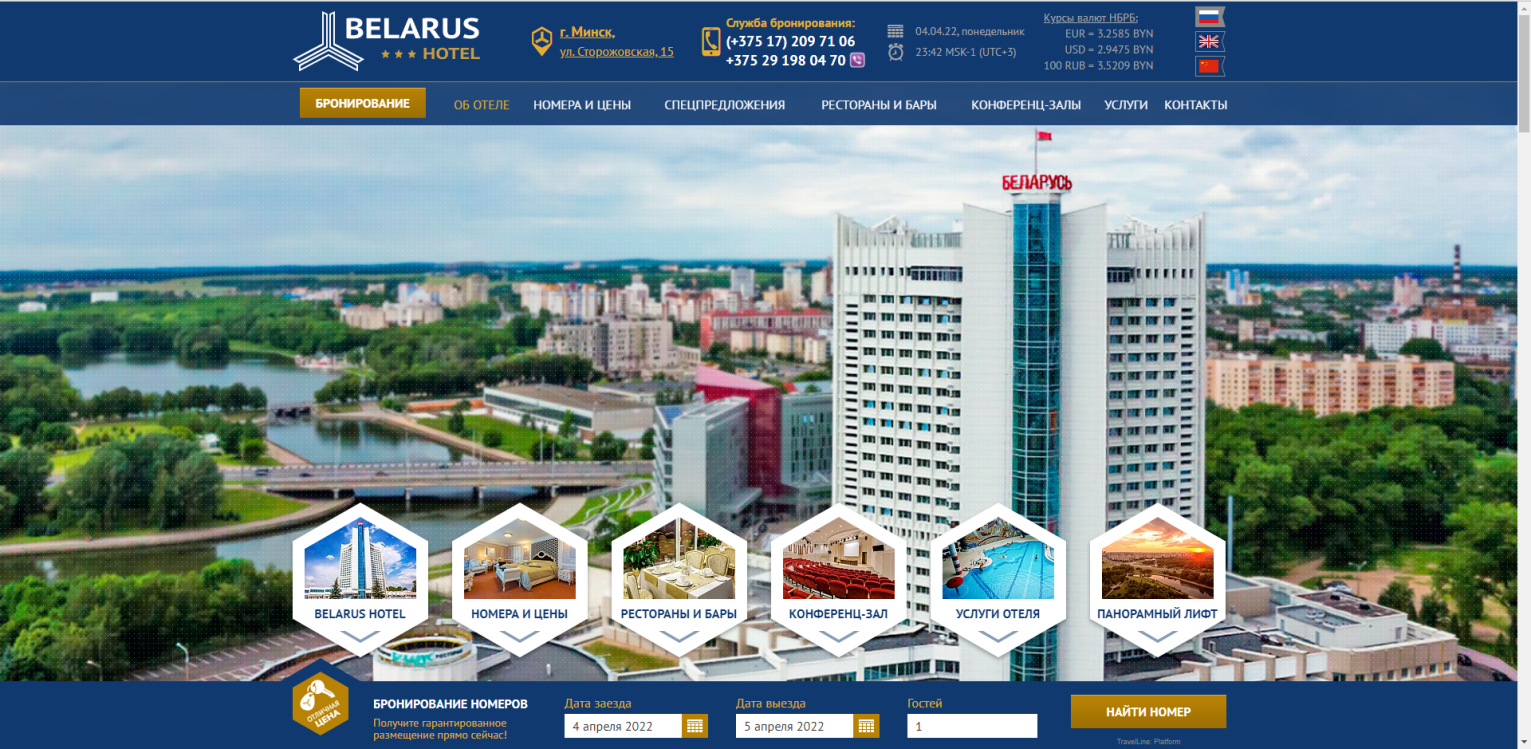


Рисунок 2



Также удобно, когда веб-сайт обладает мобильной версией, для удобного просмотра на всех устройствах. На данном сайте, для создания мобильной версии использованы медиа-запросы (рисунок .2).

В итоге сайт получается доступным для пользователей. В данном курсовом проекте, мы будем стремиться создать подобный сайт.

В качестве примера некачественного веб-сайт можно использовать сайт «Первая миля» (рисунок 1.3).



Дизайн данного сайта абсолютно не привлекает внимания, для чтения информации приходится присматриваться, количество стилей минимально.

Также важно присутствие адаптивности на сайте. Количество пользователей использующих мобильные устройства очень велико, если наш сайт будет не удобен для таких пользователей, мы потеряем часть посетителей сайта.

**1.2. Техническое задание**

**1.2.1. Формирование требований к программному продукту**

В данном проекте требуется создать многостраничный сайт гостиницы. Сайт долен содержать общую информацию о гостинице, такую как контактные данные, описание гостиницы, а также каталог услуг и форму аренды. Основным контентом являются текстовые данные, описания услуг, фотографии комнат.

Главной задачей сайта является предоставление данных о услугах для привлечения внимания потенциальных клиентов. Сайт должен быть интерактивным, содержать актуальную информацию. Данный сайт предназначен для желающих оформить проживание в данной гостинице.

На главной странице меню будет располагаться сверху. В меню будут предоставляться ссылки на основную информацию (Основная страница, аренда, сервисы, галерея, контакты и т.д.).

В мобильной версии (ширина до 560px) страница должна адаптироваться под ширину устройства. Также должна быть разработана версия для планшетного устройства (ширина до 1000px). Начиная с ширины в 1000px должна открываться версия для персонального компьютера.

Необходимо реализовать макеты страниц для главной страницы, страниц об услугах, страница аренды.

На странице «Главная» должна быть расположена основная информация. На странице «Аренды» будет расположена ворма аренды. На странице «Сервисы» расположена информация о услугах.

Основные требования к исполнителю: создание дизайна, проведение анализа существующих решений, создание прототипа, адаптивность и кроссбраузерность сайта (корректное отображение в браузерах, таких как Google Chrome, Microsoft Edge). В проекте используется гипертекстовый язык разметки HTML, CSS/Sass, графические элементы в формате SVG.

**1.2.2. Формулировка задач программного продукта**

Основными задачами курсового проекта является:

* создание многостраничного веб-сайта;
* разработка оптимизированного дизайна для любых устройств;
* реализация слайдеров;
* реализация анимаций;
* разработка адаптивности сайта;
* разработка кроссбраузерности сайта;

**1.3. Выбор средств реализации программного продукта**

Для реализации программного продукта были выбраны следующие инструменты:

* язык разметки HTML5;
* язык стилей CSS3;
* препроцессор SASS;

Курсовая работа выполняется в редакторе кода VS Code.

**1.4. Вывод**

В данном разделе были рассмотрены аналогичные решения, были проанализированы их недостатки и преимущества, для создания в последующем качественного веб-сайта. Были поставлены конкретные задачи для создания интерактивного информационного ресурса. Кроме того, были рассмотрены средства реализации программного продукта, такие как языки разметки, а также редактор кода VS Code. Были проанализированы особенности языков и основные дополнительные средства редактора. Все вышеперечисленные пункты понадобятся для достижения поставленной задачи и для создания в дальнейшем качественного продукта.

**2. Проектирование страниц веб-сайта**

**2.1. Выбор способа верстки**

Адаптивная вёрстка с использованием блочного подхода и flex элементов, также для обозначения основных элементов сайта используются семантические теги.  
 Для создания отдельных блоков страницы применяется Flexbox-верстка. С помощью этой технологии можно очень просто и гибко расставить элементы в контейнере, распределить доступное пространство между ними, и выровнять их тем или иным способом даже если они не имеют конкретных размеров. CSS Flexbox поддерживается всеми используемые на сегодняшний момент современными браузерами (с использованием префиксов: IE10+, Edge12+, Firefox 2+, Chrome 4+, Safari 3.1+, Opera 12.1+, iOS Safari 3.2, Opera mini, Android 2.1+, Blackberry 7+).

Исходя из выбора способа верстки были расположены все основные элементы сайта, представленные в приложение 1 и 2

**2.2 Выбор стилевого оформления.**

Открыл фигмю, хркнул, крякнул, плюнул, вякнуо, заклеил всё скотчем. Создал индивидуальный дизайн.

# **2.3 Выбор шрифтового оформления.**

В данном проекте используются шрифт: Montseratt. Шрифты подключались с помощью библиотеки Google Fonts. Шрифт поддерживает кириллическое написание. Шрифт используется для всего текста.

# **2.4 Разработка логотипа.**

Разработка логотипа осуществлялась в графическом редакторе Figma. Дизайн довольно минималистичен; было принято решение не загромождать его множеством декоративных элементов. Логотип представляет из себя название гостиницы (HOTEL). Надпись «HOTEL» сначала была выполнена в текстовом формате. Для этого использовался шрифт Montseratt. Далее надпись была сгруппирована и переведена в SVG формат. Дизайн логотипа представлен на рисунке 2.2.

# **2.5 Разработка пользовательских элементов.**

В проекте будут представлены такие элементы пользовательского интерфейса, как навигационное меню, карусель (слайдер), форма оформления заказа, меню описаняи услуг, гугл карта.

**2.6 Разработка спецэффектов.**

На страницах веб-сайта присутствуют анимированные кнопок, анимации передвижения картинок в слайдере.

# **2.7 Вывод.**

На данном этапе были созданы прототипы, а затем и макеты сайта для облегчения последующей верстки. Было упорядочено расположение всех элементов на страницах (навигационного меню, логотипа, основного контента каждой страницы, расположение картинок, подвала сайта). Было определено стилевое оформление сайта, то есть основная цветовая гамма, шрифтовое оформление, элементы пользовательского интерфейса, а также эффекты и переходы, придающие динамики страницам.

**3 Программа реализации проекта**

Сайт разрабатывался с помощью редактора от Microfost - Visual Studio Code.

**Visual Studio Code** (VS Code) — [редактор исходного кода](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B4%D0%B0%D0%BA%D1%82%D0%BE%D1%80_%D0%B8%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BE%D0%B4%D0%B0" \o "Редактор исходного кода), разработанный [Microsoft](https://ru.wikipedia.org/wiki/Microsoft" \o "Microsoft) для [Windows](https://ru.wikipedia.org/wiki/Windows" \o "Windows), [Linux](https://ru.wikipedia.org/wiki/Linux" \o "Linux) и [macOS](https://ru.wikipedia.org/wiki/MacOS" \o "MacOS). Позиционируется как «лёгкий» редактор кода для [кроссплатформенной](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%BE%D1%81%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D1%81%D1%82%D1%8C" \o "Кроссплатформенность) разработки [веб](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" \o "Веб-приложение)- и [облачных](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D0%BB%D0%B0%D1%87%D0%BD%D1%8B%D0%B5_%D0%B2%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F" \o "Облачные вычисления) приложений. Включает в себя [отладчик](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BB%D0%B0%D0%B4%D1%87%D0%B8%D0%BA" \o "Отладчик), инструменты для работы с [Git](https://ru.wikipedia.org/wiki/Git" \o "Git), [подсветку синтаксиса](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B4%D1%81%D0%B2%D0%B5%D1%82%D0%BA%D0%B0_%D1%81%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81%D0%B0" \o "Подсветка синтаксиса), [IntelliSense](https://ru.wikipedia.org/wiki/IntelliSense" \o "IntelliSense) и средства для [рефакторинга](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D1%84%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%BD%D0%B3" \o "Рефакторинг). Имеет широкие возможности для [кастомизации](https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D1%81%D1%82%D0%BE%D0%BC%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F" \o "Кастомизация): пользовательские темы, [сочетания клавиш](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%87%D0%B5%D1%82%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BA%D0%BB%D0%B0%D0%B2%D0%B8%D1%88" \o "Сочетание клавиш) и [файлы конфигурации](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB_%D0%BA%D0%BE%D0%BD%D1%84%D0%B8%D0%B3%D1%83%D1%80%D0%B0%D1%86%D0%B8%D0%B8" \o "Файл конфигурации). Распространяется бесплатно, разрабатывается как программное обеспечение с [открытым исходным кодом](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BA%D1%80%D1%8B%D1%82%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5" \o "Открытое программное обеспечение), но готовые сборки распространяются под [проприетарной лицензией](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%BF%D1%80%D0%B8%D0%B5%D1%82%D0%B0%D1%80%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5" \o "Проприетарное программное обеспечение).

**HTML** (Hypertext Markup Language) - это язык гипертекстовой разметки, который используется для структурирования и отображения веб-страницы и её контента. Например, контент может быть структурирован внутри множества параграфов, маркированных списков или с использованием изображений и таблиц данных.

**CSS (Cascading Style Sheets, каскадные таблицы стилей)** — язык описания внешнего вида HTML-документа. Это одна из базовых технологий в современном интернете. Практически ни один сайт не обходится без CSS, поэтому HTML и CSS действуют в единой связке.

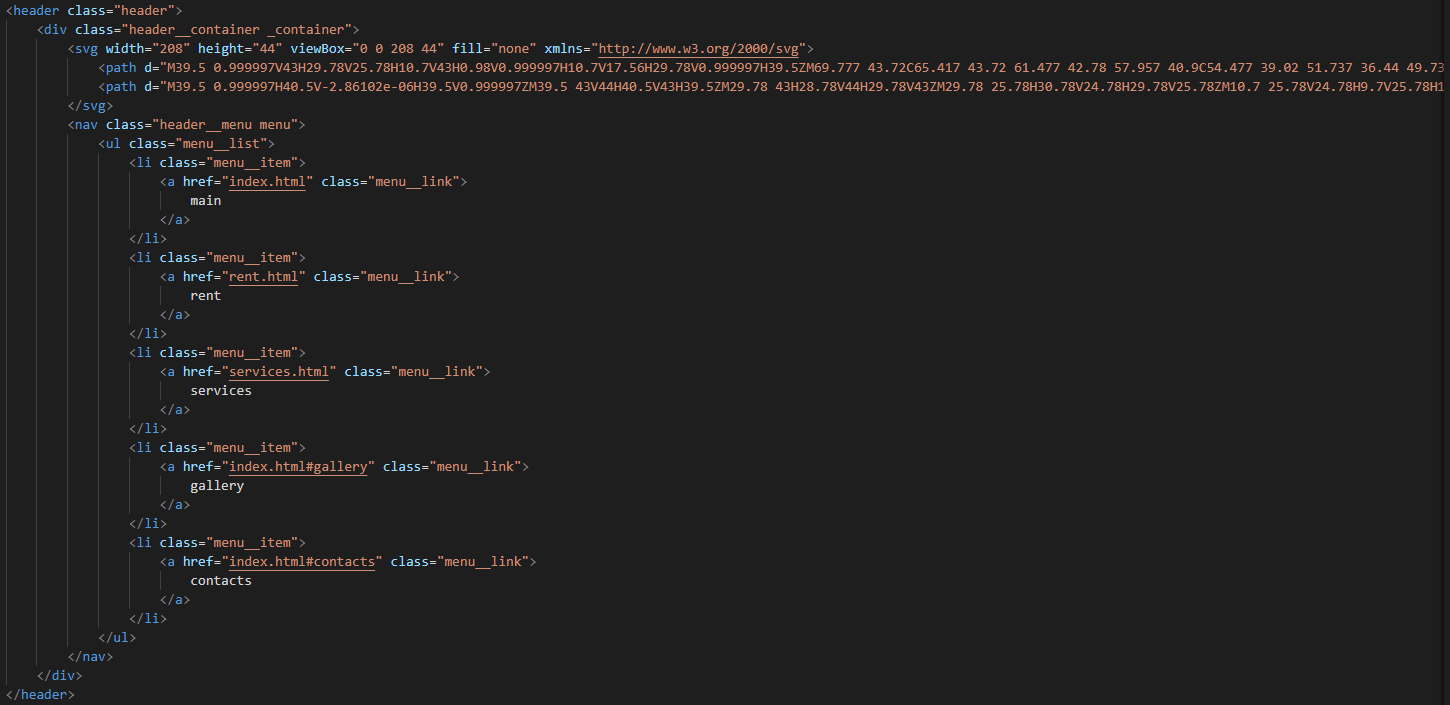
**3.1. Структура HTML-документа**

Структура HTML-документа определяет базовый «скелет» для будущего веб-сайта. Код выполнен по стандарту HTML5 и кратко изложены основные тезисы элементов. На рисунке 3.1 представлена структура одной единицы товара из каталога меню.

Внутри служебного тега head располагается выбранная кодировка (строка 4), присоедините гугл шрифта “Montseratt” (строка 6), подключение каскадной таблицы стилей содержащей общие стили всех страниц (строка 7), личные стили страницы (строка 8), файл содержащий медиазапросы (строка 9). Так же, на 10 строке устанавливает иконка на вкладку веб-сайта (рисунок x). Название страницы (строка 11).

Структура тела документа состоит из семантических тегов header, aside и footer. 

Разработка навигационного меню сайта.

В теге header находится горизонтальная навигационная панель. Она содержит ссылки на все основные разделы веб-сайта, а также логотип и название компании. Навигационное меню выполнено в качестве маркированного списка.

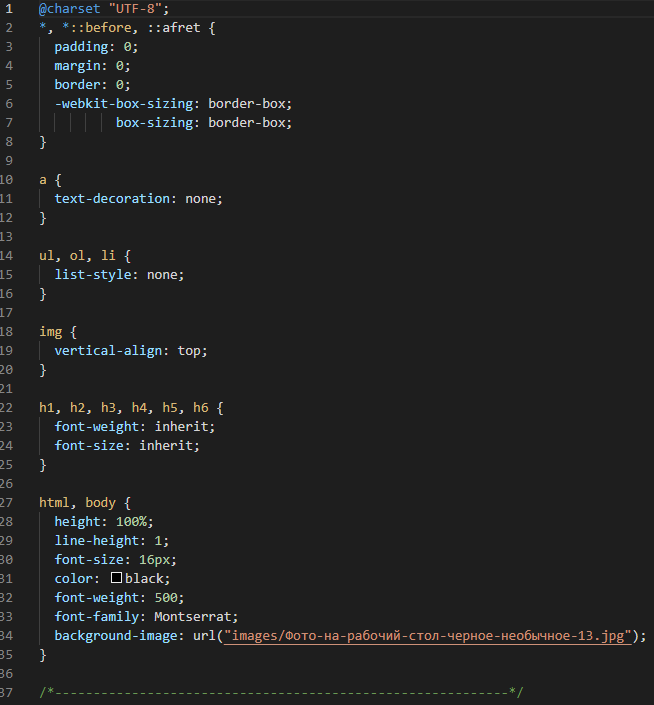
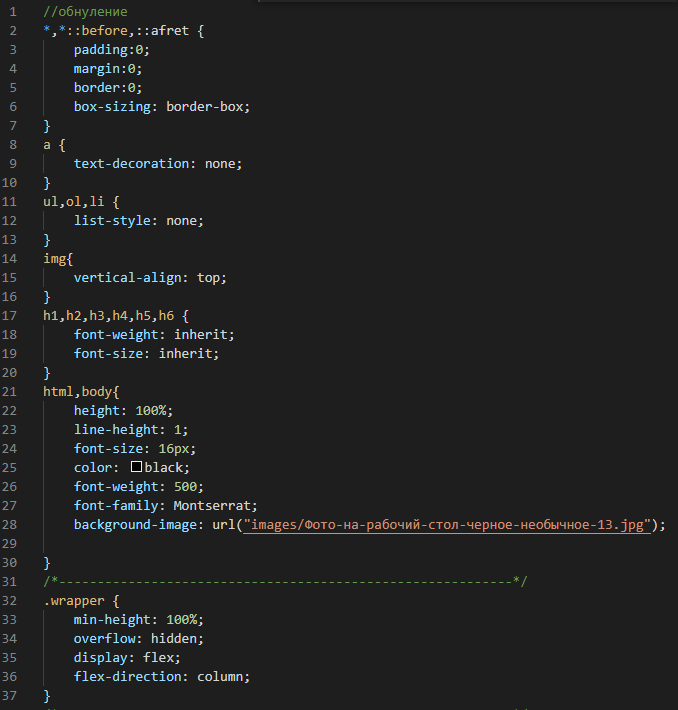
Так же, header обладает адаптивностью. При достижении фиксированных рамок ширины окна браузера элементы навигационного меню автоматически переходят друг под друга.

В теге footer расположена контактная информация комапании.

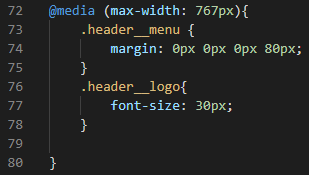
Теги header и footer, а также содержимое тега head встречаются абсолютно на всех страницах данного веб-сайта кроме форм входа и регистрации. Их содержимое не меняется.

**3.2. Добавление таблиц стилей Sass и CSS**

Все что связано со стилистическим оформлением страницы находится в папке style, например файл с расширением .css или .scss. В данной структуре используется внешнее подключение таблиц стилей через тег link, так как это самый мощный и удобный способ определения стилей и правил для веб-сайта.

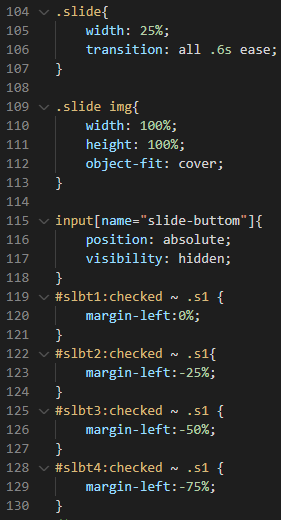
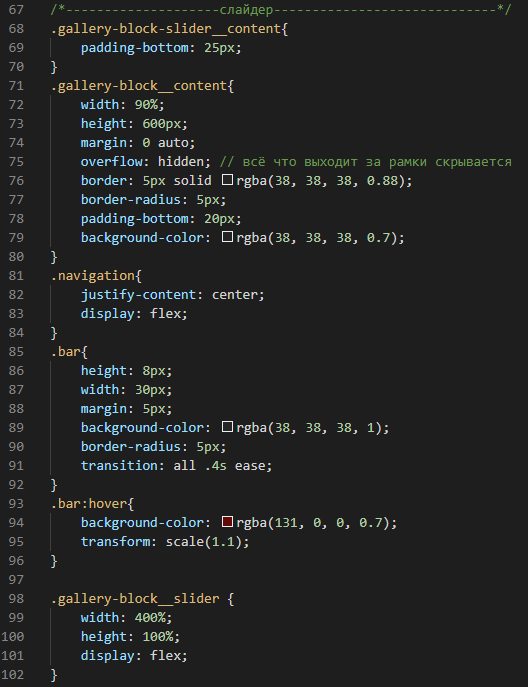


Также при разработке сайти использовались media запросы

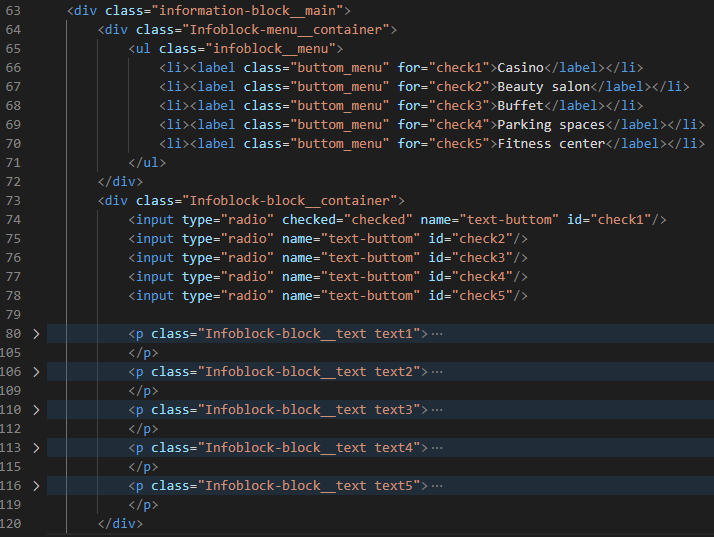


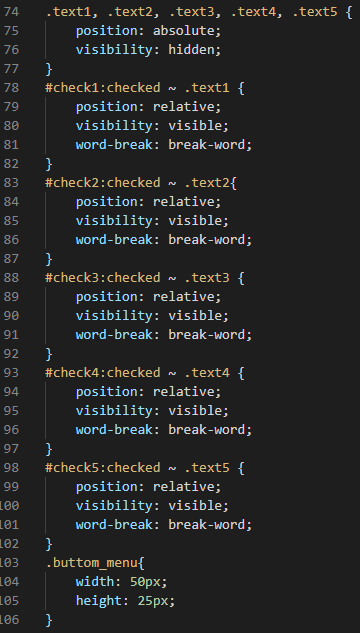
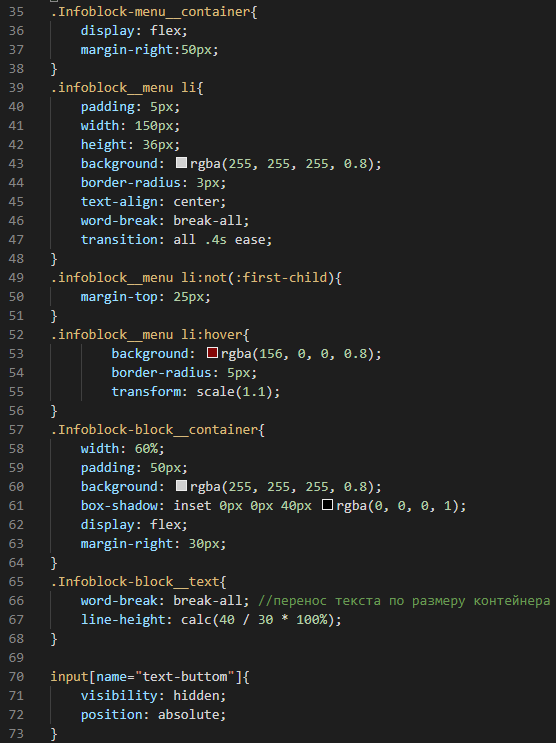
Слайдер написан на css. При создании использовались радикнопки, lable, теги тригеры.





Реализация меню на вкладке services.





**3.3. Использование стандартов XML (SVG)**

На данном веб-сайте mxl используются для логотипа в навигационной панели. При создании логотипа использовася тег svg.

**3.4. Выводы**

На данном этапе была реализована структура вебсайта на HTML, а также созданы таблицы стилей CSS/Sass. Было продемонстрировано использование стандартов XML (SVG). Было определено стилевое оформление веб-сайта, созданы анимации, а также веб-сайт был подготовлен к следующему этапу – к тестированию.

1. Тестирование веб-сайта

## **4.1. Адаптивный дизайн веб-сайта**

Адаптивная вёрстка предполагает отсутствие горизонтальной полосы прокрутки и масштабируемых областей при просмотре на любом устройстве, читабельный текст и большие области для кликабельных элементов. С помощью медиа-запросов можно управлять компоновкой и расположением блоков на странице, перестраивая шаблон таким образом, чтобы он адаптировался под разные размеры экранов устройств.

На данном этапе было принято решение все используемые медиа-запросы перенести в отдельный .css файл в папке style. Содержимое файла media.css представлено на рисунке.



Рисунок

## **4.2. Кроссбраузерность веб-сайта**

Кроссбраузерность – это способность веб-ресурса отображаться одинаково и работать во всех популярных браузерах, без перебоев в функционировании и ошибок в верстке, а также с одинаково корректной читабельностью контента.

Проверка кроссбраузерности веб-сайта осуществлялась при помощи разных браузеров (перечислить браузеры).

## **4.3. Руководство пользователя**

Данный веб-сайт предоставляет информацию потенциальному клиенту о гостинице в виде текста и графической информации и позволяет оформить аренду с веб-сайта.

Структура веб-сайта — это схема размещения его основных разделов и страниц относительно друг друга. Это план-схема, которая показывает, каким образом строится веб-сайт, логическая связка его страниц.

Такой план-схема обычно изображается графически, в виде блоков, связанных стрелочками.

На главной странице представлены слайдер, гугл карта и блок с историей отеля. Слайдер и исторический блок носят информативный характер. При нажатии на клавиши происходит смена слайдов в слайдере.

На странице “Services” присутствует меню с информации об услугах. При нажатии на кнопки происходит смена информации.

На странице “rent” реализовано меню оформления аренды номера.

## **4.4 Вывод**

На данном этапе было проанализировано поведение веб-сайта при использовании с разных устройств, а также веб-ресурс был протестирован в большом количестве браузеров. Также было добавлено руководство пользователя с прикрепленной план-схемой. По итогу выполнения данного этапа для веб-сайта был разработан адаптивный дизайн и кроссбраузерность.

**Заключение**

**Список использованных литературных источников**

<http://htmlbook.ru/html5>

http://shpargalkablog.ru/2013/08/checked.html