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

Факультет Информационных технологий

Кафедра Информационных систем и технологий

Специальность 6-05-0612-01 Программная инженерия

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА КУРСОВОГО ПРОЕКТА**

по дисциплине «Компьютерные языки разметки»

Тема «Web-сайт «Компания по металлообработке»

**Исполнитель**

студент 1 курса 9 группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.И.Милочкин

подпись, дата

**Руководитель**

ассистент \_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.Н.Николайчук

должность, учен. степень, ученое звание подпись, дата

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

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

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

дата, подпись

Курсовой проект защищен с оценкой

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.Н.Николайчук

подпись дата инициалы и фамилия

**Содержание**

[Введение 4](#_Toc166062435)

[1. Постановка задачи 5](#_Toc166062436)

[1.1. Обзор аналогичных решений 5](#_Toc166062437)

[1.2. Техническое задание 7](#_Toc166062438)

[1.3 Выбор средств реализации программного продукта. 9](#_Toc166062439)

[1.4 Выводы по разделу 9](#_Toc166062440)

[2. Проектирование страниц веб-сайта 10](#_Toc166062441)

[2.1. Выбор способа верстки 10](#_Toc166062442)

[2.2. Выбор стилевого оформления 10](#_Toc166062443)

[2.3. Выбор шрифтового оформления 10](#_Toc166062444)

[2.4. Разработка логотипа 11](#_Toc166062445)

[2.5. Разработка пользовательских элементов 11](#_Toc166062446)

[2.6. Разработка спецэффектов 12](#_Toc166062447)

[2.7. Выводы по разделу 12](#_Toc166062448)

[3. Реализация структуры веб-сайта 13](#_Toc166062449)

[3.1. Структура HTML-документа 13](#_Toc166062450)

[3.2. Добавление таблиц стилей Sass и CSS 13](#_Toc166062451)

[3.3. Использование стандартов XML (SVG) 13](#_Toc166062452)

[3.4. Выводы по разделу 14](#_Toc166062453)

[4. Тестирование веб-сайта 15](#_Toc166062454)

[4.1. Адаптивный дизайн веб-сайта 15](#_Toc166062455)

[4.2. Кроссбраузерность веб-сайта 16](#_Toc166062456)

[4.3. Руководство пользователя 17](#_Toc166062457)

[4.4. Выводы по разделу 18](#_Toc166062458)

[Заключение 19](#_Toc166062459)

[Список использованных литературных источников 20](#_Toc166062460)

[Приложения 21](#_Toc166062461)

[Приложение 1 – листинги кода 21](#_Toc166062462)

[Приложение 2 Прототипы веб-страниц 25](#_Toc166062463)

[Приложение 3 Макет структуры веб-сайта. 27](#_Toc166062464)

# **Введение**

Присутствие бизнеса в Интернете, независимо от отрасли, может оказать огромное влияние на его успех. В наши дни некоторые компании до сих пор не осознают, что большинство их клиентов посещают их веб-сайт перед совершением покупки. Наличие присутствия в Интернете веб-сайта, может иметь решающее значение для получения большего дохода.

Одна из основных причин, по которой у бизнеса должен быть интернет-ресурс — это повышение авторитета организации, а также демонстрация бренда/компании потенциальным клиентам. Он помогает четко определить, что представляет из себя компания, тем самым увеличивая вероятность выбора клиентами данной организации.

Сайт – это то, что может выделить компанию среди конкурентов. Без него это может быть невероятно сложно сделать, потому что люди не могут быстро найти качественную и достоверную информацию о компании. Как только люди найдут страницу в Интернете, они заинтересуются продуктом или услугой, предоставляемыми данной компанией, и смогут связаться с компанией благодаря информации на веб-сайте, что также даёт возможность увеличить продажи. Поскольку он работает круглосуточно и без выходных, то компания может легко публиковать обновления и объявления для своих клиентов.

Веб-ресурс даёт возможность пользоваться инструментами интернет-маркетинга: реклама, email-рассылки и т.п. В интернете удобнее и эффективнее проводить рекламные акции, скидки. Анализировать результаты также проще: следить за статистикой посещения, количеством заказов, онлайн-покупок.

Все вышеперечисленные факты отражают актуальность тематики курсовой работы.

Цель курсовой работы: разработать веб-сайт для компании по металлообработке с использованием HTML5 ,XML и JavaScript, а также с применением Sass/CSS3.

Задачи:

* проанализировать существующие языки разметки, инструменты и библиотеки для создания веб-сайта;
* разработать макет и прототип сайта;
* разработать структуру веб-сайта;
* наполнить сайт информацией по теме;
* протестировать веб-сайт;
* разработать руководство пользователя.

# **1. Постановка задачи**

**1.1. Обзор аналогичных решений**

Курсовой проект представляет из себя веб-сайт для компании по металлообработке. В бизнес-сфере, в частности в сфере металлообработки, уже существуют некоторые аналогичные решения. В этом разделе будут приведены веб-сайты существующих компаний по металлообработке.

Первым рассматриваемым веб-сайтом среди конкурентов является «west-compani.ru» [1].

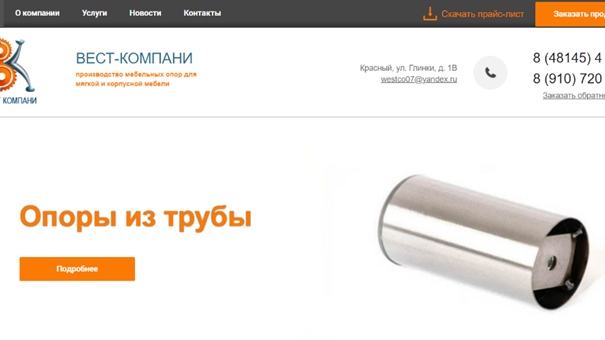
Сайт имеет строгий и современный дизайн, отражающий профессионализм компании. Цветовая палитра привлекает внимание и соответствует общему стилю отрасли. Фотографии продукции и оборудования высокого качества, что создает благоприятное впечатление о компании. Сайт демонстрирует динамичный и интерактивный подход. Пример главной страницы представлен на рисунке 1.1 ниже. 

Рисунок 1.1 – Сайт «west-compani.ru»

Также хорошо проработана мобильная версия сайта. На рисунке 1.2 ниже представлен сайт в разрешении экрана 320х769 px.



Рисунок 1.2 – Сайт «Razer.com»

Следующим рассматриваемым сайтом является «Pgkv.ru» [2]. Сайт представлен в устаревшем дизайне с запутанной навигацией, что затрудняет поиск необходимой информации. Цветовая гамма не гармонирует и вызывает дискомфорт при просмотре. Фотографии продукции представлены в низком разрешении и не корректно отображаются на странице. Сайт не обладает интерактивностью и динамикой, что делает его менее привлекательным для потенциальных клиентов. Дизайн представлен на рисунке 1.3 ниже.

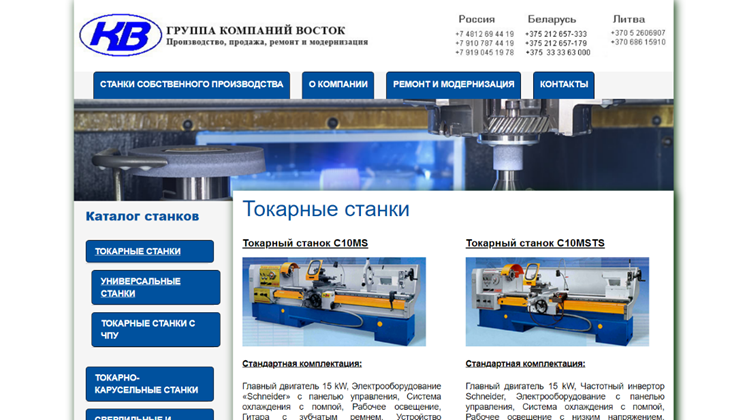


Рисунок 1.3 – Сайт «Pgkv.ru»

Как таковой мобильной версии сайта нет. Сайт просто сжимает размер блоков, не меняя их позиционирование.

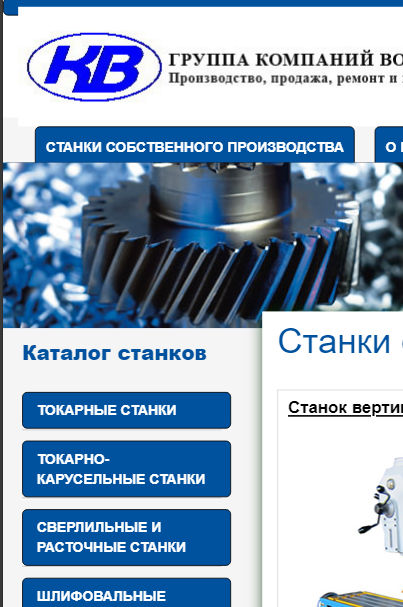


Рисунок 1.4 – Мобильная версия сайта «Pgkv.ru»

Последним рассматриваемым сайтом является «Irongroup.by» [3]. Cайт содержит каталог продукции, описание оборудования, сертификаты качества, возможности по заказу и контролю выполнения работ. На сайте размещены новости компании, блог о металлообработке, сведения о профессиональной команде и прочие разделы, подчеркивающие компетентность и надежность компании. Хорошо подобранные цвета. Удачный шрифт, который делает текст читабельным. Так-же, как и у первого сайта имеются качественные фотографии, которые привлекают внимание пользователя. Главная страница сайта представлена на рисунке 1.5 ниже.

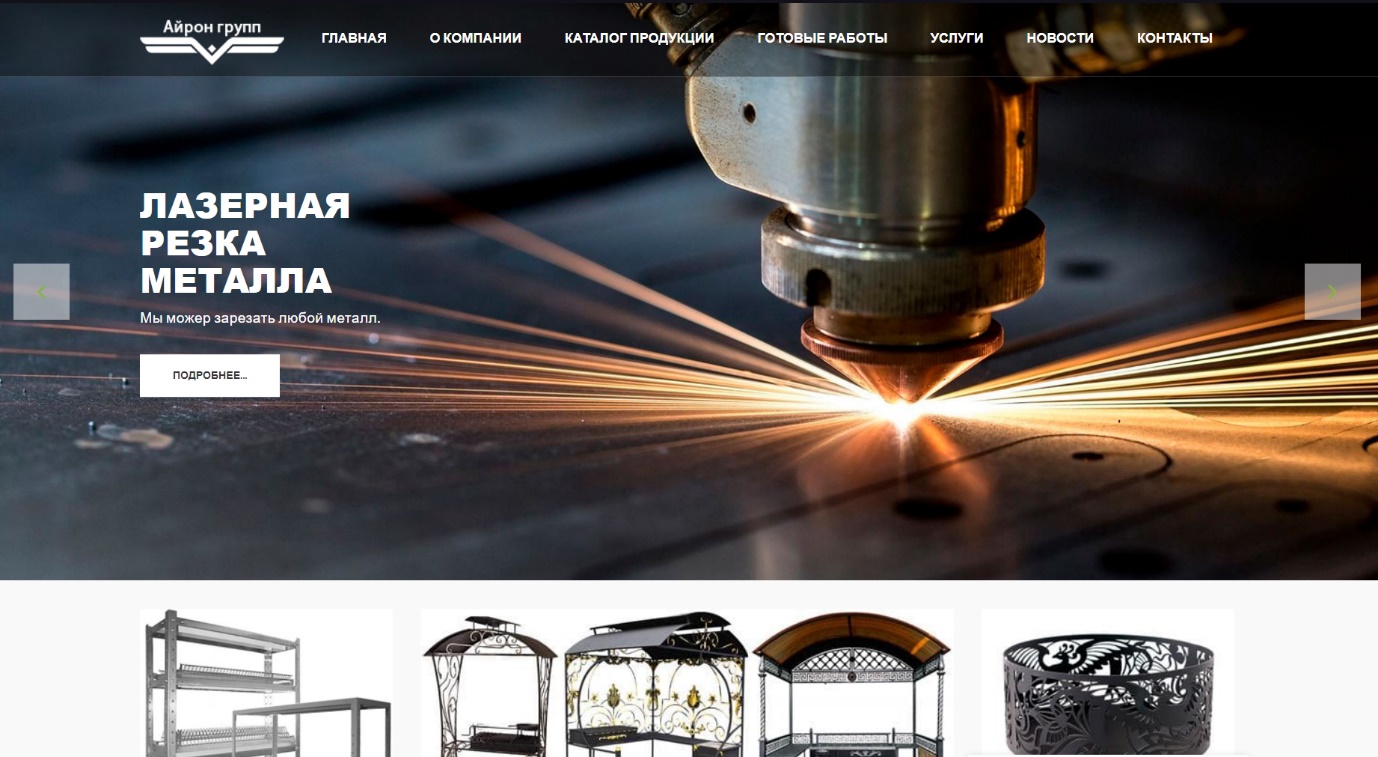


Рисунок 1.5 – Сайт «Irongroup.by»

Также хорошо проработана мобильная версия сайта, горизонтальное меню становится выпадающим. На рисунке 1.6 представлен сайт в разрешении экрана 320х769 px.



Рисунок 1.6 – Сайт «Irongroup.by»

Также можно заметить простой и понятный интерфейс сайта, в котором с легкостью разберутся люди разных возрастов.

* 1. **Техническое задание**

1. Веб-сайт «Компании по металлообработке» является визиткой для знакомства с компанией и её работами. Торговля в интернете.
2. Главной целью разрабатываемого веб-сайта является знакомство клиентов с работами компании, консультации и в долгосрочном сотрудничестве . На сайте должна быть только полезная информация. Веб-сайт должен иметь простой, но современный, дизайн, чтобы пользователю было удобно им пользоваться.
3. Используется язык гипертекстовой разметки HTML, таблицы стилей CSS/Sass, формат хранения данных Json и JavaScript, который наделяет весь сайт функционалом.
4. Прототип с пояснениями.

Было решено сделать 3 страницы: главная, для заказа и ознакомление с работами.

На главной странице будет размещена информация о видах работ, примеров деталей, достижениях компании.

На странице «Контакты» пользователь сможет оставить заявку и связаться с консультантом.

На странице «Услуги» пользователь сможет ознакомиться с уже выполненными работами и ценами на них.

В хедере будет размещен логотип и кнопки перехода на другие страницы.

Футер будет содержать информацию о компании.

1. Дизайн. (тут обоснование выбранного стиля)

Сайт должен корректно отображаться во всех основных браузерах.

Дизайн должен быть выполнен с использованием адаптивной верстки и использованием Sass. Допускается редактирование изображений в графических редакторах.

* Цветовая гамма должна включать несколько основных, хорошо контрастирующих цветов.
* Используемые изображения и логотипы должны иметь разрешение .svg для логотипов и иконок, и .jpg для остальных фотографий.

**1.3 Выбор средств реализации программного продукта.**

Для реализации использовались следующие языки: HTML, CSS/SСSS, XML.

Для управления элементами DOM и работы с данными был использован JavaScript. Структура сайта сделана с помощью языка разметки HTML, дизайн веб-сайта оформлен с помощью CSS/SCSS. В XML формате хранятся данные о коллекциях/проектах.

Курсовая работа выполняется в редакторе кода Visual Studio Code.

**1.4 Выводы по разделу**

Были исследованы аналоги веб-сайта, что позволило выявить ошибки и под-черкнуть успешные приемы. Исходя из обзора аналогов, было принято решение со-здать веб-сайт, который сможет удивить потенциального покупателе. Создание адаптивности также является одной из главных целей при создании веб-страницы. С помощью этой информации будет разработан проект, сочетающий лучшие стороны подобных сервисов.

**2. Проектирование страниц веб-сайта**

**2.1. Выбор способа верстки**

Для расположения элементов на странице применяется flex-верстка. Она предлагает удобные инструменты для создания адаптивного и гибкого дизайна веб-страницы, позволяет контролировать расположение элементов на веб-странице, а также их возможность подстраиваться под изменения ширины экрана. Также, flex-верстка позволяет легко настраивать элементы, что делает ее одним из самых оптимальных решений для осуществления поставленной задачи. Все современные браузеры поддерживают flexbox. В соответствии с выбором способа верстки были разработаны макеты страниц веб-сайта.

**2.2. Выбор стилевого оформления**

В процессе разработки веб-сайта было принято решение о применении минималистичного дизайна, чтобы создать простой и сдержанный внешний вид, так как пользователю так будет удобнее пользоваться сайтом, проще воспринимать информацию и не потеряться. Сайт должен привлечь внимание своей уникальностью, современностью.

Цветовые решения. Можно выделить два ключевых цвета интерфейса: голубой, светло-серый. Пример использования цветов предоставлен в приложении 2.

**2.3. Выбор шрифтового оформления**

В курсовом проекте будет использован один основной шрифт: Proxima Nova Condensed. Он отлично подчеркивает уникальность и футуристичность.



- Proxima Nova Condensed

**2.4. Разработка логотипа**

Было решено использовать в качестве логотипа стилизованное название компании «IronCity». Шрифт Proxima Nova Condensed, цвет сине-голубой.



Рисунок 2.1 – Логотип

**2.5. Разработка пользовательских элементов**

В хедере сайта будет представлены логотип и горизонтальная навигационная панель (рисунок 2.2).



Рисунок 2.2 – Хедер

В футере будет находится фраза “Copyright Ironcity. Made by Ilya Milochkin” (Рисунок 2.3).

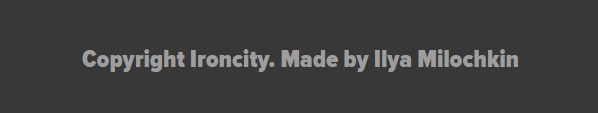


Рисунок 2.3 – Футер

**2.6. Разработка спецэффектов**

Спецэффекты важная часть любого сайта – они делают сайт более удобным для пользователя, оптимизируют взаимодействия с веб-страницей.

Будет осуществлять анимация навигационной панели: название страницы, при наведении текст меняет цвет, при нажатии так же.

Так же будет реализовано PopUp окно “оставить заявку”



Рисунок 2.4- «До»



Рисунок 2.5- «После»

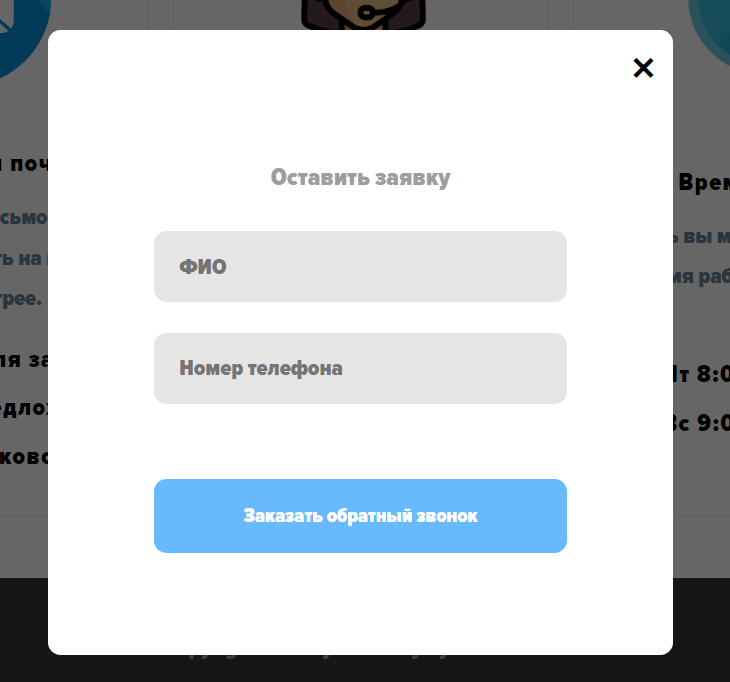


Рисунок 2.6 - PopUp окно

**2.7. Выводы по разделу**

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

**3. Реализация структуры веб-сайта**

**3.1. Структура HTML-документа**

Структура HTML-документа определяет базовый «скелет» для будущего веб-сайта. Код выполнен по стандарту HTML5.

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

В теге header находится горизонтальная навигационная панель. Она содержит ссылки на все основные разделы сайт, логотип и название компании.

При необходимости создания списка использовались теги ul, которые состоят из некоторого числа элементов li, каждый из которых содержит в себе некоторую информацию. Для вставки и оформления изображений был использован тег img, атрибут src которого содержит в себе ссылку на изображение для дальнейшего его внедрения. Для оформления основной текстовой информации был применен тег p, задача которого – содержать некоторую информацию, размещение которой требует выделенного пространства.

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

При разработке веб-сайта ключевое место занимают стилевые решения, которые определяют внешний вид каждого элемента на веб-страницах. Для оформления основной части файла использовался язык стилей CSS.

|  |
| --- |
| <head>  <link rel="stylesheet" href="css/reset.css">  <link rel="stylesheet" href="css/styles.css">  <link rel="stylesheet" href="css/input.css">  </head> |

Листинг 3.2 – пример внешнего подключение CSS

Для оформления хедера использовался Sass. Он (SCSS) облегчает применение свойств благодаря миксинам. Они позволяют один раз создать набор правил, чтобы потом использовать их многократно или смешивать с другими правилами.

|  |
| --- |
| .header {  padding: 20px 0;  position: fixed;  background-color: #FFF;  top: 0;  left: 0;  width: 100%;  z-index: 52;  box-shadow: 0px 0px 8px;    &\_nav {  li + li {  margin-left: 40px;  }    ul {  display: flex;    a {  font-size: 20px;  color: #676770;  transition: color 0.2s;    &:hover {  color: #0082F3;  }    &:active {  color: #0062b6;  }  }  }  }  &\_container {  display: flex;  justify-content: space-between;  align-items: center;  }  .logo {  font-size: 48px;  text-transform: uppercase;  letter-spacing: 4px;  color: #69B9FF;  }  } |

Листинг 3.3 – Применение Sass для header

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

Для создания некоторых картинок было принято использование SVG-формата. Этот формат позволяет картинкам сохранять свое качество, несмотря на размер экрана.

|  |
| --- |
| <svg width="1920" height="962" viewBox="0 0 1920 962" fill="none" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">  <rect width="1920" height="962" fill="url(#pattern0\_903\_2)"/>  <defs>  <pattern id="pattern0\_903\_2" patternContentUnits="objectBoundingBox" width="1" height="1">  <use xlink:href="#image0\_903\_2" transform="matrix(0.000652398 0 0 0.00130208 -0.168056 0)"/>  </pattern>  <image id="image0\_903\_2" width="2048" height="768" xlink:href="data:image/jpeg;base64,  </defs>  </svg> |

Листинг 3.4 – Пример использования стандартов SVG

Для соблюдения стандартов оформления веб-сайтов был разработан файл XML (Extensible Markup Language), основной задачей которого является хранение текстовой информации для упрощенного доступа без внедрения в HTML-документ, для которого предназначен данный XML-файл. Его основным достоинством также является возможность создания пользовательских тегов, что упрощает чтение файлов разработчиком, как и создание алгоритмов для получения данных. В проекте был создан файл name.xml, информация которого предназначена для веб-страницы index.html. В нем были использованы следующие теги:

1.Outline – Информация о разработчике в футере

Путем обработки данных тегов можно получить информацию как о разработчике сайта.

3.4 Использование JavaScript

JavaScript (JS) является одним из самых распространенных языков программирования для веб-разработки. Этот язык позволяет добавлять динамичность и интерактивность на веб-страницы, что делает пользовательский опыт более увлекательным и позволяет создавать более сложные веб-приложения.

|  |
| --- |
| const openPopUp = document.getElementById('open\_pop\_up');  const closePopUp = document.getElementById('pop\_up\_close');  const popUp = document.getElementById('pop\_up')  openPopUp.addEventListener('click', function(e){  e.preventDefault();  popUp.classList.add('active');  formContent.style.display = 'block';  thankYouMessage.style.display = 'none';  })  closePopUp.addEventListener('click', () => {  popUp.classList.remove('active');      })  const requestForm = document.getElementById('requestForm');  const formContent = document.getElementById('formContent');  const thankYouMessage = document.getElementById('thankYouMessage');  requestForm.addEventListener('submit', function(e) {  e.preventDefault();  formContent.style.display = 'none';  thankYouMessage.style.display = 'block';  setTimeout(function() {  popUp.classList.remove('active');  }, 3000);  }); |

Листинг 3.5 – Пример использования JS для работы PopUp окна

JavaScript может использоваться для создания различных интерактивных элементов на странице, таких как выпадающие меню, слайдеры, анимации, формы обратной связи и многое другое. С помощью JS можно изменять содержимое, стиль и расположение элементов на странице, а также обрабатывать действия пользователя, такие как клики и ввод текста.

**3.4. Выводы по разделу**

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

**4. Тестирование веб-сайта**

**4.1. Адаптивный дизайн веб-сайта**

Для достижения адаптивности в проекте применялись методы верстки grid и flex, а также использовались медиа-запросы. Ресурс должен одинаково хорошо демонстрироваться на экране компьютера (рис. 4.1), планшете(рис. 4.2), смартфоне(рис. 4.3).



Рисунок 4.1 – Вид страницы для компьютера

Медиа- запросы использовались для перемещения элементов основной части страницы при размерах мобильной версии.



Рисунок 4.2 – Вид страницы для планшетных устройств

Адаптивность является очень важным аспектом веб-разработки, так как позволяет сайту корректно отображаться на различных устройствах с разными размерами экранов. С увеличением числа устройств с различными размерами экранов, таких как смартфоны, планшеты, ноутбуки и настольные компьютеры, становится критически важным, чтобы сайт был доступен и удобочитаем на всех устройствах.

Адаптивность позволяет автоматически изменять расположение и размер элементов страницы в зависимости от размера экрана устройства, на котором происходит просмотр сайта. Таким образом, пользователь получает оптимальное визуальное восприятие сайта на любом устройстве и не испытывает неудобств при просмотре страницы.

Кроме того, адаптивность сайта имеет важное значение для улучшения SEO-оптимизации, так как поисковые системы высоко оценивают мобильную доступность сайтов и ранжируют их в соответствии с этим фактором. Адаптивный сайт также может увеличить время пребывания пользователей на сайте и увеличить вероятность выполнения целевых действий (например, покупки товара или заполнения формы заказа).



Рисунок 4.3 – Вид страницы для мобильных устройств

Для этого создается адаптивный дизайн и используется адаптивная верстка. В итоге сайт отлично позиционируется на всех основных типах устройств, что гарантирует полноценный охват аудитории.

**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.

Разные пользователи используют разные браузеры, что осложняет работу верстальщикам, так как не все свойства CSS используются одинаково во всех браузерах. Для этого и была придумана кроссбраузерность — способность веб-ресурса отображать одинаково и работать во всех популярных браузерах, без всяких проблем.

После написания основной структуры страницы на HTML и внешнего стилевого оформления появился вопрос о тестировании. Веб-сайт был открыт при помощи различных браузеров. В ходе успешного тестирования было выяснено, что сайт поддерживается в таких браузерах как Microsoft Edge, Opera и Google Chrome.

На рисунке 4.2 ниже представлен сайт, открытый в браузере Microsoft Edge.

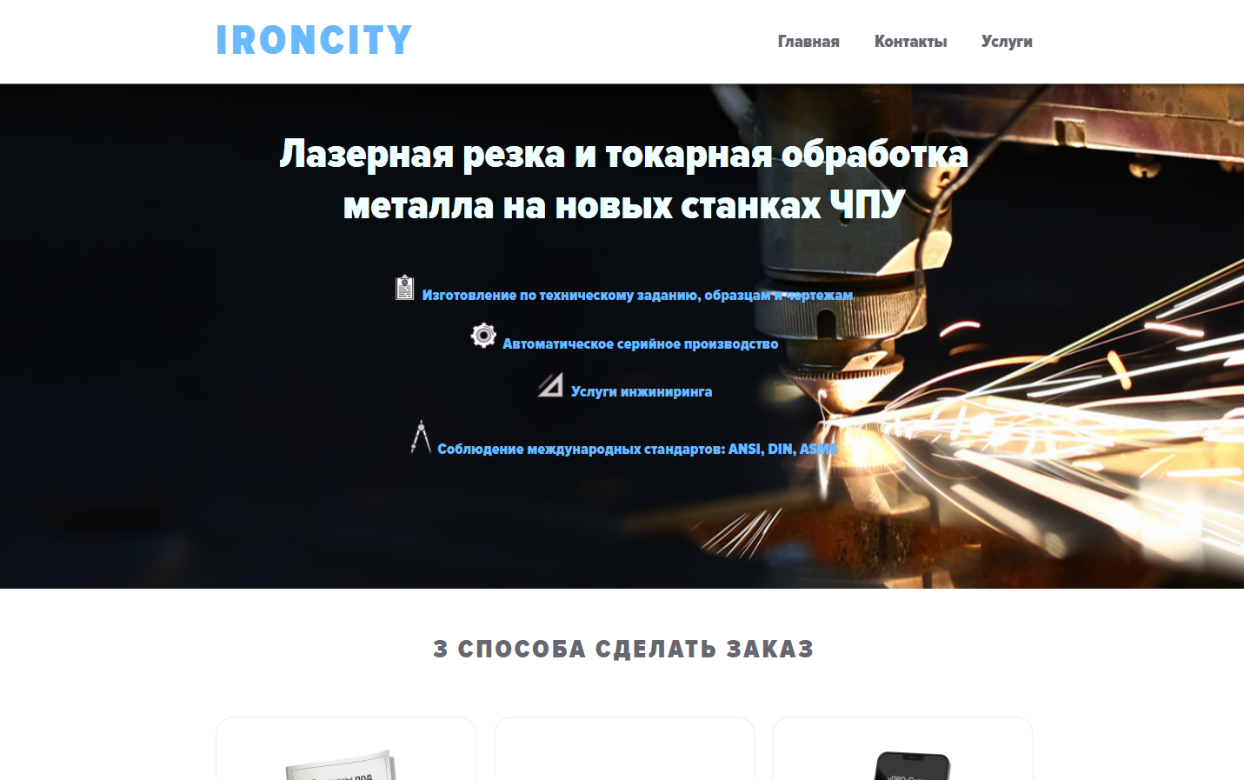


Рисунок 4.2 – Сайт, открытый в браузере Microsoft Edge

Также, на рисунке 4.3 ниже представлен сайт, открытый в браузере Google Chrome.

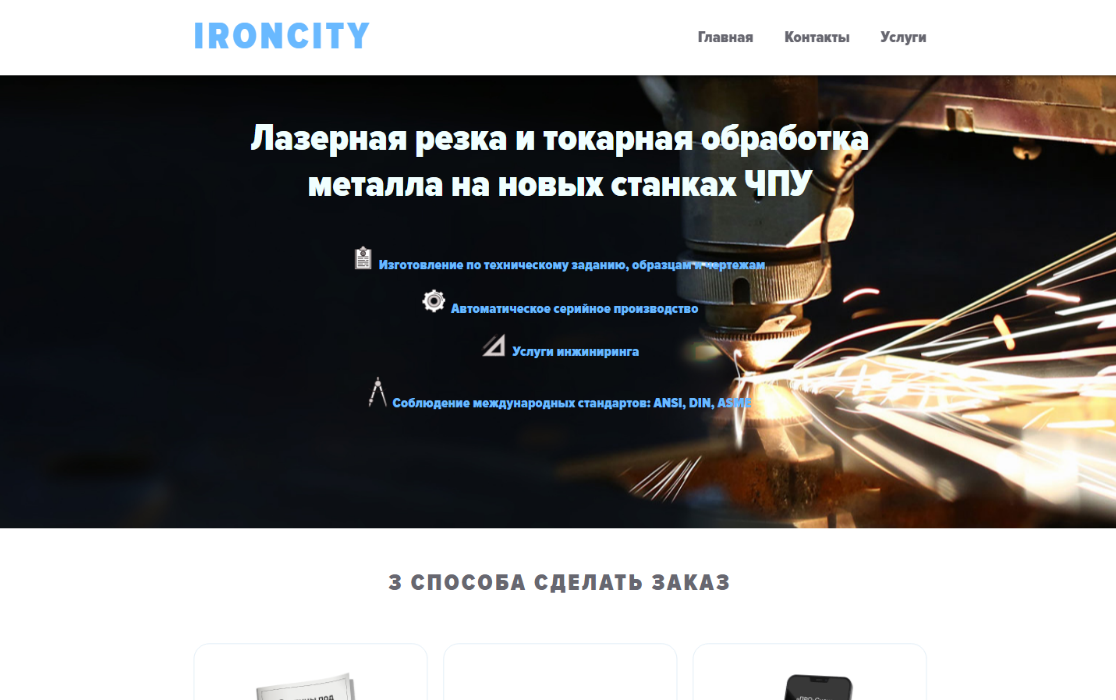


Рисунок 4.3 – Сайт, открытый в браузере Google Chrome

Как можно заметить – сайты не отличаются друг от друга независимо от используемого браузера. Шрифты, цвета, фотографии и отступы сохраняют свои свойства и не «слетают».

**4.3. Руководство пользователя**

Пользователю представляется краткая техническая информация о товаре, количество единиц товара в упаковке, размеры, конечно же, его цена. После успешного определения нужной ему товарной единицы пользователь может оформить заказ. Для удобного пользования сайтом реализовано:

* разработан удобный и приятный пользовательский интерфейс, который будет понятен любому человека.
* веб-страница оснащена гиперссылками для быстрого перехода между страницами (разделами).

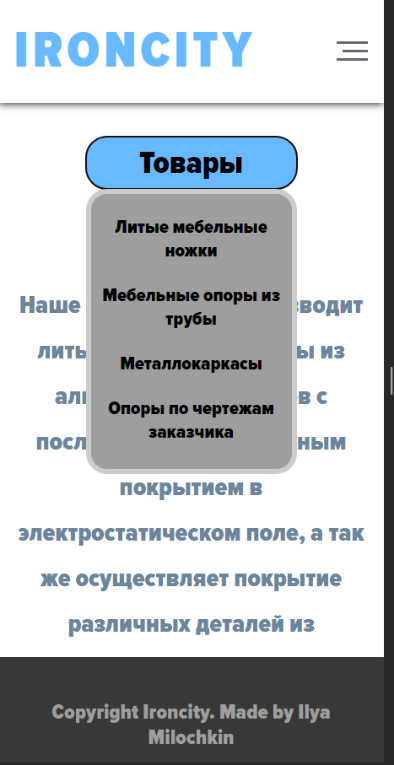


Рисунок 4.4 – Меню выбора товаров



Рисунок 4.5 – Карточка товара

Для получения информации о продукте созданы карточки товара, продемонстрированные на рисунке 4.5, с минимально необходимым набором информации о конкретном товаре.

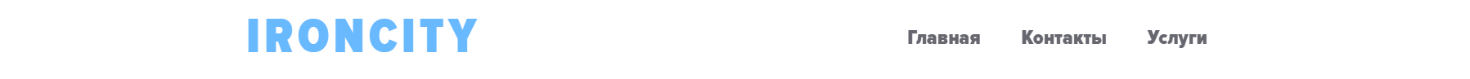


Рисунок 4.6 – Меню сайта (Desktop)

На главной странице имеется понятное меню, которое перенаправляет пользователя в нужный ему раздел, продемонстрированное на рисунке 4.6.

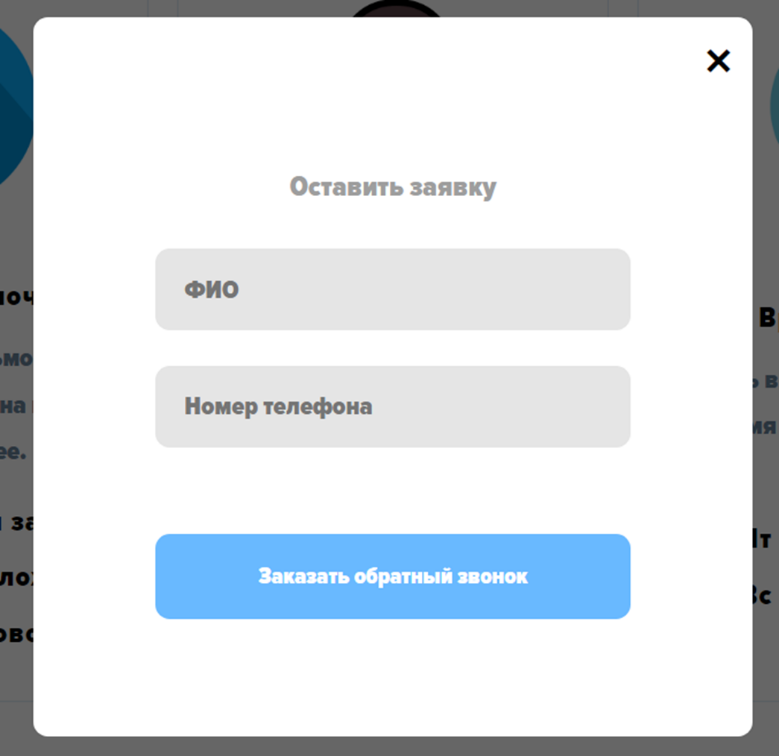


Рисунок 4.7 PopUp окно

На рисунке 4.7 представлено PopUp окно для отправки заявки оператору колл-центра

4.4 Тестирование кода

Тестирование кода веб-страниц является критически важной частью процесса веб-разработки. Это процесс проверки программного кода, чтобы убедиться, что он работает так, как должен работать, и что он соответствует определенным стандартам и требованиям.

Тестирование позволяет убедиться, что код работает правильно и соответствует заданным требованиям. Это помогает обеспечить высокое качество и надежность сайта. Тестирование помогает обнаружить уязвимости и потенциальные угрозы безопасности, что позволяет разработчикам принимать меры для их устранения до того, как они могут быть использованы злоумышленниками. Ошибка в коде, которая обнаруживается на ранней стадии, может быть исправлена быстрее и с меньшими затратами, чем если бы она была обнаружена позже в процессе разработки или после выпуска сайта. Правильно работающий сайт, который не вызывает ошибок и проблем, улучшает опыт пользователя, что может привести к большей лояльности и повторному использованию сайта.

Весь код был протестирован с помощью онлайн-сервиса The W3C Markup Validation Service. Результат представлен на рисунке ниже.



Рисунок 4.8 – Результат тестирования кода с помощью The W3C Markup Validation Service.

В целом, тестирование кода веб-страниц является неотъемлемой частью процесса веб-разработки и помогает гарантировать высокое качество, безопасность и удобство использования сайта.

**4.5. Выводы по разделу**

В данном разделе было рассмотрено, как сайт ведет в себя на разных устройствах и в разных браузерах. И ходе этого тестирования выявил некоторые ошибки и исправил их. Так же в этом разделе есть руководство пользователя, где кратко указана информация о функциональности сайта.

**Заключение**

Цель проекта - разработать веб-сайт для компании по металлообработке, используя HTML5, XML, JavaScript, Sass/CSS3. Основная задача - создать информативный и удобный сайт для знакомства клиентов с работами компании. Акцент сделан на модернизацию и авторитетность компании через онлайн-присутствие.

**Ключевые этапы разработки:**

1. **Проектирование страниц:**
   * Flexbox для удобной верстки;
   * Минималистичный дизайн с цветовой гаммой и шрифтами;
   * Разработка логотипа и пользовательских элементов;
   * Добавление спецэффектов для интерактивности.
2. **Реализация структуры сайта:**
   * HTML5 для базовой структуры документа;
   * Стили CSS/Sass для оформления;
   * Использование XML (SVG) для стандартов оформления;
   * Тестирование и подготовка к адаптивному дизайну.
3. **Тестирование веб-сайта:**
   * Реализация адаптивного дизайна для различных устройств;
   * Кроссбраузерность веб-сайта для гарантированного исполнения в различных браузерах;
   * Создание руководства пользователя для удобного пользования сайтом.

Курсовой проект успешно реализован с учетом современных технологий веб-разработки, что позволило создать функциональный и привлекательный сайт для компании по металлообработке.

Список использованных литературных источников

1. Документация по Figma [Электронный ресурс]. — Режим доступа: https://help.figma.com
2. Аналог «West-Compani.ru» [Электронный ресурс]. — Режим доступа: https://west-compani.ru
3. Аналог «pgkv.ru» [Электронный ресурс]. — Режим доступа: https://pgkv.ru
4. Аналог «IronGroup» [Электронный ресурс]. — Режим доступа: https://www.irongroup.by
5. Документация по Sass/SCSS [Электронный ресурс]. — Режим доступа: https://sass-scss.ru/guide/
6. Документация по JavaScript/HTML [Электронный ресурс]. — Режим доступа: https://developer.mozilla.org
7. Введение в кросс-браузерное тестирование [Электронный ресурс]. — Режим доступа:

https://developer.mozilla.org/ru/docs/Learn/Tools\_and\_testing/Cross\_browser\_testing/Introduction

1. Репозиторий проекта на GitHub [Электронный ресурс]. — Режим доступа: https://github.com/maulichkk