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 «Веб-сайт «магазина мелкой бытовой техники»

**Исполнитель**

студент 1 курса 9 группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_ К.В. Хуторцов

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](#_Toc166321381)

[1 Постановка задачи 5](#_Toc166321382)

[1.1 Обзор аналогичных решений 5](#_Toc166321383)

[1.1.1 Аналог «5 ЭЛЕМЕНТ» 5](#_Toc166321384)

[1.1.2 Аналог «ЭЛЕКТРОsila.by» 6](#_Toc166321385)

[1.1.3 Аналог «DNS» 7](#_Toc166321386)

[1.2 Техническое задание 8](#_Toc166321387)

[1.3 Выбор средств реализации программного продукта 9](#_Toc166321388)

[1.4 Вывод 9](#_Toc166321389)

[2 Проектирование страниц веб-сайта 10](#_Toc166321390)

[2.1 Выбор способа верстки 10](#_Toc166321391)

[2.2 Выбор стилевого оформления 10](#_Toc166321392)

[2.3 Выбор шрифтового оформления 11](#_Toc166321393)

[2.4 Разработка логотипа 11](#_Toc166321394)

[2.5 Разработка пользовательских элементов 12](#_Toc166321395)

[2.6 Разработка спецэффектов 15](#_Toc166321396)

[2.7 Выводы 17](#_Toc166321397)

[3 Реализация структуры веб-сайта 18](#_Toc166321398)

[3.1 Структура HTML-документа 18](#_Toc166321399)

[3.2 Добавление таблиц стилей Scss и CSS 19](#_Toc166321400)

[3.3 Использование стандартов XML (SVG) 20](#_Toc166321401)

[3.4 Использование JavaScript 20](#_Toc166321402)

[3.5 Выводы 21](#_Toc166321403)

[4 Тестирование веб-сайта 22](#_Toc166321404)

[4.1 Адаптивный дизайн веб-сайта 22](#_Toc166321405)

[4.2 Кроссбраузерность веб-сайта 24](#_Toc166321406)

[4.3 Руководство пользователя 25](#_Toc166321407)

[4.4 Тестирование кода 27](#_Toc166321408)

[4.5 Выводы 28](#_Toc166321409)

[Заключение 29](#_Toc166321410)

[Список использованных источников 30](#_Toc166321411)

Введение

Сайт – обязательная часть продвижения и развития компании, визитка и официальное лицо, который сочетает в себе современный и модный подход к ведению бизнеса. Также это место, где можно найти подробную информацию о компании, услугах, выполнении услуг и контактах. Информацию на нём можно легко и быстро публиковать и изменять, что является преимуществом перед различными видами традиционной рекламы. Это значит, что сайт является ещё и хорошим рекламным инструментом, инструментом привлечения новых клиентов. С помощью каталогов и поисковиков Интернет-пользователь может ознакомится с предлагаемыми услугами и стать потенциальным клиентом. Что и подчёркивает актуальность курсового проекта.

Цели курсового проекта:

* разработка структуры web-сайта для мелкой бытовой техники с использованием современных средств проектирования и разработки;
* применение навыков, полученных в результате прохождения курса «Компьютерные языки разметки».

Задачи курсового проекта:

* 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.

1 Постановка задачи

1.1 Обзор аналогичных решений

Перед тем как приступать к разработке своей системы рекомендуется изучить уже готовые решения и посмотреть, что по мнению других разработчиков является наиболее необходимым для пользователей, чтобы приложение было полезным и удобным в использовании.

1.1.1 Аналог «5 ЭЛЕМЕНТ»

На рисунке 1.1 представлен внешний вид главного каталога сайта [1].



Рисунок 1.1 – Окно главного каталога сайта «5 ЭЛЕМЕНТ»

Сайт выполнен в приятном минималистичном дизайне, что позволяет обратить внимание пользователя именно на продукт сайта. Цветовая гамма выбрана с акцентом на белый с частями красного. На главной странице можно выбрать категории товаров и с помощью слайдеров посмотреть актуальные акции.

Данный сайт специализируется на продаже любой техники. Из данного сайта можно вынести основные разделы с различными видами техники, которые необходимы покупателю для выбора подходящего каталога товаров.

Далее необходимо рассмотреть подробнее один из разделов, для того чтобы определить наиболее важные данные, которые необходимо предоставить пользователю далее. Данная страница представлена на рисунке 1.2.

На рисунке представлены страница для выбора смартфона, в которой можно увидеть предложенные категории в виде названия разных моделей смартфонов, фильтр для выбора подходящей модели и выбор количества моделей показываемых на странице пользователя для улучшения удобства выбора.

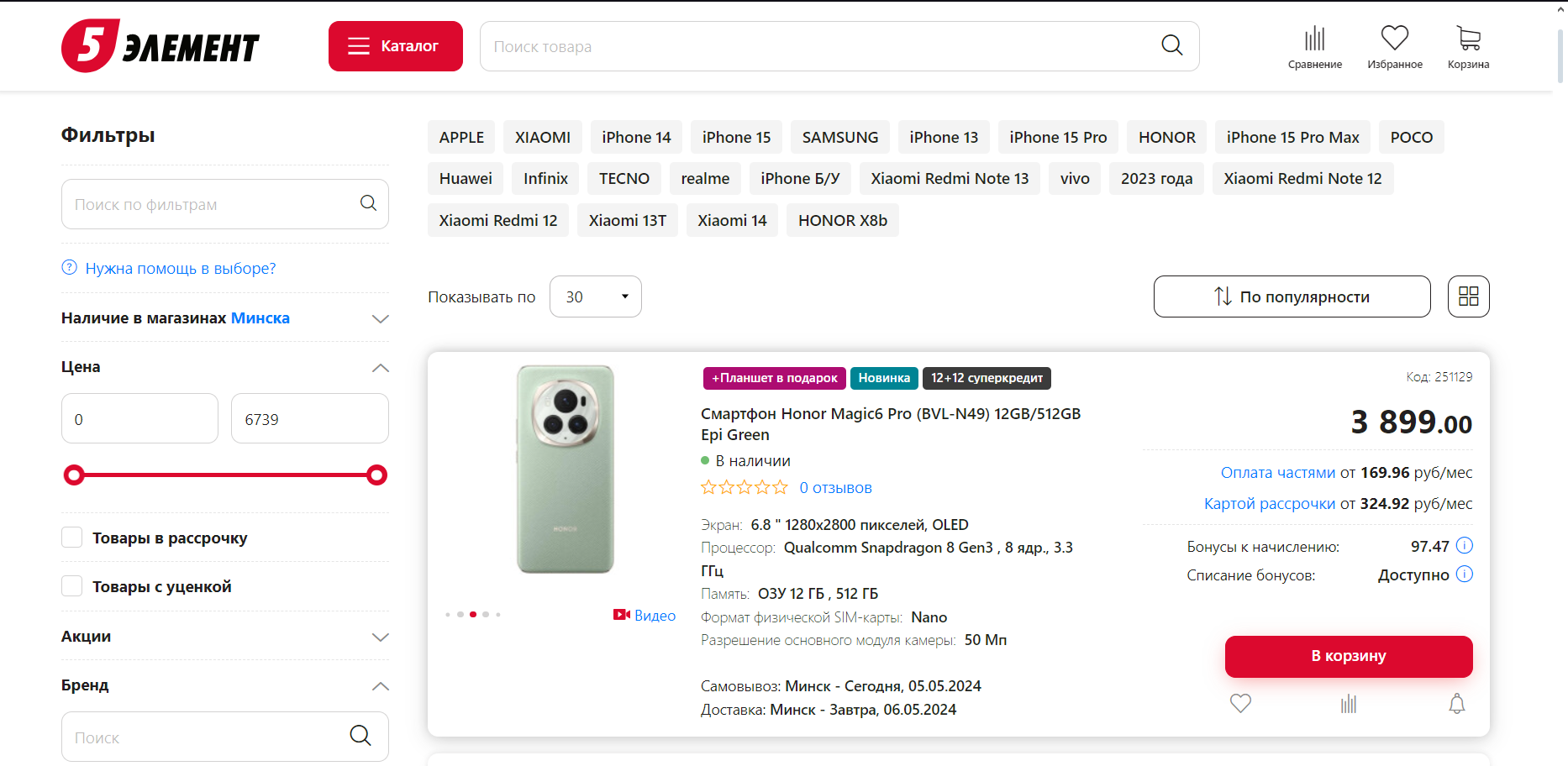


Рисунок 1.2 – Окно выбранного каталога с товарами «5 ЭЛЕМЕНТ»

Минусов у данного сайта я не обнаружил.

1.1.2 Аналог «ЭЛЕКТРОsila.by»

На рисунке 1.3 представлен внешний вид главного каталога сайта [2].

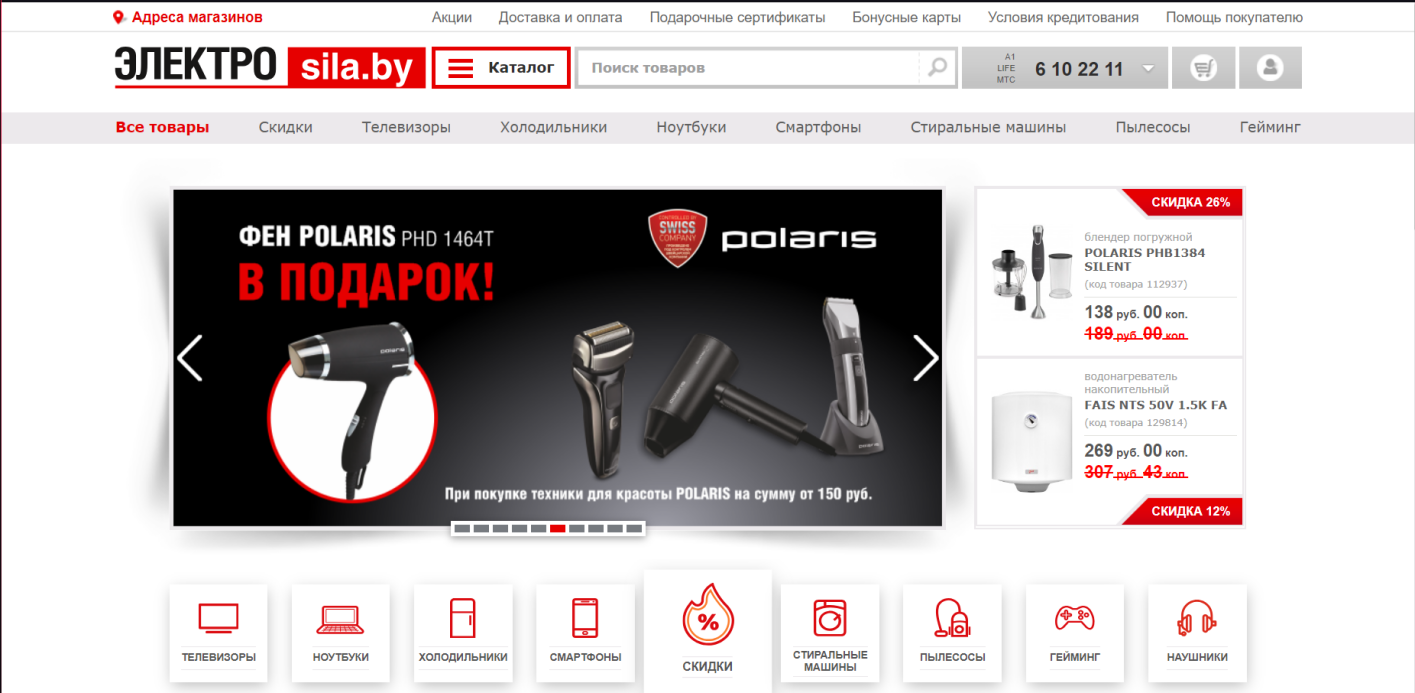


Рисунок 1.3 – Окно каталога сайта «ЭЛЕКТРОsila.by»

Дизайн данного сайт мне нравится меньше. Потому что дизайн данного сайта очень похож на предыдущий, но выглядит при этом крайне устаревшим.

Цветовая гамма точно также, как и в предыдущем сайте, выбрана с акцентом на белый с частями красного.

Далее рассматривается персональная страница товара, она представлена на рисунке 1.4.

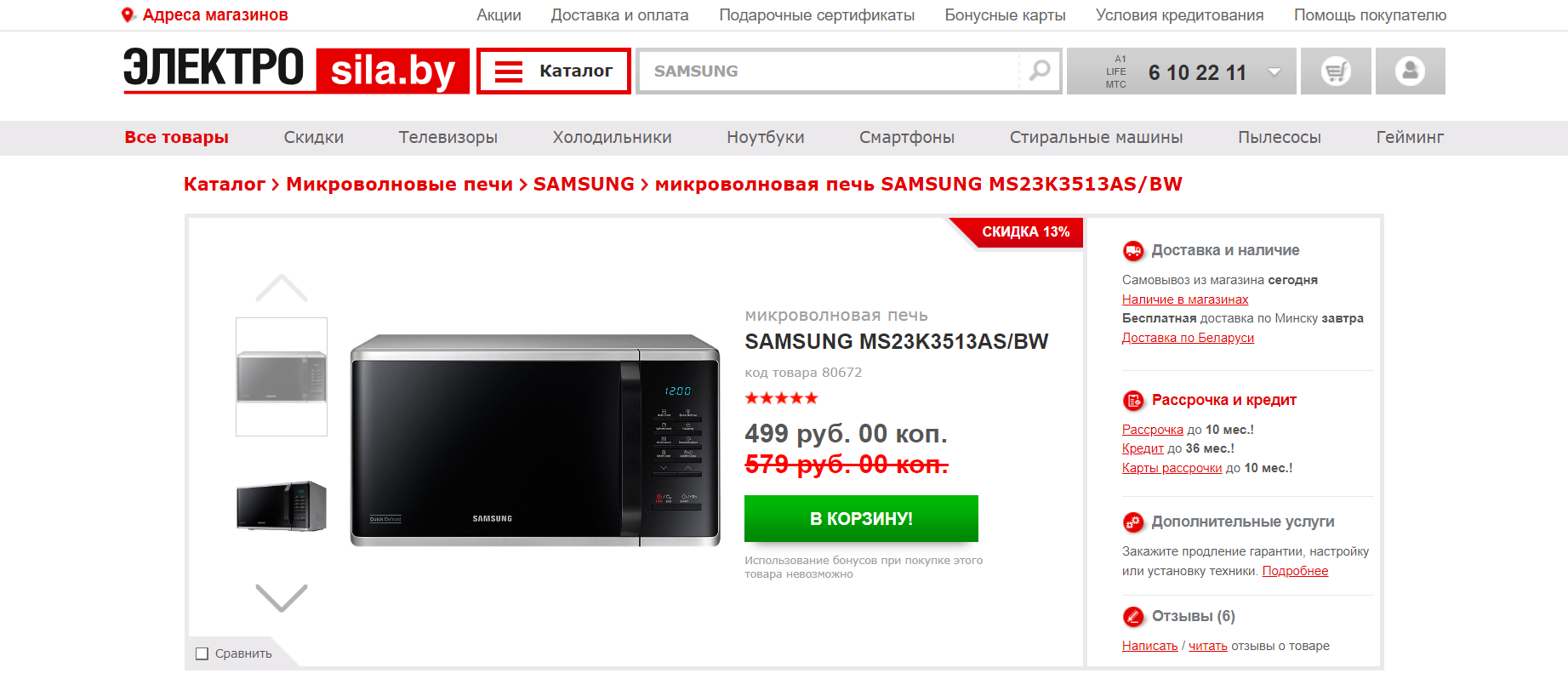


Рисунок 1.4 – Окно товара сайта «ЭЛЕКТРОsila.by»

Данная страница является удобной и функциональной, но всё такой же устаревший дизайн, как и на главной странице, сильно бросается в глаза.

1.1.3 Аналог «DNS»

Главная страница сайта [3] представлена на рисунке 1.5.

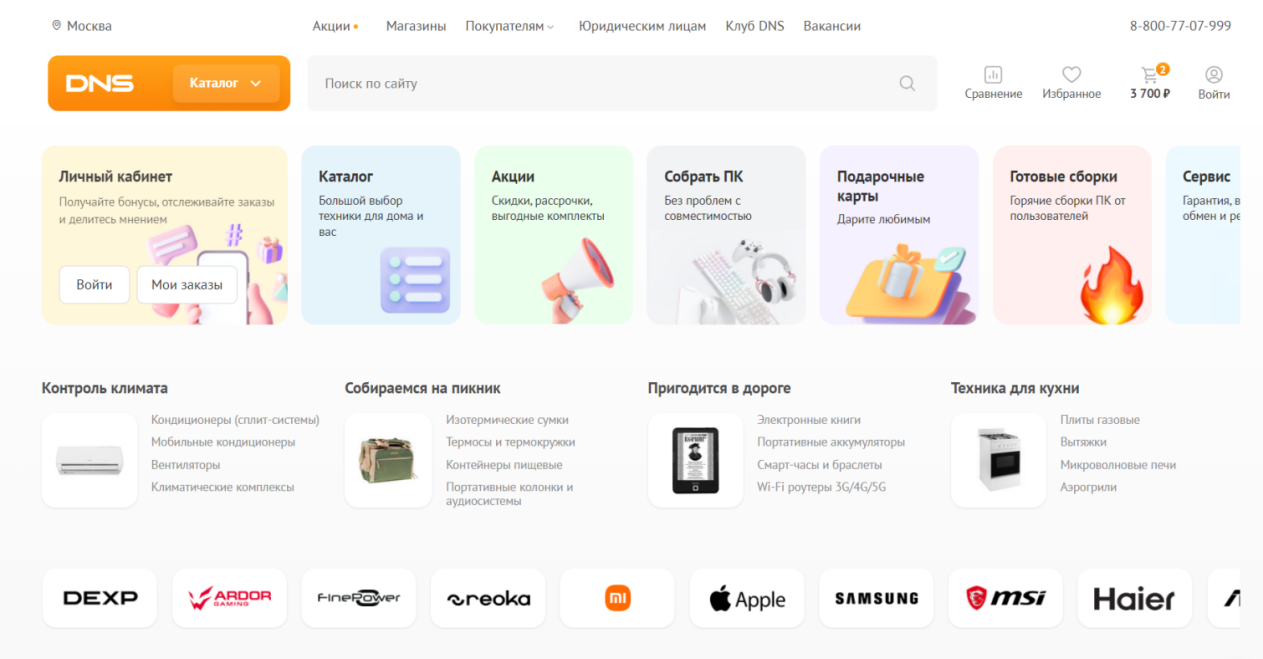


Рисунок 1.5 – Окно главного каталога сайта «DNS»

Общий вид сайта создает хорошее впечатление. На главной странице представлены разделы товаров, облегчая поиск товара. Цветовая гамма выбрана с акцентом на белый с частями оранжевого.

Далее рассматривается выбранный каталог товаров, он представлена на рисунке 1.6.

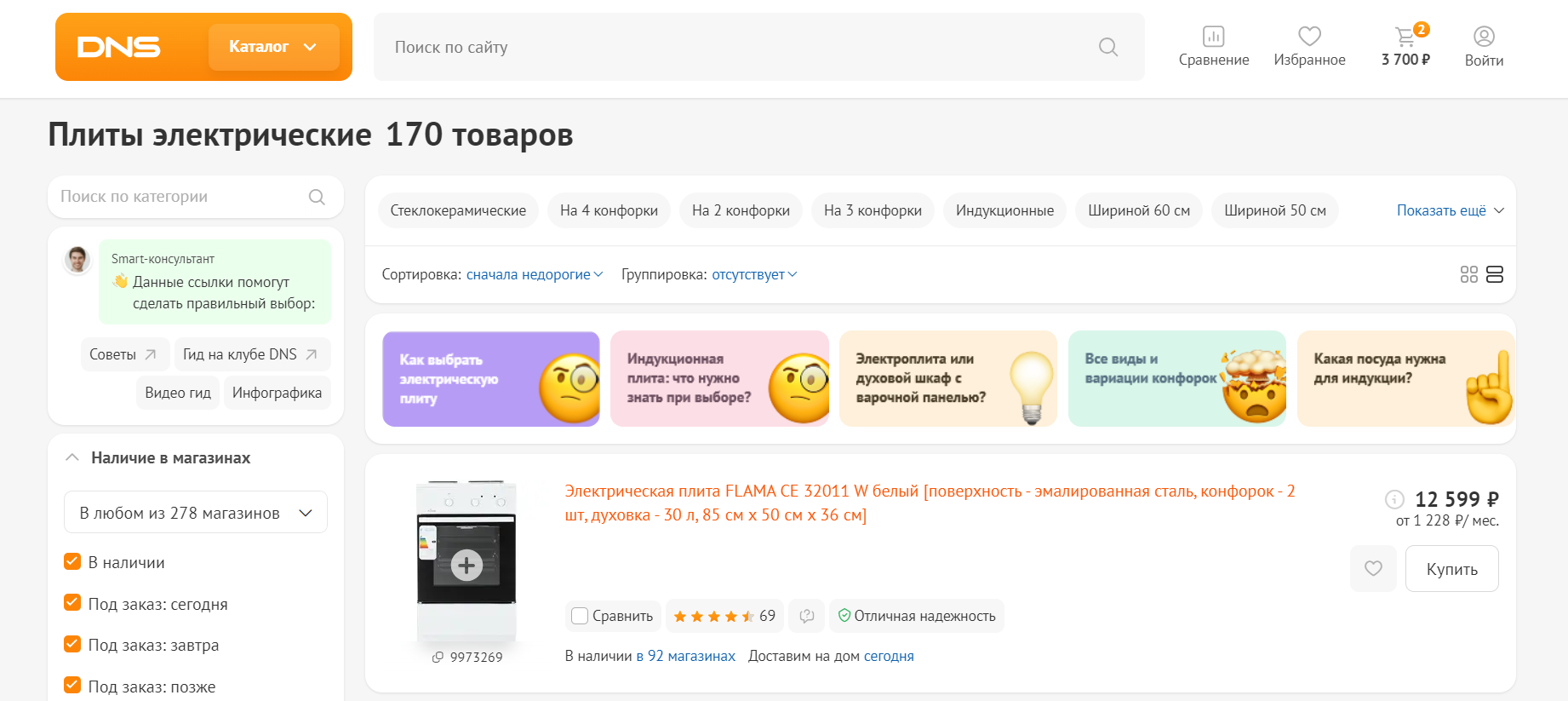


Рисунок 1.6 – Окно выбранного каталога с товарами сайта «DNS»

Здесь многие данные схожи с первым сайтом, такое же красивое оформление. Функционально и визуально оба сайта крайне хороши.

1.2 Техническое задание

Поставлена задача реализовать полноценный многостраничный веб-сайт на тему «Магазин мелкой бытовой техники». На сайт должна быть представлена информацию о магазине, такую как контактные данные, описание магазина, а также каталог товаров. Основным содержимым является каталог товаров мелкой бытовой техники.

Для привлечения потенциальных клиентов главная задача сделать удобный интерактивный сайт с подробной информацией о продукте и актуальной информацией.

Основными задачами сайта являются:

1. Предоставление навигационного меню;

2. Предоставление блоков-ссылок;

3. Предоставление информацию о товарах.

На главной странице меню будет располагаться сверху. В меню будут предоставляться ссылки на основную информацию, а также ссылки на элементы каталога товаров.

Веб-страница должна быть кроссбраузерной и адаптивной для мобильной версии, планшетного устройства.

1.3 Выбор средств реализации программного продукта

Курсовая работа выполняется в редакторе кода Visual Studio Code.

При создании веб-страниц были использованы языки: HTML, CSS, JS, XML.

HTML (HyperText Markup Language) — стандартный язык разметки гипертекстовых страниц в Интернете. Страницы успешно интерпретируются браузерами, которые отображают их на экранах различных электронных устройств в удобном для человека виде.

CSS – это фактически язык стилей, который определяет отображение HTML-документов. CSS работает со шрифтами, с цветами символов и фона, с полями, со строками, с высотой и с шириной элементов отображения, с фоновыми изображениями, с позиционированием элементов и со многим другим.

Если HTML необходим для структурирования содержания страницы, то CSS необходим для того, чтобы форматировать это структурированное содержание.

JavaScript это язык, который позволяет применять сложные вещи на web странице — каждый раз, когда на ней происходит что-то большее, чем просто её статичное отображение —без JavaScript не обошлось. Во все основные браузеры встроен интерпретатор JavaScript, именно поэтому они могут выполнять скрипты на странице. JavaScript можно использовать не только в браузере, это полноценный язык, программы на котором можно запускать и на сервере.

XML (eXtensible Markup Language) — это язык разметки, который используется для описания и передачи данных в удобочитаемом формате. XML позволяет определять пользовательские теги, что делает его более гибким, чем предыдущие языки разметки, такие как HTML. XML широко используется для обмена данными между приложениями и системами, включая веб-сервисы, базы данных и многие другие.

1.4 Вывод

В данном разделе были рассмотрены плюсы и минусы аналогичных решений на выбранную тему сайта для предотвращения ошибок в реализации веб-сайта. Были определены основные технические задания по созданию страницы. Также выбраны средства реализации программного продукта.

2 Проектирование страниц веб-сайта

2.1 Выбор способа верстки

Требования проекта включают в себя адаптивность и кроссбраузерность. По этой причине была выбрана grid и flex-box вёрстки.

Grid-модель позволяет создавать структуры, необходимые для обеспечения отзывчивости сайтов на различных устройствах. Это означает, что сайт будет одинаково хорошо смотреться на компьютере, телефоне и планшете.

Flex – это новая технология, которая уже имеет достаточно широкую поддержку браузеров. Flexbox предоставляет инструменты для быстрого создания сложных, гибких макетов, и функции, которые были сложны в традиционных методах CSS.

2.2 Выбор стилевого оформления

Выбор стилевого оформления является важным этапом разработки проекта по нескольким причинам:

Создание уникального имиджа. Стилевое оформление проекта может помочь создать уникальный и запоминающийся имидж. Оно может помочь подчеркнуть уникальность проекта, выделить его на фоне конкурентов и привлечь больше внимания потенциальных пользователей.

Улучшение пользовательского опыта. Стилевое оформление проекта может влиять на пользовательский опыт. Четко продуманное и привлекательное оформление может сделать проект более удобным и привлекательным для пользователей, что может повысить их удовлетворенность использованием проекта.

Усиление бренда. Стилевое оформление может помочь усилить бренд проекта. Оно может помочь создать определенный образ и ассоциации, которые будут связаны с проектом. Это может помочь увеличить узнаваемость и лояльность пользователей.

Улучшение узнаваемости. Хорошо продуманное и привлекательное стилевое оформление может помочь увеличить узнаваемость проекта. Люди запоминают лучше визуальную информацию, поэтому стилевое оформление может помочь проекту стать более узнаваемым и запоминающимся.

Для акцентирования внимания клиента на товаре при оформлении веб-сайта был выбран минимализм.

Минимализм как стиль оформления интернет-магазина имеет свои преимущества и может быть эффективным выбором по следующим причинам:

Фокус на продукте. Минималистичный дизайн позволяет убрать все лишнее, что может отвлечь пользователя от продукта, и сосредоточить его внимание на самом товаре. Это может увеличить вероятность того, что пользователь совершит покупку.

Улучшенная навигация. При минималистичном дизайне интернет-магазина упрощается навигация по сайту и повышается его скорость загрузки. Это может улучшить опыт покупки пользователя и увеличить конверсию.

Стиль и элегантность. Минималистичный дизайн может создать ощущение стиля и элегантности, что может быть важным для привлечения пользователей и установления доверия к бренду.

В целом, стилевое оформление является важным аспектом разработки проекта, который может оказать значительное влияние на его успех и эффективность.

2.3 Выбор шрифтового оформления

Было выбрано разношрифтовое оформление текста на веб-странице, это имеет ряд плюсов:

Улучшение читаемости: разношрифтовое оформление текста может помочь повысить читабельность страницы, так как различные шрифты могут использоваться для различных элементов страницы, таких как заголовки, подзаголовки и основной текст. Это может помочь сделать текст более легким для чтения и более привлекательным для посетителей сайта.

Создание уникального дизайна: использование разных шрифтов может помочь создать уникальный дизайн страницы и помочь выделить ее среди других сайтов. Разнообразие шрифтов может помочь улучшить общий визуальный облик страницы и сделать ее более привлекательной для посетителей.

В целом, использование разношрифтового оформления текста может помочь улучшить читаемость, выделить важные элементы, создать уникальный дизайн и подчеркнуть тему и настроение страницы.

2.4 Разработка логотипа

Логотип – это фирменный знак, который люди ассоциируют с конкретным брендом. Логотип – это не абстрактный набор символов или просто красивое изображение.

Логотип сайта является важным элементом брендинга и визуальной идентичности. Вот несколько причин, почему логотип сайта является важным:

Узнаваемость бренда. Логотип является визуальным символом бренда и помогает создать узнаваемость среди потенциальных клиентов. Когда пользователи видят логотип, они могут легко связать его с брендом и лучше запомнить его.

Отражение имиджа компании. Логотип может отражать имидж компании и ее ценности. Например, если компания заботится об окружающей среде, ее логотип может содержать зеленый цвет и символ природы.

Отличие от конкурентов. Логотип может помочь бренду отличиться от конкурентов и выделиться на фоне других компаний.

Привлечение внимания. Логотип может быть ярким и привлекательным, что поможет привлечь внимание потенциальных клиентов.

Усиление брендовой лояльности. Логотип может помочь усилить брендовую лояльность клиентов. Когда клиенты видят логотип, связанный с их любимым брендом, они могут чувствовать более сильную привязанность к бренду.

Логотип выполнен под концепцию сайта. На нём изображена мелкая бытовая техника и название самого интернет-магазина, чтобы при каждом просмотре логотипа название бросалось в глаза и запоминалось у клиента.



Рисунок 2.2 – Логотип сайта «XKK»

Логотип является частью фирменного стиля, который позволяет сформировать имидж или образ компании ([сайта](https://beseller.by/uslugi/sozdaniye-saytov.html) или [интернет-магазина](https://beseller.by/)), связать товары, услуги, рекламу именно с вашим бизнесом, выделить вас среди конкурентов.

2.5 Разработка пользовательских элементов

В проекте будут представлены элементы пользовательского интерфейса, с которыми пользователь сможет взаимодействовать: блоки с информацией о товаре, навигационное меню, футер с основной информацией.

Навигация – ключевой критерий удобства сайта, своеобразная «карта местности». Представляет собой набор специальных приемов, методов и элементов, дающих возможность посетителям перемещаться между различными страницами ресурса. Навигационное меню представлено на рисунке 2.3.

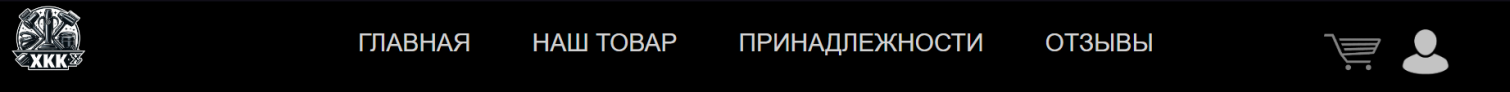


Рисунок 2.3 – Навигационное меню

Блоки-ссылки на товары в интернет-магазине должны быть оформлены так, чтобы привлекать внимание пользователей и помогать им быстро найти нужный товар. Вот несколько рекомендаций для оформления блоков-ссылок на товары:

Изображение товара. В блоке-ссылке должно быть качественное изображение товара. Это поможет привлечь внимание пользователей и показать, как выглядит товар.

Название товара. Название товара должно быть коротким, но информативным. Оно должно содержать ключевые слова, чтобы помочь поисковым системам правильно индексировать товар.

Цена. Цена должна быть ясно указана на блоке-ссылке. Это позволит пользователю быстро понять, насколько дорогой или доступный товар.

Кнопка "Купить" или "Добавить в корзину". Четко выделенная кнопка позволит пользователю легко перейти к оформлению заказа. Также кликабельной является сама картинка с товаром.

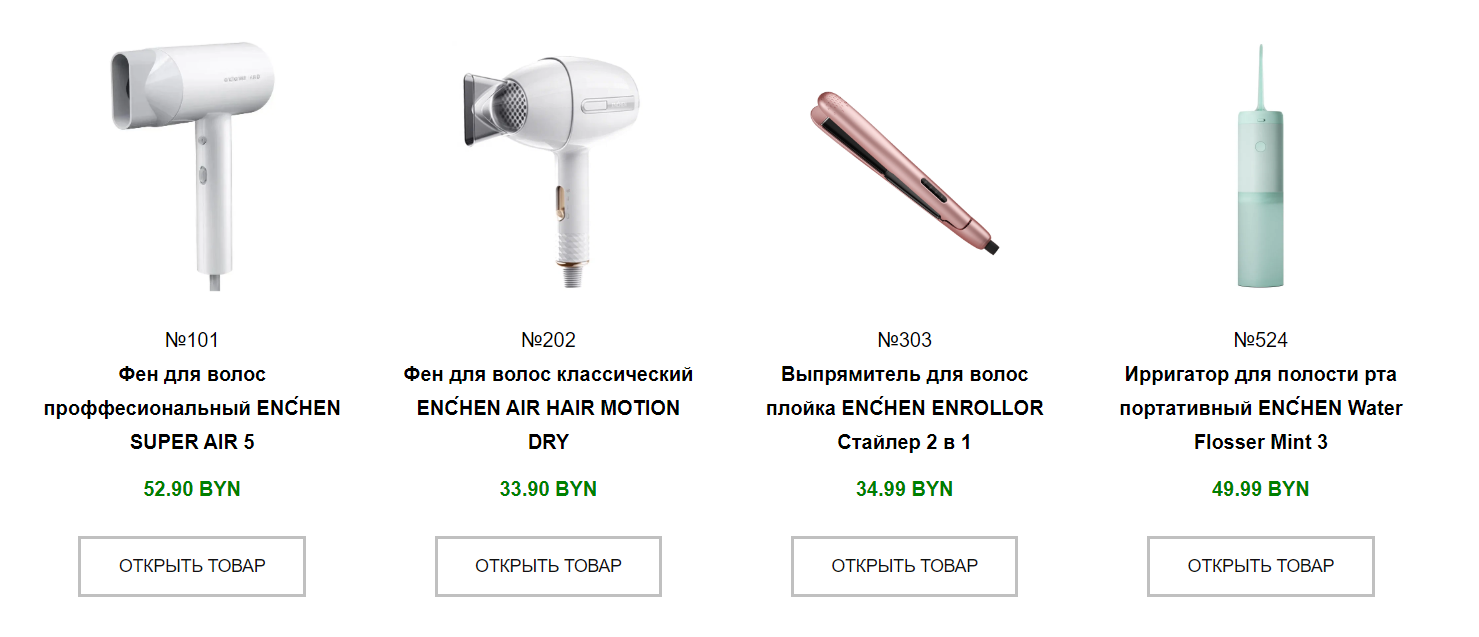


Рисунок 2.4 – Блоки-ссылки на товары

Рассмотрим пользовательские элементы на странице товара. Пользовательские элементы на странице товара выполняют несколько задач:

Показать изображения товара. Хорошие изображения товара могут помочь пользователям лучше понять, как выглядит товар. Пользовательские элементы могут позволить пользователям просматривать изображения товара.

Дать пользователю возможность добавить товар в корзину. На странице товара должна быть кнопка "Добавить в корзину", которая позволит пользователю легко добавить товар в корзину и перейти к оформлению заказа.

Показать рейтинг товара и отзывы. Пользовательские элементы могут показывать рейтинг товара и отзывы от других пользователей, чтобы помочь пользователям принять решение о покупке.

Предложить сопутствующие товары. Некоторые пользовательские элементы могут показывать сопутствующие товары, которые могут быть интересны пользователям и повысить объем покупок.

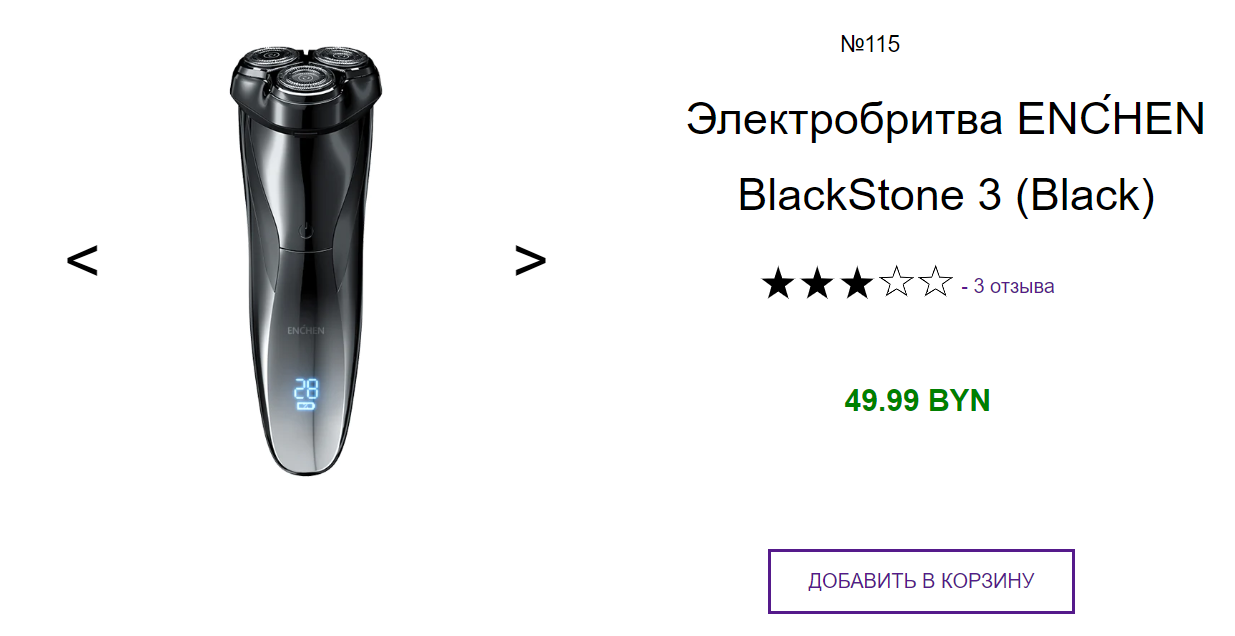


Рисунок 2.5 – Страница товара

Над кнопкой добавления товара находиться блок с основной информацией о нем: номер артикула, оценка, цена.

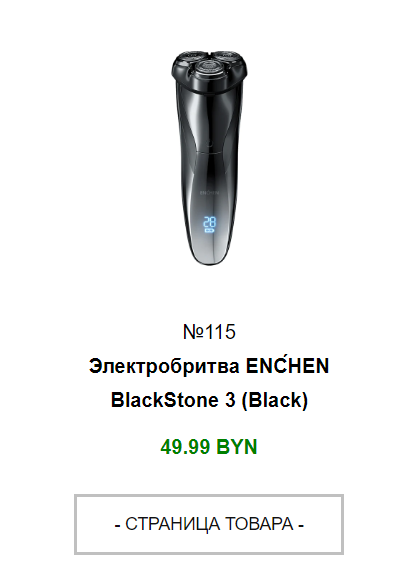


Рисунок 2.6 – Вид товара в каталоге

Таким образом пользовательские элементы помогают увеличить удобство использования сайта клиентом и предоставить все виды информации.

2.6 Разработка спецэффектов

На сайте присутствуют динамические эффекты, которые придадут сайту интерактивность и сделают его более привлекательным для пользователя.



Рисунок 2.7.1 – Слайдер с изображениями



Рисунок 2.7.2 – Слайдер с изображениями



Рисунок 2.7.3 – Слайдер с изображениями



Рисунок 2.7.4 – Слайдер с изображениями

Слайдеры – это графические элементы, которые могут добавить интересный визуальный эффект на страницу товара. Вот несколько плюсов использования слайдеров на странице товара:

Многофункциональность. Слайдер может отображать много изображений или дополнительную информацию.

Экономия места. Слайдер может помочь сэкономить место на странице. Вместо размещения нескольких изображений друг под другом, слайдер может позволить отобразить все изображения в одном компактном блоке.

Простота использования. Слайдеры обычно имеют простой интерфейс управления, который позволяет пользователям легко просматривать изображения или информацию.

Эстетика. Слайдеры могут помочь улучшить эстетику страницы товара, особенно если они оформлены со вкусом и хорошо сочетаются с дизайном сайта.

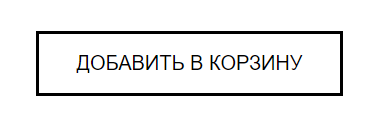


Рисунок 2.8.2 – Кнопка добавить в корзину до нажатия

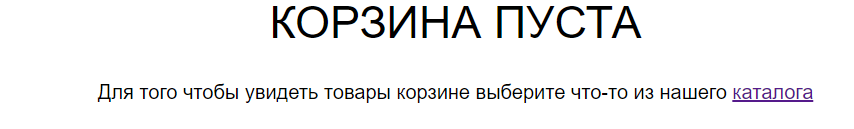


Рисунок 2.8.2 – Страница пустой корзины

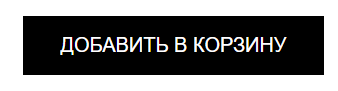


Рисунок 2.9.1 – Кнопка добавить в корзину при наведении и последующем нажатии

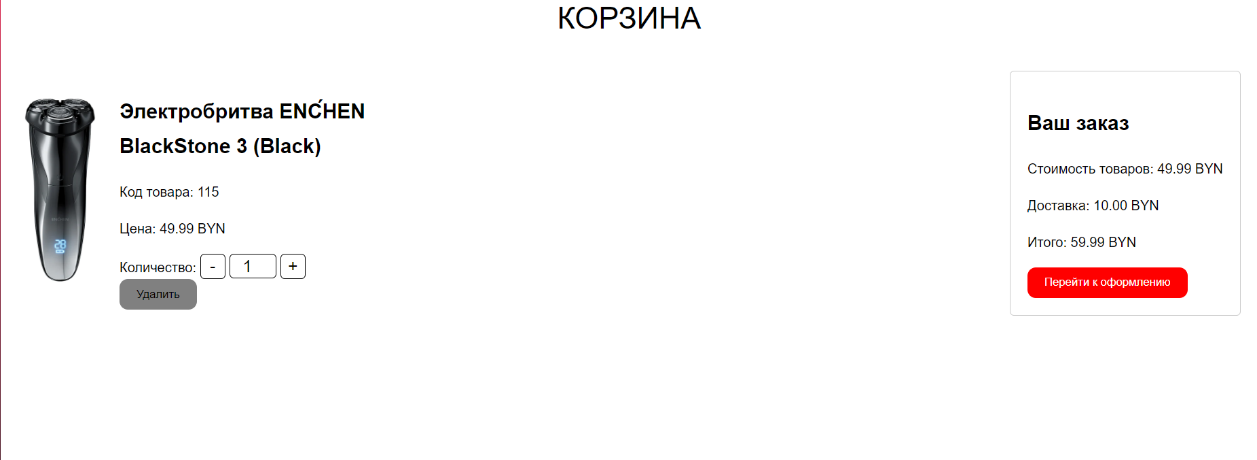


Рисунок 2.9.2 – Страница корзины с добавленным товаром

Анимация кнопки при наведении также имеет некоторые плюсы: усиление визуальной привлекательности кнопки и увеличение вероятности того, что пользователь обратит на нее внимание; возможность повышения конверсии, так как дополнительные визуальные эффекты могут привлечь внимание пользователя и стимулировать его к действию (например, к добавлению товара в корзину).



Рисунок 2.10 – Подвижная галерея рекомендованных товаров

Подвижная галерея товаров в интернет-магазине может иметь следующие плюсы: увеличение количества товаров, которые можно показать на странице, что может повысить эффективность использования пространства на странице; повышение удобства пользователей при просмотре товаров и быстром переходе к интересующим их позициям; возможность использования анимации для привлечения внимания пользователей и улучшения пользовательского опыта; демонстрация разнообразия товаров и возможности выбора, что может стимулировать пользователей к дальнейшему исследованию каталога товаров и увеличению среднего чека.

2.7 Выводы

В данном разделе был создан прототип веб-сайта, определены задачи по созданию содержания на веб-сайте. Выбрано единое цветовое и шрифтовое оформления сайта для удобства просмотра пользователем. Разработан дизайн и расположение на сайте пользовательских элементов, спецэффектов и анимации. Логотип выполнен под концепцию сайта.

3 Реализация структуры веб-сайта

3.1 Структура HTML-документа

Структура веб-сайта представляет собой организацию связанных между собой документов, которая позволяет пользователям легко перемещаться по страницам. При формировании структуры необходимо учитывать потребности посетителей, их запросы и интересы.

Хорошая структура HTML-документа имеет большое значение для эффективной работы сайта. Вот несколько причин, почему это так важно:

Улучшение доступности: хорошая структура HTML-документа может сделать сайт более доступным для пользователей с ограниченными возможностями. Например, правильное использование заголовков и других элементов разметки может помочь пользователям, использующим программы чтения с экрана, легче понимать контент.

Хорошая структура HTML-документа может помочь поисковым системам лучше понимать контент страницы. Это может привести к улучшению рейтинга сайта и его поисковой выдаче.

Улучшение скорости загрузки: хорошая структура HTML-документа может ускорить загрузку страницы. Например, правильное использование семантических тегов может помочь браузеру быстрее отрисовать страницу.

Улучшение совместимости: хорошая структура HTML-документа может помочь сделать сайт более совместимым с различными браузерами и устройствами. Это может улучшить пользовательский опыт и увеличить количество посетителей на сайте.

Тег <head> предназначен для хранения служебных элементов, которые не отображаются на странице, за исключением заголовка <title>.

Основное содержимое сайта размещается внутри тега <body>, который включает в себя семантические теги, такие как header, section и footer, которые помогают программам понимать тип информации, содержащейся на странице.

Структура HTML главной страницы представлена в листинге 3.1.

|  |
| --- |
| < ! DOCTYPE html>  <html>  <head>…</head>  <body>  <header class=”header”>…</header>  <section class=”section-1”>…</section>  <section class=”section-2”>…</section>  <section class=”section-3”>…</section>  <section class=”section-4”>…</section>  <section class=”section-5”>…</section>  <footer class=”footer”>…</footer>  </body>  </html> |

Листинг 3.1 – Структура HTML главной страницы

С помощью тега <header> была создана шапка сайта для всех страниц и создано навигационное меню. “Подвал” страницы был, идентичный на всех страницах сайта, был реализован с помощью тега <footer>. Теги <section> были созданы для размещения на страницы и разметки основной информации.

3.2 Добавление таблиц стилей Scss и CSS

В ходе разработки проекта были применены различные подходы к подключению таблиц стилей, включая внешнее, внутреннее и строковое подключение.

|  |
| --- |
| <head>  <title>главная</head>  <link rel=”stylesheet” href=”общее.css”>  <link rel=”stylesheet” href=”главная.css”>  </head> |

Листинг 3.2 – Пример внешнего подключения

|  |
| --- |
| <style>  .error{  color: blue;  font-size: 52px;  margin-top: 68px;  }  .success{  color: black;  font-size: 10px;  margin-top: 10px;  }  </style> |

Листинг 3.3 – Пример внутреннего подключения

Для определения стилей для повторяющихся элементов и основной разметки страниц было использовано внешнее подключение, при котором стили размещаются в отдельном файле и могут быть использованы на любой странице сайта. Внутреннее подключение было применено для внесения изменений в контент на второстепенных страницах при минимальном объеме кода.

|  |
| --- |
| <a style=”text-decoration: none;”><<Вернуться на главную</a> |

Листинг 3.4 – Пример внутреннего подключения

Для небольших изменений в стиле текста на страницах сайта было использовано строковое подключение. Все таблицы стилей были организованы с использованием селекторов по классу, что позволило улучшить структуру кода и обеспечить более эффективную поддержку сайта в будущем. Такой подход позволяет быстро и легко вносить изменения в стили сайта, облегчает сопровождение проекта и ускоряет его загрузку.

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.

|  |
| --- |
| <?xml version="1.0" standalone="no"?>  <!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 20010904//EN"  "http://www.w3.org/TR/2001/REC-SVG-20010904/DTD/svg10.dtd">  <svg version="1.0" xmlns="http://www.w3.org/2000/svg"  width="489.000000pt" height="480.000000pt" viewBox="0 0 489.000000 480.000000"  preserveAspectRatio="xMidYMid meet">  <g transform="translate(0.000000,480.000000) scale(0.100000,-0.100000)"  fill="#000000" stroke="none">  <path d="M2125 3818 c-154 -468 -284 -852 -290 -853 -5 -2 -401 -13 -880 -25  -478 -11 -871 -22 -873 -24 -1 -2 304 -230 680 -506 376 -277 689 -508 696  -515 10 -9 -39 -182 -242 -856 -140 -464 -253 -846 -251 -848 2 -1 326 217  720 485 394 269 718 489 720 489 2 0 326 -220 720 -489 394 -268 718 -486 720  -485 2 2 -111 384 -251 848 -203 674 -252 847 -242 856 7 7 320 238 696 515  376 276 681 504 680 506 -2 2 -397 13 -878 25 -481 12 -878 24 -882 28 -3 3  -131 386 -284 851 -152 465 -277 846 -278 847 -1 1 -127 -381 -281 -849z"/>  </g>  </svg> |

Листинг 3.5 – Пример использования стандартов XML(SVG)

SVG-формат удобен для использования веб-дизайнерами и разработчиками по нескольким причинам:

Масштабируемость: SVG-файлы могут быть масштабированы без потери качества изображения.

Малый размер: SVG-файлы обычно имеют меньший размер, чем другие форматы изображений, такие как JPG или PNG, что уменьшает время загрузки страницы и повышает производительность сайта.

Данные для связи удобно хранятся отдельно в XML-документе.

XML-формат удобен для хранения данных по нескольким причинам:

Независимость от программного обеспечения: XML-формат не зависит от программного обеспечения или операционной системы, что делает его переносимым между различными платформами.

Возможность расширения: XML-формат позволяет создавать собственные теги и атрибуты, что дает возможность расширять его для конкретных нужд проекта.

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.

|  |
| --- |
| window.addEventListener('load', function() {  var xhttp = new XMLHttpRequest();  xhttp.onreadystatechange = function() {  if (this.readyState == 4 && this.status == 200) {  document.getElementById("contact-info-container").innerHTML = this.responseText;  }  };  xhttp.open("GET", "xkk.xml", true);  xhttp.send();  }); |

Листинг 3.6 – Пример использования JS для загрузки данных из файла

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.

3.5 Выводы

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).

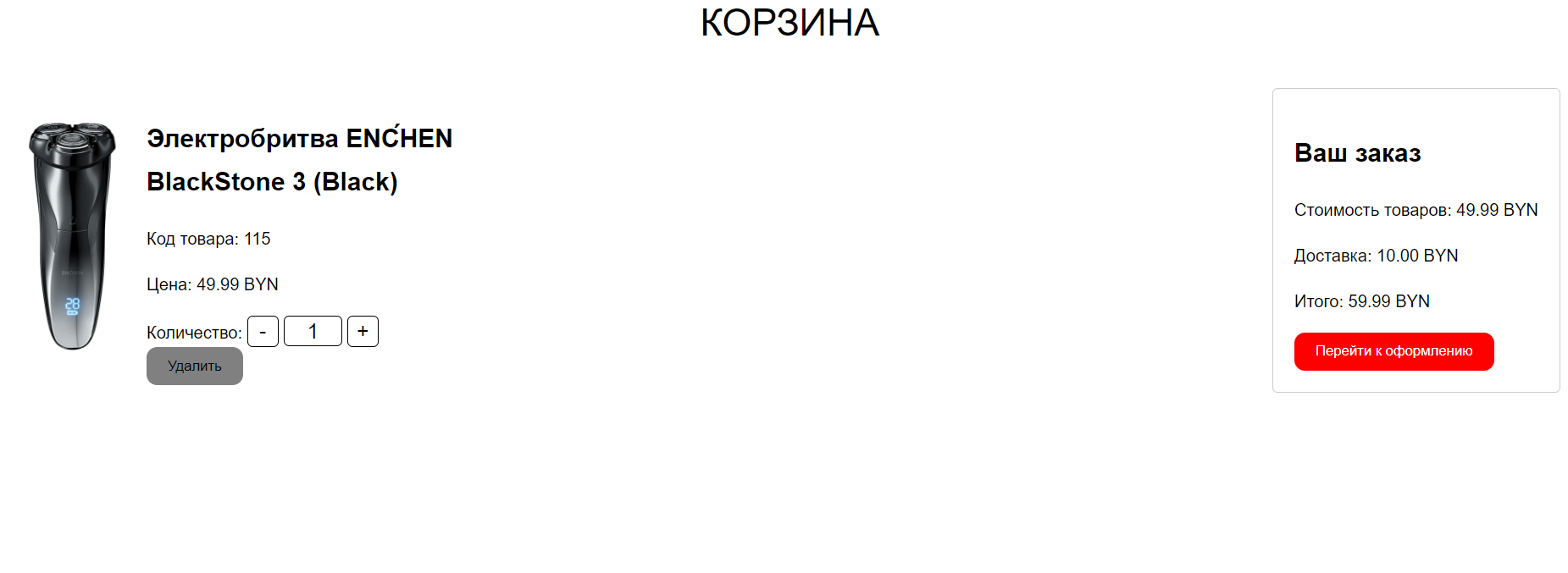


Рисунок 4.1 – Вид страницы для компьютера

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

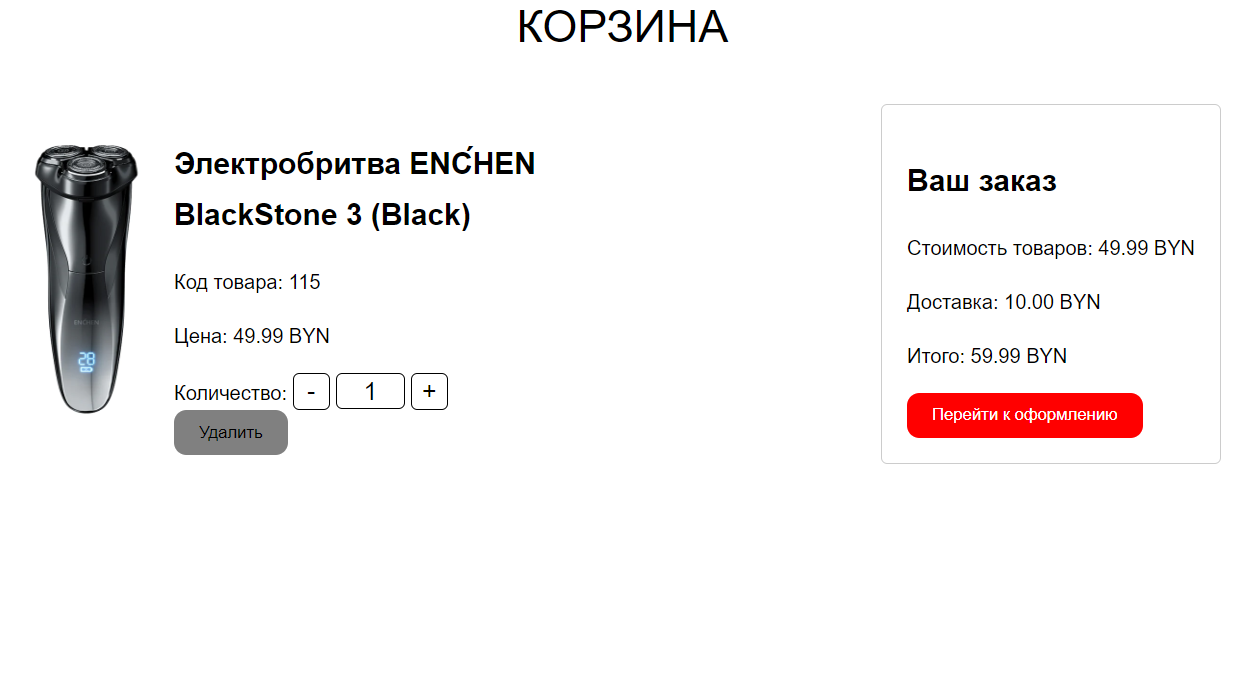


Рисунок 4.2 – Вид страницы для планшетных устройств

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).

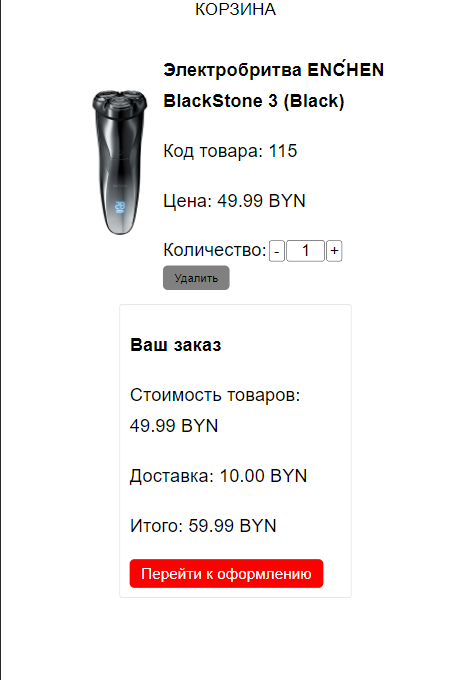


Рисунок 4.3 – Вид страницы для мобильных устройств

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 – важный критерий корректной работы веб-ресурса. Это инструмент поддержания посетительской лояльности, повышения конверсии и поднятия рейтинга в поисковых системах.

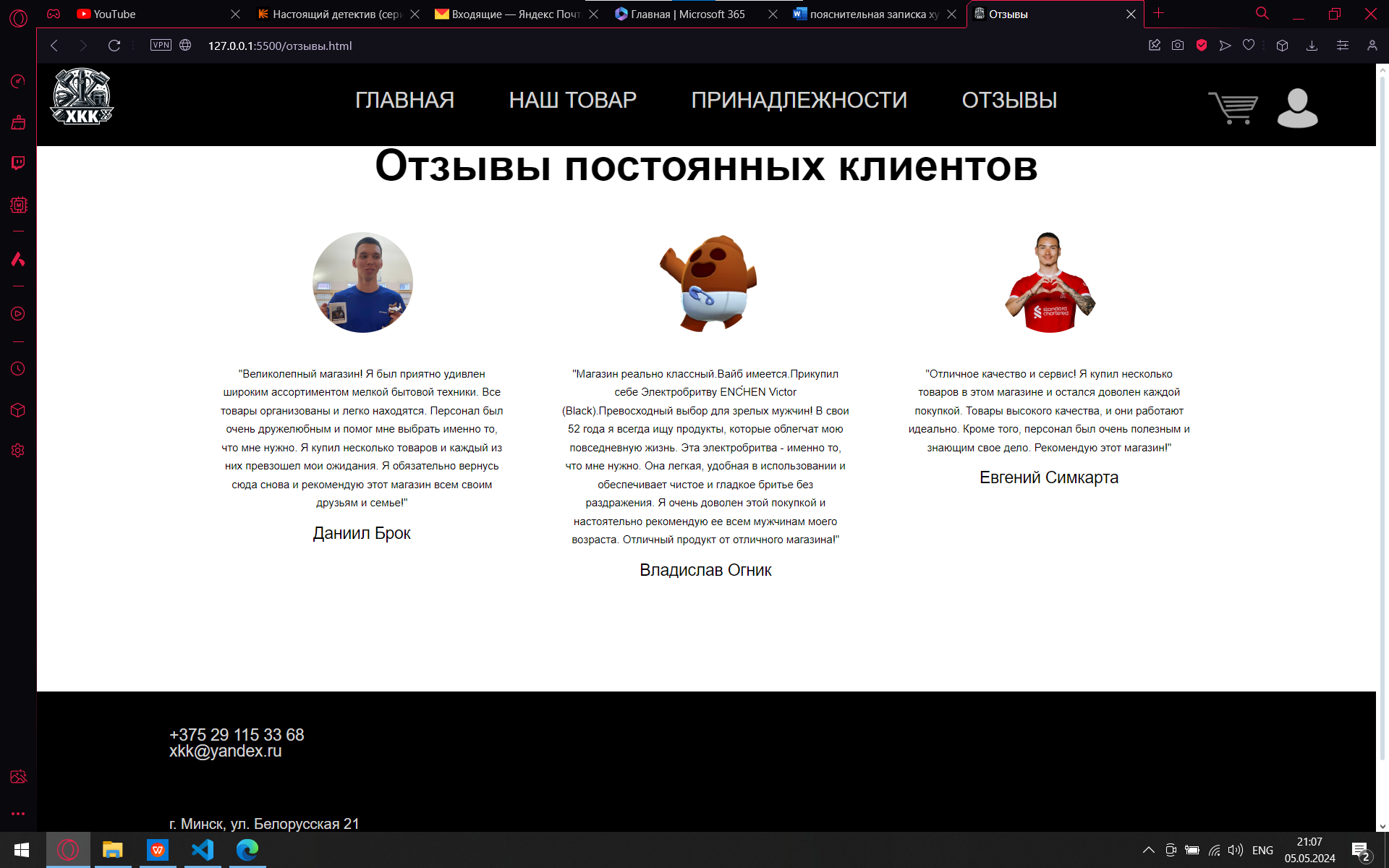


Рисунок 4.4 – Вид страницы в браузере Opera GX

После написания основной структуры страницы на HTML и внешнего стилевого оформления появился вопрос о тестировании. Веб-сайт был открыт при помощи различных браузеров. Существует множество браузеров, каждый из которых имеет свои особенности и способы интерпретации кода. Кроссбраузерность гарантирует, что ваш сайт будет работать одинаково хорошо во всех этих браузерах. Пользователи могут использовать разные браузеры в зависимости от своих предпочтений, операционной системы или устройства. Кроссбраузерность обеспечивает лучший пользовательский опыт для всех, независимо от выбора браузера. Более того, кроссбраузерное тестирование позволяет разработчикам уверенно выпускать обновления, зная, что функциональность сайта останется стабильной. Это также способствует улучшению общего качества кода, поскольку требует строгого соблюдения стандартов веб-разработки. В связи с этим, тестирование на кроссбраузерность становится неотъемлемой частью процесса разработки, помогая обнаружить и исправить возможные проблемы совместимости. Таким образом, кроссбраузерность является ключевым фактором при разработке веб-сайтов, чтобы обеспечить широкую доступность и функциональность для всех пользователей.

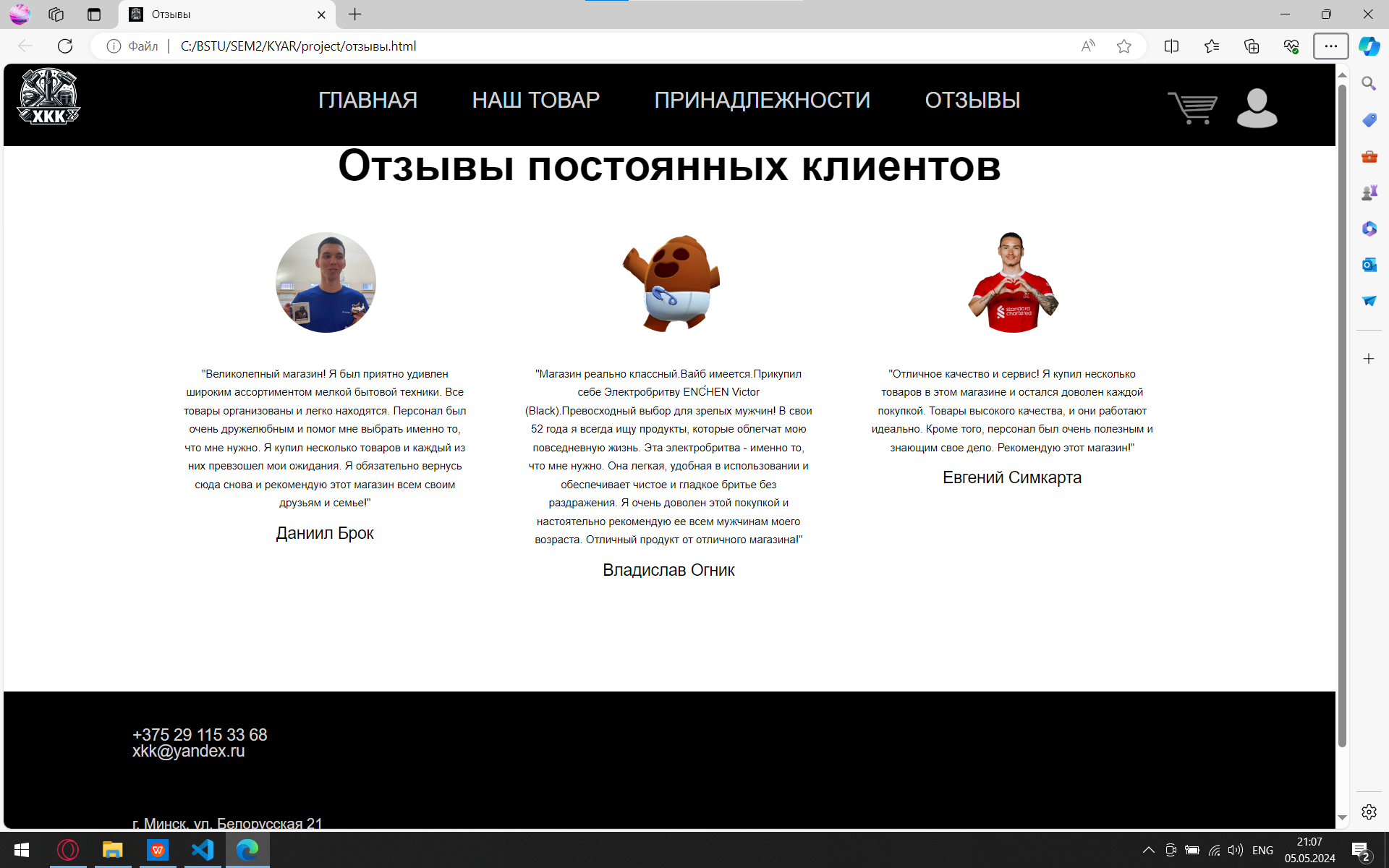


Рисунок 4.5 – Вид страницы в браузере Microsoft Edge

Сайт, который выглядит и работает хорошо во всех браузерах, создает профессиональное впечатление и демонстрирует заботу о пользователях.

4.3 Руководство пользователя

Интерфейс сайта интуитивно понятен пользователю. Основное взаимодействие с сайтом в поиске товара удобнее всего осуществлять с помощью ссылок на различные категории (рис. 4.6).



Рисунок 4.6 – Ссылки на категории

С их помощью пользователь может перейти на интересующую его категорию, для последующего выбора товара на странице (рис 4.7).



Рисунок 4.7 – Ссылки на товары

На рисунке представлена часть страницы по выбранной категории. На ней расположены ссылки на сами товары.

После переходу по блоку-ссылке на страницу товара пользователь может ознакомиться с информацией о нем(рис.4.8).

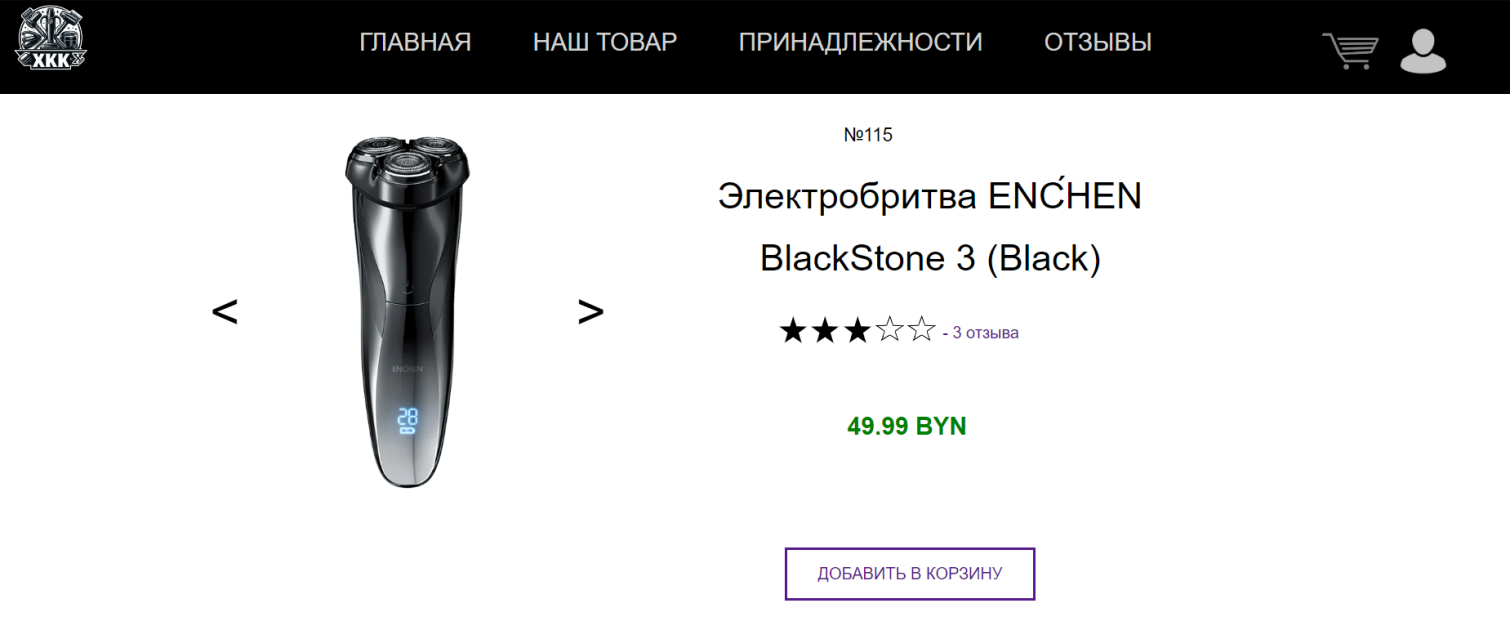


Рисунок 4.8 – Основная информация о товаре

После этого по нажатую на кнопку «Добавить к корзину» покупатель может увидеть свой товар в корзине и после чего оформить заказ оформить заказ(рис 4.9).

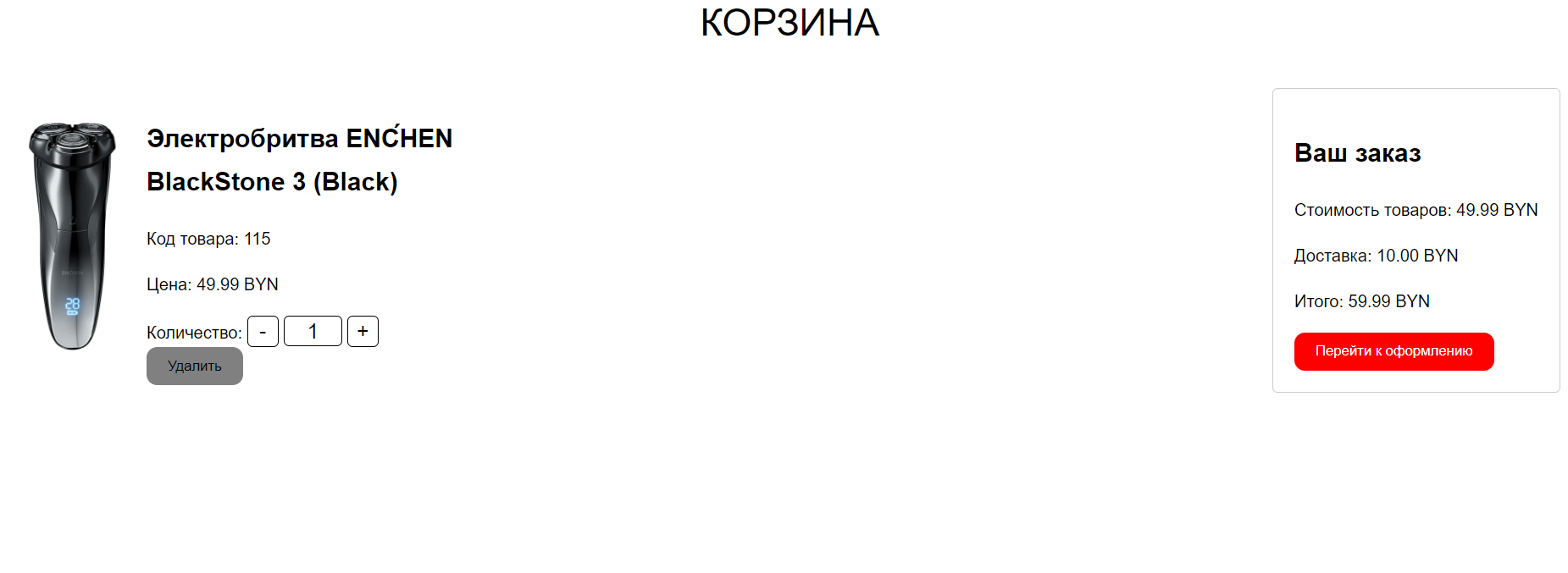


Рисунок 4.9 – Вид страницы с добавленным в корзину товаром

При возникновении каких-либо вопросов пользователь может воспользоваться специальной формой(рис. 4.10).

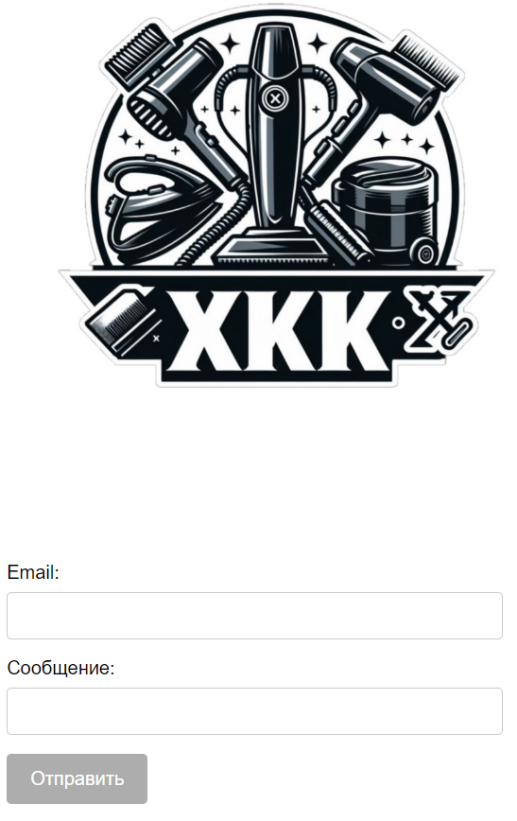


Рисунок 4.10 – Форма для связи с сотрудниками магазина

Таким образом, был разработан сайт с целью обеспечить максимально простое и интуитивно понятное использование.

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 4.11 – 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

В данном разделе было рассмотрено, как сайт ведет в себя на разных устройствах и в разных браузерах. С помощью тестирования были выявлена некоторые недочеты для дальнейшего усовершенствования продукта.

Заключение

В ходе проделанной работы был создан полноценный сайт магазина для онлайн-заказа мелкой бытовой техники.

При разработке данного веб-сайта был использован широкий ряд языков разметки и таблиц стиля, который позволил увеличить функциональность и многократно повысил качество конечного продукта. Была использована адаптивная верстка для корректного отображения на всех устройствах.

В ходе выполнения данной задачи были рассмотрены плюсы и минусы аналогичных решений. В процессе осмотра выяснилось, каких тенденции в разработке дизайна веб-сайта стоит придерживаться.

При формировании технического задания были продумано содержание основных страниц веб-сайта.

Были выбраны инструменты для разработки веб-сайта.

При проектировании были разработаны прототипы, а далее созданы макеты дизайна страниц веб-сайта, при опоре на который был реализован сам веб-сайт. Для разработки прототипов и макетов станиц было использована программа Figma, ранее не изученная в ходе учебного процесса.

Были разработаны пользовательские элементы, спецэффекты и логотип, для идентификации веб-сайта, среди миллионов других интернет ресурсов.

Был создан XML-документ для последующей вставки на веб-страницу HTML.

В ходе тестирования была продемонстрирована валидность, кроссбраузерность и адаптивность веб-сайта.

Таким образом была полностью достигнута поставленная цель по разработке веб-сайта и соблюдены все требования.

Список использованных источников

1. Документация по Figma [Электронный ресурс]. — Режим доступа: https://help.figma.com
2. Аналог «5 ЭЛЕМЕНТ» [Электронный ресурс]. — Режим доступа: https://5element.by
3. Аналог «ЭЛЕКТРОsila.by» [Электронный ресурс]. — Режим доступа: https://sila.by
4. Аналог «DNS» [Электронный ресурс]. — Режим доступа: https://www.dns-shop.ru
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/Xutrikk