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: Интернет-Магазин электротоваров.

**Исполнитель**

студент 1 курса 6 группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е. И. Лахвич

подпись, дата

**Руководитель**

ассистент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е. И. Комарова

должность, учен. степень, ученое звание подпись, дата

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

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

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

дата, подпись

Курсовой проект защищен с оценкой

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е. И. Комарова

подпись дата инициалы и фамилия

Содержание

[**Введение** 3](#_Toc134038402)

[**1. Постановка задач** 4](#_Toc134038403)

[1.1 Обзор аналогичных решений. 4](#_Toc134038404)

[1.2 Техническое задание. 7](#_Toc134038405)

[1.3 Выбор средств реализации программного продуктa. 8](#_Toc134038406)

[1.4 Вывод. 8](#_Toc134038407)

[**2. Макетирование страниц веб-сайта.** 9](#_Toc134038408)

[2.1 Выбор способа вёрстки. 9](#_Toc134038409)

[2.2 Выбор стилевого оформления. 9](#_Toc134038410)

[2.3 Выбор шрифтового оформления. 9](#_Toc134038411)

[2.4 Разработка логотипа. 10](#_Toc134038412)

[2.5 Разработка пользовательских элементов. 10](#_Toc134038413)

[2.6 Разработка спецэффектов. 11](#_Toc134038414)

[2.7 Вывод. 11](#_Toc134038415)

[**3. Реализация структуры веб-сайта** 12](#_Toc134038416)

[3.1. Структура HTML-документа. 12](#_Toc134038417)

[3.2. Добавление таблиц стилей Scss и CSS. 13](#_Toc134038418)

[3.3. Использование стандартов XML (SVG) 18](#_Toc134038419)

[3.3. Управление элементами DOM 19](#_Toc134038420)

[3.5 Выводы 20](#_Toc134038421)

[4.1. Адаптивный дизайн веб-сайта 21](#_Toc134038422)

[4.2 Кроссбраузерность веб-сайта 21](#_Toc134038423)

[4.3. Руководство пользователя 22](#_Toc134038424)

# **Введение**

Электронный магазин — это прикладная система, построенная с использованием технологии системы электронной торговли. Подобно обычному магазину, электронный магазин реализует следующие основные функции: представление товаров (услуг) покупателю, обработку заказов, продажу и доставку товаров.

Интернет-магазин объединяет элементы прямого маркетинга с образом посещения традиционного магазина. Отличительной чертой интернет-магазина по сравнению с обычной формой торговли является то, что интерактивный магазин может предложить значительно большее количество товаров и услуг, и обеспечить потребителей значительно большим объемом информации, необходимым для принятия решения о покупке. Кроме того, за счет использования компьютерных технологий возможна персонализация подхода к каждому из клиентов, исходя из истории его посещений магазина и сделанных ранее покупок.

Аудитория в интернете стремительно растет, а продажи через интернет в крупных городах достигают  до 25%, при этом специалисты подчеркивают тенденцию к росту продаж именно через интернет. Ежегодно количество интернет-магазинов увеличивается, так как это действительно прибыльно и удобно для покупателя, не говоря об экономии бюджета и времени. Интернет-магазин работает круглосуточно и может продавать определенные товары в автоматическом режиме без участия продавца. К преимуществам так же можно отнести то, что не надо закупать товар заранее, а это существенная экономия, на складских помещениях. Необходимо лишь договориться с поставщиками, и в нужный момент, просто выкупить товар, который у вас заказан.

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

Задачи курсового проекта является:

* 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 задач**

## 1.1 Обзор аналогичных решений.

Курсовой проект представляет собой веб-сайт по продаже электротоваров. В сфере деятельности магазинов электротоваров уже существуют некоторые аналогичные решения. В этом разделе будут приведены веб-сайты существующих веб-сайтов по продаже электротоваров.

Сайт kilovolt.by. Цветовая палитра сайта приятна глазу, шрифт читабельный. Дизайн представлен на рисунке 1.1. Также хорошо проработана мобильная версия сайта. (рисунок 1.2).

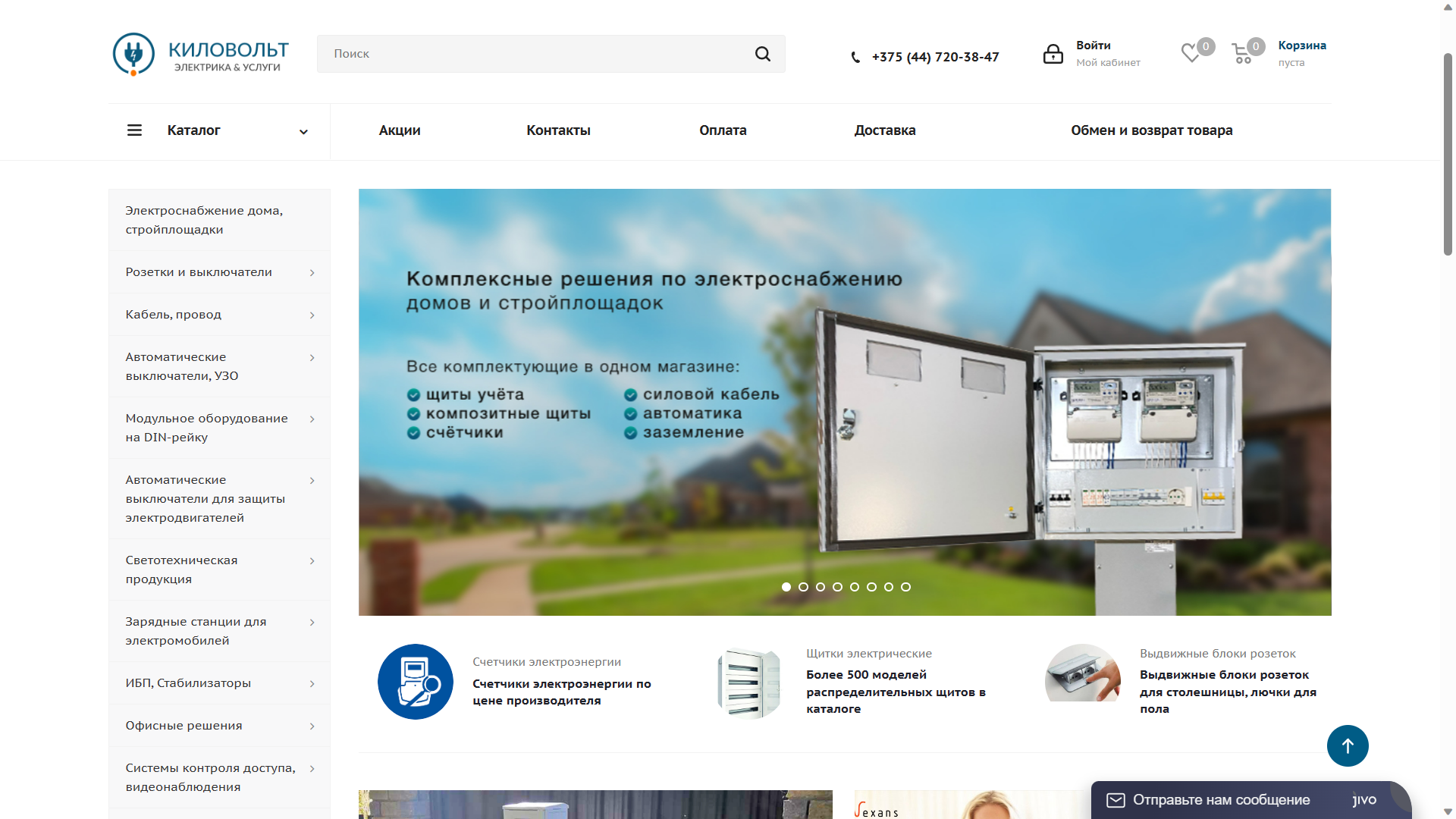


Рисунок 1.1 – Веб-сайт kilovolt.by

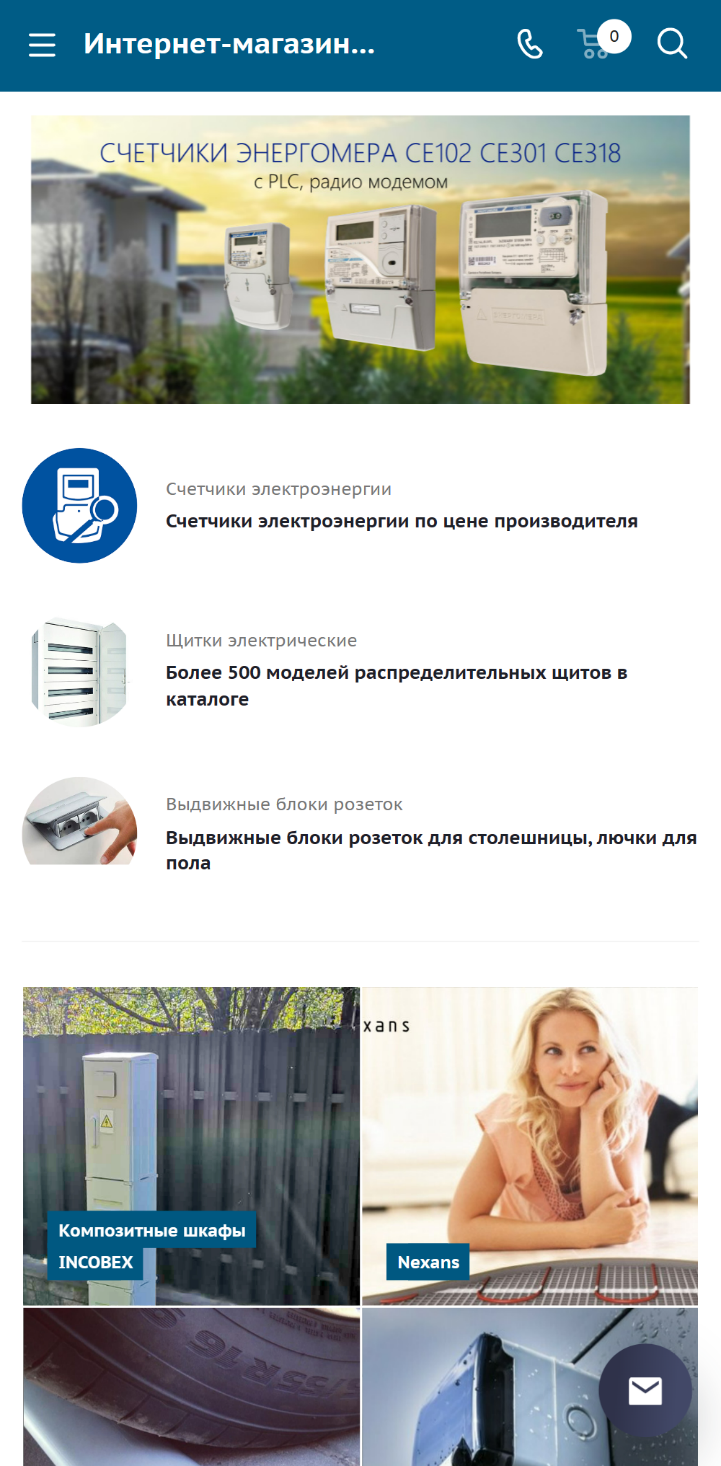


Рисунок 1.2 – Веб-сайт kilovolt.by в мобильной версии

Следующий сайт eurosvet.by. В разделе «Новости» информация представлена не особо приятным дизайном, шрифты и его размеры просто нечитабельны (рисунок 1.3). В мобильной версии сайта присутствует адаптивность страницы, но плохо реализована (рисунок 1.4).

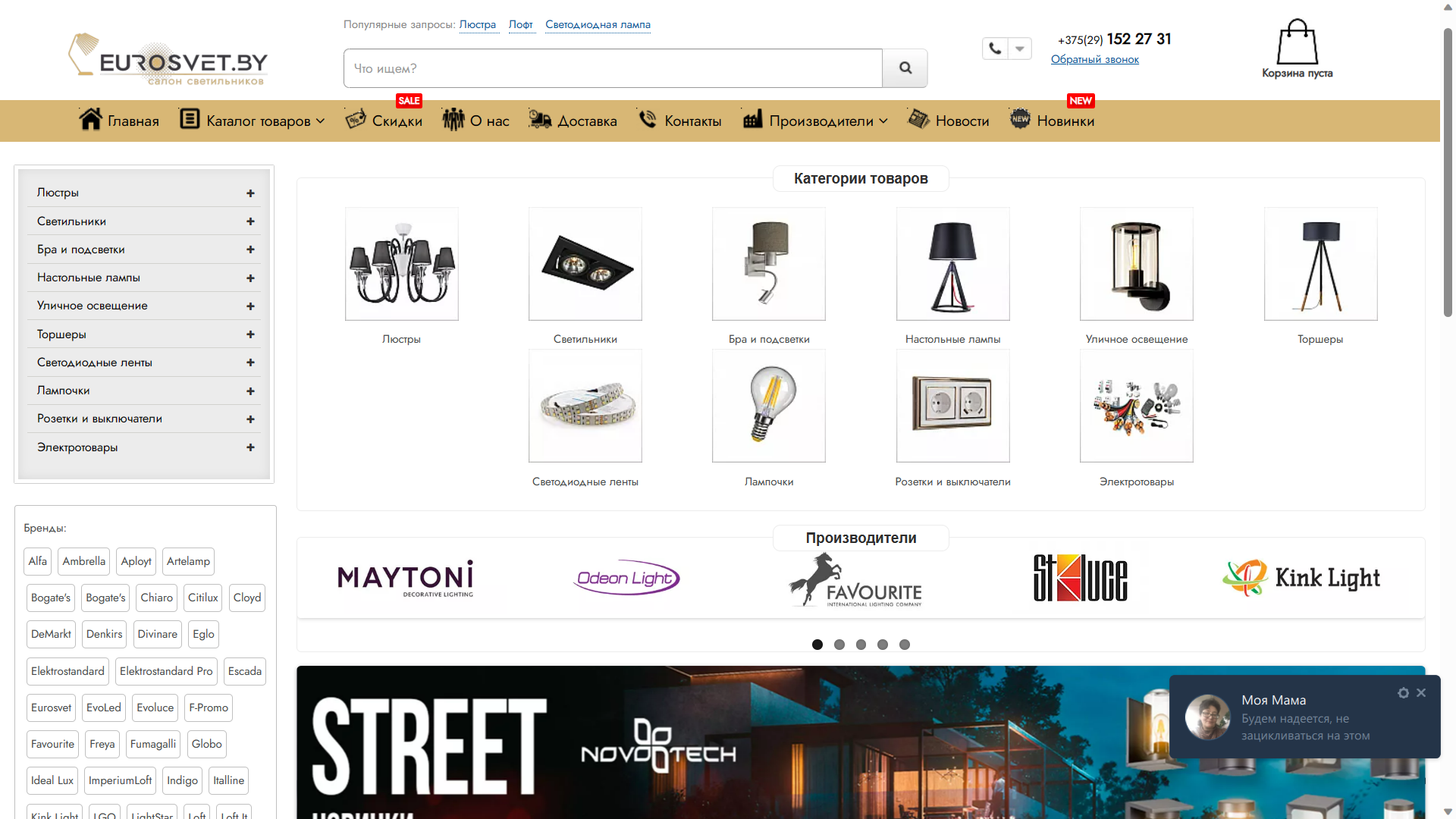


Рисунок 1.2 – Веб-сайт eurosvet.by

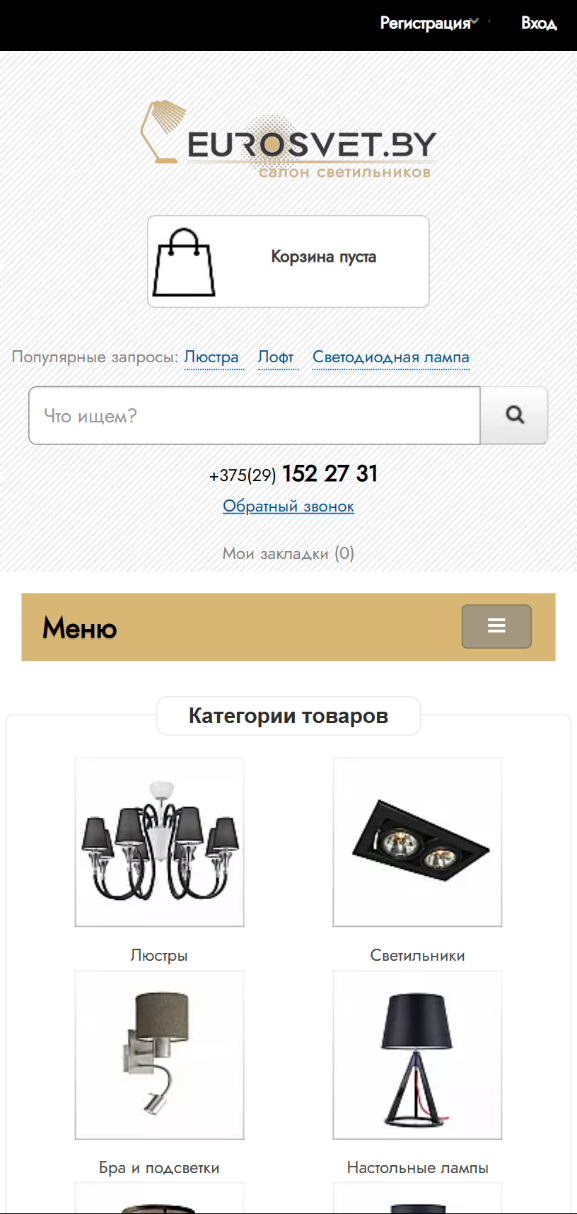


Рисунок 1.2 – Веб-сайт eurosvet.by в мобильной версии

## 1.2 Техническое задание.

В данном проекте требуется создать многостраничный сайт по продаже электротоваров. Сайт должен содержать общую информацию о магазине, такую как контактные данные, описание магазина, а также каталог товаров, доставка и форму заказа. Основным контентом являются текстовые данные, описание товаров, фотографии товаров.

Главной задачей сайта является предоставление данных о товарах для привлечения внимания потенциальных клиентов. Сайт должен быть интерактивным, содержать актуальную информацию. Данный сайт предназначен для желающих приобрести электротоваров.

На главной странице меню будет располагаться сверху. В меню будут предоставляться ссылки на основную информацию (о магазине, контакты, информация и т.д.), а также ссылки на элементы каталога товаров.

В мобильной версии (ширина до 768px) 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 о товарах, страницы описания магазина, контактной страницы, страницы просмотра одного товара каталога, страницы для заказа товара, новостная страница.

На странице «Каталог» должна быть расположена основная информация. На странице «О нас» будет расположено описание самого магазина. На странице «Информация» должна быть расположена информация об условиях доставки и оплаты товара, гарантии и т.д. На странице «Категория товаров» будут размещены товары одной категории. На странице «Товар» будет размещена вся информация о товаре (описание, характеристики, цена и т.д.). На странице «Новости» будут размещены актуальная информация от компании.

Основные требования к программному продукту являются: создание дизайна, проведение анализа существующих решений, создание прототипа, адаптивность и кроссбраузерность сайта (корректное отображение в браузерах, таких как Google Chrome, Mozilla Firefox, Microsoft Edge, Opera.). В проекте используется гипертекстовый язык разметки HTML, CSS/Scss, JavaScript, графические элементы в формате SVG и XML.

Задачи курсового проекта:

- Упростить поиск информации о магазине для пользователей.

- Размещение основной информации на сайте.

- Размещение информации о товаре.

- Представление имиджа компании.

## 1.3 Выбор средств реализации программного продуктa.

Для реализации проекта были выбраны языки: HTML, CSS/Sass, JavaScript, XML. Структура сайта создана с помощью языка разметки HTML, дизайн сайта оформлен с помощью CSS/Scss, а также для создания интерактивных веб-страниц используется JavaScript. В XML формате будут храниться данные о товарах.

Курсовая работа выполняется в редакторе кода VS Code.

## 1.4 Вывод.

В данном разделе были представлены аналогичные решения, были проанализированы их недостатки и преимущества, для создания в последующем качественного веб-сайта. Были поставлены конкретные задачи для создания интерактивного информационного ресурса. 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. Вышеперечисленные пункты понадобятся для достижения поставленной задачи и для создания в дальнейшем качественного продукта.

# **2. Макетирование страниц веб-сайта.**

## 2.1 Выбор способа вёрстки.

В соответствие с требованиями к проекту, в частности адаптивность и кроссбраузерность сайта, было принято решение об использовании Grid-верстки. Grid обеспечивает двухмерное выравнивание, использует нисходящий подход к макету, допускает явное перекрытие элементов и обладает более мощными связующими возможностями. Flexbox фокусируется на распределении пространства по оси, использует более простой восходящий подход к макету, может использовать систему переноса строк на основе размера контента для управления своей вторичной осью и опирается на базовую иерархию разметки для построения более сложных макетов.

Для создания отдельных блоков страницы, такие как горизонтальное меню, будет использована 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 и разработаны макеты страниц. Макеты веб-сайта смотреть в приложении 2.

## 2.2 Выбор стилевого оформления.

Для реализации стилевого оформления сайта было выбрано минималистическое направление с элементами классического веб-дизайна.

Сайт не загромождается лишними деталями (боковыми панелями и т.п.), большим количеством спецэффектов. Акцент делается на основной контент и фотографии товаров.

Все изображения выдержаны в одном стиле, не перегружены мелкими элементами. Цветовые решения не выделяющие, довольно нейтральные. Основные цвета: #FFFFFF, #C0C0C0, #BDB912.

## 2.3 Выбор шрифтового оформления.

В данном проекте будет использоваться основной шрифт: Rubik. Шрифт подключался с помощью библиотеки Google Fonts. Шрифт применялся ко всему тексту.

## 2.4 Разработка логотипа.

Разработка логотипа осуществлялась в графическом редакторе Inkscape. Логотип представляет собой название магазина (TokShop), а также полупрозрачный серый круг с первыми буквами названий магазина, а за ним – молния, что отражает сферу деятельности магазина. Надписи «TokShop» сначала были выполнены в текстовом формате. Далее название компании было размещено под молнией и все вместе было переведено в SVG формат. Дизайн логотипа представлен на рисунке 2.1

Изображение выглядит как логотип

Автоматически созданное описание

Рисунок 2.1 – Логотип TokShop

## 2.5 Разработка пользовательских элементов.

В проекте будут представлены такие элементы пользовательского интерфейса, как форма заявок и карта местности. Элементы представлены на рисунках 2.2, 2.3.

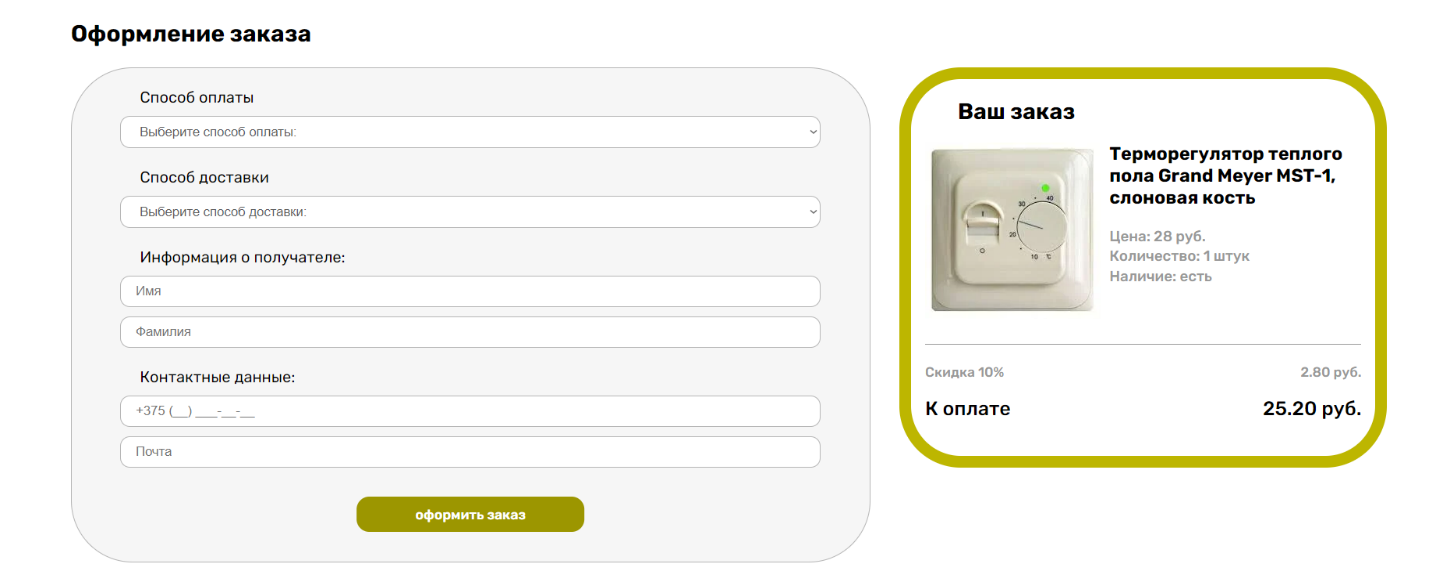


Рисунок 2.2 – Форма заказа

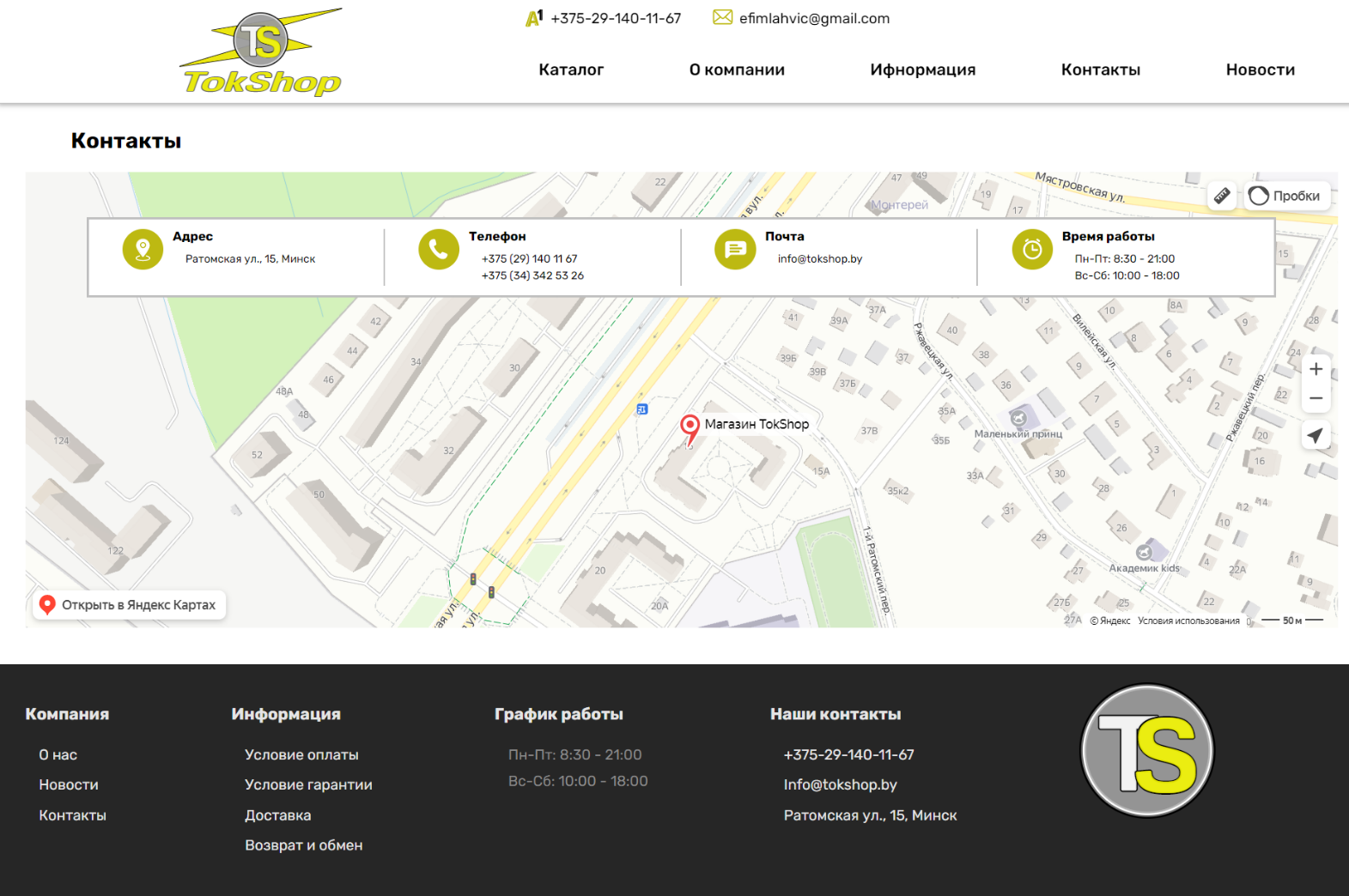


Рисунок 2.3 – Карта местности

## 2.6 Разработка спецэффектов.

На сайте будут разработаны некоторые динамические эффекты и анимации, которые позволяют придать сайту интерактивность и сделают его более привлекательным для пользователя.

На всех страницах, в шапке и в подвале сайта при наведении будет меняться цвет с помощью псевдокласса :hover, который позволяет добавить или изменить какие-либо стили элементов при наведении.

## 2.7 Вывод.

На данном этапе были созданы прототипы, 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). Был определен стиль оформления сайта.

# **3. Реализация структуры веб-сайта**

## 3.1. Структура HTML-документа.

Структура HTML-документа выполнена по стандарту HTML-5. В листинге 3.1 представлена декларация документа и основное содержимое тега head.

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Каталог</title>

<link rel="stylesheet" href="CSS/main.css">

<link rel="icon" href="SVG/logoicon.svg" type="image/svg+xml">

</head>

Листинг 3.1 – Декларация и тег head

Элемент <!DOCTYPE> предназначен для указания типа текущего документа.  Атрибут lang тега html устанавливает язык документа. Метатеги, расположенные в теге head устанавливают текущую кодировку, область просмотра и её начальное масштабирование. Тег title устанавливает название страницы. В тегах link происходит подключение каскадных таблиц стилей.

Тег <body></body>, следующий за <head>, содержит все элементы веб-сайта. Структура тела документа состоит из семантических тегов, таких как nav, header, main, aside и footer. Они помогают компьютерным программам понимать тип информации, хранящейся в тегах. В теге nav содержится навигационное меню, представляющее собой маркированный список с ссылками на основные страницы сайта. Код представлен на листинге 3.2.

<ul class="menu\_\_body">

<li class="menu\_\_link"><a href="index.html" class="link">Каталог</a></li>

<li class="menu\_\_link"><a href="compani.html" class="link">О компании</a></li>

<li class="menu\_\_link"><a href="info.html" class="link">Ифнормация</a></li>

<li class="menu\_\_link"><a href="contact.html" class="link">Контакты</a></li>

<li class="menu\_\_link"><a href="" class="link">Вход</a></li>

</ul>

Листинг 3.2 – Навигационная панель

В теге header располагается логотип сайта, который выполнен с помощью SVG, а также контактные данные и навигационное меню. Листинг в приложении 8.

Шапка сайта является адаптированной под мобильные устройства. При достижении определенных размеров экрана браузера, меню скрывается, а на его месте появляется меню-«бургер», содержащее ссылки как на основные разделы веб-сайта, так и на элементы каталога. Оно раскрывается при нажатии на меню-бургер. При повторном нажатии меню закрывается. Код представлен в листинге 3.3.

div id="burger" class="menu\_\_burger">

<span></span>

</div>

<nav class="menu">

<ul class="menu\_\_body">

<li class="menu\_\_link"><a href="index.html" class="link">Каталог</a></li>

<li class="menu\_\_link"><a href="compani.html" class="link">О компании</a></li>

<li class="menu\_\_link"><a href="info.html" class="link">Ифнормация</a></li>

<li class="menu\_\_link"><a href="contact.html" class="link">Контакты</a></li>

<li class="menu\_\_link"><a href="" class="link">Вход</a></li>

</ul>

</nav>

Листинг 3.3 – Мобильное меню

В теге footer находятся логотип svg, режим работы и ссылка на основную страницу сайта. Теги header и footer имеют отличия для каждой страницы сайта.

Содержимое тега main главной страницы, страниц «Католог», «О нас», «Информация», «Контакты», «Новости» разное.

## 3.2. Добавление таблиц стилей Scss и CSS.

Все css и scss файлы были помещены в отдельную папку css для удобства последующей работы с ними. Подключение стилей осуществлялось внешне с помощью тега link. В таблицах стилей используются селекторы по классу, по идентификатору, по атрибуту, а также селекторы потомков, псевдоклассы и псевдоэлементы. Также, при разработке таблиц стилей использовался препроцессор Scss. Особенности, реализованные в Scss, упрощают работу при создании таблиц стилей. Например, вложенность правил и медиа-запросов, а также переменные.

Пример использования Scss при стилизации первого навигационного меню представлен в листинге 3.4. Реализация второго навигационного меню, мобильного меню и подвала сайта представлены в приложениях 4, 5.

.header {

border-bottom: solid 1px #c0c0c0;

box-shadow: 0 0 5px 2px rgb(181, 181, 181);

background-color: $bc-cl;

width: 100%;

position:fixed;

z-index: 10;

}

.content {

display: flex;

}

.column {

&\_\_logo {

flex: 1 0 200px;

display: grid;

align-items: center;

justify-content: center;

@media (max-width: 767.99px) {

display: block;

}

img{

@media (max-width: 767.99px) {

width: 90px;

}

margin: 6px 15px 0px;

width: 180px;

}

}

&\_\_second-header{

margin: 0;

flex: 1 0 auto;

}

}

.contacts {

@media (max-width: 767.99px) {

display: none;

}

a{

display: inline-block;

margin: 10px 0 10px 30px;

padding-left: 30px;

transition: all 0.5s ease;

&:hover{

color:#6d6d6d;

}

}

.hed-tel {

background: url("../SVG/a1.svg") no-repeat -2px -6px;

}

.hed-email {

background: url("../SVG/message.svg") no-repeat -2px -4px;

}

}

.menu {

@media (max-width:767.99px) {

display: none;

}

&\_\_body {

display: flex;

padding: 0;

margin: 10px -10px;

vertical-align: middle;

}

&\_\_link {

flex: 1 0 auto;

display: inline-block;

margin: 0 5px;

.link {

display: block;

padding: 10px;

font-weight: 500;

font-size: 18px;

background-color: $bc-cl;

border-radius: 10px;

text-align: center;

transition: border-color 0.5s, background-color 0.5s;

@media(min-width: 767.99px){

&:hover{

background-color: #e6e5e5;

}

}

}

}

}

Листинг 3.4 – Использование Scss

В листинге 3.5 представлен сгенерированный CSS-код.

.header {

  border-bottom: solid 1px #c0c0c0;

  box-shadow: 0 0 5px 2px rgb(181, 181, 181);

  background-color: #ffffff;

  width: 100%;

  position: fixed;

  z-index: 10;

}

.content {

  display: flex;

}

.column\_\_logo {

  flex: 1 0 200px;

  display: grid;

  align-items: center;

  justify-content: center;

}

@media (max-width: 767.99px) {

  .column\_\_logo {

    display: block;

  }

}

.column\_\_logo img {

  margin: 6px 15px 0px;

  width: 180px;

}

@media (max-width: 767.99px) {

  .column\_\_logo img {

    width: 90px;

  }

}

.column\_\_second-header {

  margin: 0;

  flex: 1 0 auto;

}

@media (max-width: 767.99px) {

  .contacts {

    display: none;

  }

}

.contacts a {

  display: inline-block;

  margin: 10px 0 10px 30px;

  padding-left: 30px;

  transition: all 0.5s ease;

}

.contacts a:hover {

  color: #6d6d6d;

}

.contacts .hed-tel {

  background: url("../SVG/a1.svg") no-repeat -2px -6px;

}

.contacts .hed-email {

  background: url("../SVG/message.svg") no-repeat -2px -4px;

}

@media (max-width: 767.99px) {

  .menu {

    display: none;

  }

}

.menu\_\_body {

display: flex;

  padding: 0;

  margin: 10px -10px;

  vertical-align: middle;

}

.menu\_\_link {

  flex: 1 0 auto;

  display: inline-block;

  margin: 0 5px;

}

.menu\_\_link .link {

  display: block;

  padding: 10px;

  font-weight: 500;

  font-size: 18px;

  background-color: #ffffff;

  border-radius: 10px;

  text-align: center;

  transition: border-color 0.5s, background-color 0.5s;

}

@media (min-width: 767.99px) {

  .menu\_\_link .link:hover {

    background-color: #e6e5e5;

  }

}

Листинг 3.5 – Сгенерированный CSS-код

Динамические эффекты на страницах были реализованы с помощью трансформаций и псевдоклассов. Пример реализации при наведении представлен в листинге 3.8.

.menu\_\_link .link {

  display: block;

  padding: 10px;

  font-weight: 500;

  font-size: 18px;

  background-color: #ffffff;

  border-radius: 10px;

  text-align: center;

  transition: background-color 0.5s;

}

@media (min-width: 767.99px) {

  .menu\_\_link .link:hover {

    background-color: #e6e5e5;

  }

}

Листинг 3.6 – Анимация элементов меню

При наведении курсором мыши на элементы меню появляется анимация. Код представлен в приложениях 4, 5.

## 3.3. Использование стандартов XML (SVG)

В курсовом проекте для хранения данных использовался XML. Эти данные удобно хранить отдельно в XML-документе. В проекте в XML-файле хранятся пути к изображениям, названия товаров, их цены, скидки, наличие и описание. Листинг предоставлен в приложении 7.

Масштабируемая векторная графика (SVG) использовалась в шапке сайта, логотипе и подвале, а также в странице контактов. Листинг логотипа смотреть в приложении 8. Реализация иконки A1 представлена в листинге 3.7.

<svg width="27" height="30" viewBox="0 0 27 30" fill="none" xmlns="http://www.w3.org/2000/svg">

<path d="M11.5452 7.67407L14.607 8.32707C14.6853 8.36915 14.7244 8.40978 14.7244 8.45041L18.8703 23.1661C18.9108 23.2067 18.9108

23.2473 18.8703 23.2894C18.8325 23.33 18.7933 23.3707 18.7542 23.3707L16.9722 23.6159L16.1959 23.7378H16.1595L14.8797 23.4926C14.8027 23.4519 14.7636 23.4113 14.7636 23.3707L14.2992 21.3261L15.3454 21.2854L15.9259 23.288C15.9259 23.3707 16.0434 23.4519 16.1203 23.4519H16.1973C16.3147 23.4113 16.3917 23.2894 16.3525 23.1661L15.7329 20.9996C15.7124 20.9516 15.6797 20.9108 15.6387 20.8818C15.5976 20.8528 15.5498 20.8368 15.5007 20.8356L12.3214 20.9996L10.4611 21.0808L9.57013 24.4735C9.57013 24.5562 9.49318 24.5969 9.45403 24.5969L5.80903 25.0873H5.73208L4.14313 24.311C4.06483 24.2704 4.02703 24.1876 4.06483 24.1064L9.33658 8.69565C9.33658 8.65502 9.37573 8.61439 9.41353 8.61439L11.0808 7.75533L5.80903 24.4343C5.77123 24.5577 5.84818 24.681 5.96428 24.7202H6.04123C6.11953 24.7202 6.19648 24.6389 6.23563 24.5577L11.5452 7.67407ZM12.9019 11.4353C12.8628 11.4353 12.7467 11.5166 12.7089 11.5979L10.8094 17.9334C10.8094 18.0161 10.8094 18.0974 10.8486 18.138C10.9255 18.1786 10.9647 18.2207 11.0416 18.2207H14.7244C14.7959 18.2188 14.8644 18.1894 14.9175 18.138C14.9566 18.0974 14.9944 18.0161 14.8783 17.9348L13.0963 11.5979C13.0963 11.5166 12.9802 11.4353 12.9019 11.4353ZM12.9802 12.4569L14.4517 17.7709H13.4838L12.5145 14.0096L12.9802 12.4569Z" fill="#BDB912"/>

<path d="M22.4289 4.77197L22.95 5.4032V5.57879L22.9162 10.7781L22.9108 11.6096L22.8757 17.369L22.8487 17.3879L22.279 17.8319L19.822 17.5794V8.77412C19.5979 9.02661 19.2631 9.32119 18.8905 9.65929C18.3316 10.1236 17.8105 10.4589 17.55 10.5851V8.31267C17.55 8.26913 17.5878 8.26913 17.6242 8.2285C17.6985 8.18497 18.5179 7.80768 19.1511 7.04876C19.7856 6.28983 19.8963 6.07942 20.0448 5.6586C20.2311 5.23778 20.2311 5.068 20.2311 5.068C20.2311 5.068 20.2311 5.02592 20.2689 5.02592H20.3053L21.982 4.77197V17.2428C21.982 17.4111 22.0941 17.4953 22.2048 17.4953C22.3533 17.4953 22.4289 17.369 22.4289 17.2428V4.77197ZM19.3725 9.94226V10.7113C19.336 10.7549 19.2618 10.7549 19.224 10.797C19.0755 10.8811 18.5544 11.1336 18.5544 11.1336L17.7714 11.0495C18.2714 10.7762 18.7457 10.4516 19.1875 10.0801C19.2834 10.0238 19.3711 9.95296 19.4481 9.8697L19.3725 9.94226Z" fill="black"/>

</svg>

Листинг 3.7 – Значок A1



Листинг 3.7 – Результат SVG

## 3.3. Управление элементами DOM

DOM – это представление веб-страницы в виде древовидной структуры объектов, которыми можно манипулировать с помощью JavaScript. Управление элементами DOM позволяет динамически изменять содержимое страницы без перезагрузки страницы.

В моём проекте JavaScript использовался для трёх целей:

1. Создание динамического бургер-меню для устройств с шириной экрана меньше, чем 768px;
2. Добавление ЯндексКарта;
3. Чтение информации из XML файла и добавление в документ HTML;
4. Добавление и извлечение параметров GET запросов в адресной строке;
5. Переключение информационных полей (описание, характеристики) в странице просмотра одного товара.

Пример использование JavaScript при переключении информационных полей в странице просмотра одного товара. Полный листинг JavaScript предоставлен в приложении 6.

let button\_op = document.querySelector(".info\_\_op");

let button\_ch = document.querySelector(".info\_\_ch");

let content\_op = document.querySelector(".content-info\_\_opisanie");

let content\_ch = document.querySelector(".content-info\_\_characteristic");

    document.querySelectorAll("#but").forEach((button) => {

    button.addEventListener('click', function() {

        let classname = this.getAttribute('value');

        let info = document.querySelector(classname);

if((info.style.display == "none")) {

            if (this.parentElement.classList.contains('info\_\_op')) {

                content\_op.style.display = "block";

                content\_ch.style.display = "none";

            } else {

                content\_op.style.display = "none";

                content\_ch.style.display = "block";

            }

            button\_op.classList.toggle('active');

            button\_ch.classList.toggle('active');

            content\_op.classList.toggle('active');

            content\_ch.classList.toggle('active');

            }

        });

    });

## 3.5 Выводы

На текущем этапе была создана HTML-структура, разработаны таблицы стилей на основе SCSS/CSS, а также продемонстрировано использование стандартов XML (SVG) и языка программирования JavaScript, который был применен для создания различных динамических объектов.

# **4. Тестирование веб-сайта**

## 4.1. Адаптивный дизайн веб-сайта

Адаптивный дизайн веб-сайта – это метод создания сайтов, который позволяет оптимизировать веб-страницу для просмотра на различных устройствах с разными размерами экранов. С помощью адаптивного дизайна сайт адаптируется под любое устройство, будь то настольный компьютер, планшет или мобильный телефон. Адаптивный дизайн позволяет улучшить пользовательский опыт, увеличить удобство использования и повысить конверсию сайта.

Для создания адаптивного дизайна используются различные технологии и инструменты, такие как медиа-запросы CSS, гибкие сетки и изображения, а также JavaScript. Каждый элемент сайта должен быть адаптирован под разные размеры экрана, включая шрифты, изображения, видео, кнопки и меню. Важно убедиться, что сайт быстро загружается на любом устройстве, чтобы пользователи не уходили на другой сайт из-за длительного времени загрузки.

Результатом использования адаптивного дизайна является сайт, который выглядит и работает хорошо на любом устройстве, удобен для пользователей и позволяет достигнуть поставленных целей веб-сайта. Это важный аспект веб-разработки, который помогает улучшить взаимодействие с пользователем и повысить уровень удовлетворенности пользователей веб-сайтом.

Адаптивный дизайн в основном создавался с помощью медиа-запросов. Медиа-запросы назначают страницам стили на основе размера окна браузера. Использовались медиа запросы под разные размеры, например для шапки и подвала использовались размеры 768px, 1000px, для элементов товаров 440px и 768px, и много других медиа-запросов, на которых веб-страницы отображались не так, как это предполагалось. Пример адаптивности главной страницы под различные устройства на рисунке 4.1.

## 4.2 Кроссбраузерность веб-сайта

Кроссбраузерность является одним из важных аспектов при разработке веб-сайтов, так как существует множество браузеров и устройств, которые могут иметь различную интерпретацию HTML, CSS и JavaScript. В связи с этим, для того чтобы веб-сайт корректно отображался на всех устройствах и в браузерах, необходимо уделить внимание кроссбраузерности.

Основными принципами кроссбраузерной разработки являются использование стандартных технологий и правильное использование CSS. Для этого необходимо следить за тем, чтобы веб-сайт был написан в соответствии с современными стандартами HTML и CSS, а также использовать префиксы для свойств CSS, чтобы обеспечить поддержку всех браузеров.

## 4.3. Руководство пользователя

Главная страница является точкой входа на веб-сайт. Вверху страницы располагается горизонтальная навигационная панель в компьютерной версии или выпадающее меню-«бургер» в мобильной версии. Она содержит ссылки на основные разделы сайта, такие как «Каталог», «О нас», «Информация», «Контакты», и ссылку на страницу «Новости». Нажав на ссылку на панели навигации, пользователь попадает на соответствующую страницу веб-сайта.

Внизу страницы также располагается смешанная навигационная панель в компьютерной версии: разделы размещены горизонтально, а их подразделы – вертикально; или вертикальная панель в мобильной версии. Она содержит разделы «Компания», «Информация», «Режим работы» и «Контакты». Их подразделы содержат ссылки на соответствующие страницы веб-сайта, кроме раздела «Режим работы», который содержит только информационную часть о режиме работы.

На странице «Каталог» находится информация о каталоге товаров. Нажав на определенный каталог, пользователя перенесет на страницу соответствующего каталога товаров, где он сможет увидеть содержание товаров одного выбранного пользователем каталога. Слева сбоку будет размещен блок «Каталог товаров», чтобы пользователю было удобнее перемещаться по каталогам.

Нажав на данной странице на кнопку «Купить», пользователь попадет на страницу просмотра одного товара. На этой странице пользователь может увидеть полную информацию о товаре: описание, наличие, характеристики. Также он может увидеть, что под *что-то* располагается две кнопки «Описание» и «Характеристики», которые позволяют переключать информацию на соответствующие разделы. При загрузке страниц автоматически будет включено раздел «Описание».

Нажав в странице просмотра одного товара на кнопку «Купить», пользователя перенесет на страницу форма заказов, где он должен внести свои личные данные для оформления заказа. После чего, нажав на «Оформить заказ», пользователя возвращает на страницу «Каталог» и появляется уведомление, что он успешно оформил заказ.

Станица «О нас» содержит информацию о магазине электротоваров TokShop.

Станица «Информация» содержит информацию об условиях доставки и оплаты, о гарантии, возврате и обмену товаров.

Страница «Контакты» содержит контактную информацию, такую как адреса магазинов, номера телефонов, карту расположения магазинов.

Страница «Новости» размещается актуальная информация на определенный момент времени: о проведении распродаж, акций, об изменении в режиме работы магазина и т.д..

# **Заключение**

В ходе курсового проектирования был создан веб-сайт электротоваров. Он был создан таким образом, чтобы отвечать всем требованиям пользователя. В процессе разработки были проанализированы веб-сайты с подобной тематикой, подобраны качественные фотоматериалы.

Были реализованы следующие задачи:

1. Анализ аналогичных решений.
2. Выбор способа верстки.
3. Выбор стилевого оформления.
4. Разработка логотипа.
5. Разработка пользовательских элементов.
6. Разработка спецэффектов.
7. Добавление веб-сайту адаптивности.
8. Кроссбраузерность веб-сайта.
9. Тестирование веб-сайта.
10. Создание руководства пользователя.

При написании данного курсового проекта были проанализированы и использованы существующие языки и редакторы создания веб-сайта, рассмотрены источники по теме работы, разработано руководство пользователя.

Результатом курсового проекта стал веб-сайт, предоставляющий пользователю всю необходимую информацию.

# **Список использованных литературных источников.**

1. Справочник по HTML / Html book – режим доступа: http://htmlbook.ru – Дата доступа: 20.03.2023.
2. CSS [Электронный ресурс]. – Режим доступа: https://developer.mozilla.org/en-US/docs/Web/CSS. – Дата доступа: 15.03.2023.
3. 220SHOP.by – режим доступа: https://220shop.by/product\_list – Дата доступа: 20.03.2023.
4. Подбор шрифта для веб-сайта [Электронный ресурс]. – Режим доступа: https://html5book.ru/shrift-dlya-sita. – Дата доступа: 11.03.2023.
5. Руководство по графическому редактору Figma [Электронный ресурс]. – Режим доступа: https://vc.ru/design/184941-figma-vse-chto-vam-nuzhno-znat. Дата доступа: 19.02.2022.
6. Flexbox [Электронный ресурс]. – Режим доступа: https://developer.mozilla.org/ru/docs/Learn/CSS/CSS\_layout/Flexbox. – Дата доступа: 15.03.2022.