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

Факультет информационных технологий

Кафедра информационных систем и технологий

Специальность 1-40 05 01 «Информационные системы и технологии»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА КУРСОВОГО ПРОЕКТА**

по дисциплине «Скриптовые языки программирования»

Тема: Web-сайт ««Интернет-сервис по продажи автомобильной резины и дисков»

**Исполнитель**

студент 2 курса 1 группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С. Л. Скалкович

подпись, дата

**Руководитель**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н. А. Жиляк

должность, учен. степень, ученое звание подпись, дата

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

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

дата, подпись

Курсовой проект защищен с оценкой

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н. А. Жиляк

подпись дата инициалы и фамилия

**Содержание**

[Введение 3](#_Toc121744205)

[1. Постановка задачи 4](#_Toc121744206)

[2. Проектирование веб-сайта 5](#_Toc121744207)

[2.1 Обзор аналогичных решений 5](#_Toc121744208)

[2.2 Выбор средств реализации программного продукта 8](#_Toc121744209)

[2.3 Проектирование страниц 9](#_Toc121744210)

[2.4 Выбор стилевого оформления 11](#_Toc121744211)

[2.5 Разработка пользовательских элементов 11](#_Toc121744212)

[2.6 Выводы 14](#_Toc121744213)

[3. Реализация структуры веб-сайта 15](#_Toc121744214)

[3.1 Структура JSX-документа 15](#_Toc121744215)

[3.2 Структура Redux-Toolkit 19](#_Toc121744216)

[3.3 Добавление таблиц стилей Sass 20](#_Toc121744217)

[3.4 Выводы 21](#_Toc121744218)

[4. Тестирование веб-сайта 22](#_Toc121744219)

[4.1 Адаптивный дизайн веб-сайта 22](#_Toc121744220)

[4.2 Кроссбраузерность веб-сайта 23](#_Toc121744221)

[4.3 Тестирование функционала сайта 25](#_Toc121744222)

[4.4 Выводы 25](#_Toc121744223)

[Заключение 26](#_Toc121744224)

[Список использованных источников 27](#_Toc121744225)

[Приложение 28](#_Toc121744226)

[Приложение А Макет структуры веб-сайта 28](#_Toc121744227)

[Приложение Б Листинг JSX-документа 30](#_Toc121744228)

[Приложение В Листинг (SASS) 38](#_Toc121744229)

[Приложение Г Листинг Redux 43](#_Toc121744230)

# Введение

В настоящее время количество новых сайтов увеличивается с неимоверной скоростью. Именно поэтому, основной задачей, ставящейся при разработке сайта – сделать его максимально привлекательным для потенциальных пользователей и придать ему индивидуальность. На заре развития интернета можно было привлечь достаточно большое количество посетителей на сайт, не прикладывая к его разработке больших усилий и без особых материальных затрат. Сейчас же, в условиях возросшей конкуренции, функциональность сайта, его стиль и запоминаемость играют очень большую роль. Ведь независимо от того создается сайт для представления какой-либо фирмы или просто для заработка на рекламе, основной его функцией является привлечение как можно большей аудитории. Поэтому для достижения успеха важен правильный подход к выбору стратегии и исполнителей.

Профессиональные студии веб – дизайна предлагают создание сайтов различного предназначения, это может быть корпоративный сайт, сайт – визитка, интернет – магазин, сайты для ведения бизнеса как онлайн, так и в реальном мире.

Наше время называют «информационным веком». Это название возникло потому, что самым важным, ценным и необходимым ресурсом является информация. Обладание информацией и умение своевременно, быстро и четко донести необходимую информацию до клиентов – ключ к успеху в реалиях современного бизнеса. В настоящий момент именно Интернет, способен оперативно и массово передавать текст, звук, изображения и даже видео-ролики, т.е. практически любую информацию. C точки зрения пользователя, Интернет – это огромный информационный ресурс, в котором можно найти все, что угодно: от прогноза погоды до личных предпочтений голливудских кинозвезд. Аудитория сети Интернет растет с каждым годом. Интернет – самое быстроразвивающееся средство передачи информации за все историю человечества. Пользователи сети интернет являются целью особого интереса для рекламодателей, потому что факт остается фактом – среди пользователей сети куча потенциальных клиентов.

Цель курсовой работы: разработать интернет-магазин с использованием React JS, а также с применением Sass/CSS3.

Задачи:

* Проанализировать и описать основные принципы работы React;
* Разработать приложение, используя React;
* Провести тестирование приложения;
* Провести анализ и оценку качества приложения.;

# 1. Постановка задачи

Интернет-сервис должен быть реализован в минималистичном стиле, чтобы быть удобным и привлекательным для пользователя. Для этого стоит использовать неширокий спектр цветов, преимущественно белый и серый. Для привлечения пользователя к отдельным элементам интерфейса необходимо использовать более яркие, бросающиеся глазу цвета: зеленый, синий, оранжевый.

Структура интернет-сервиса должен состоять из веб-страниц, таких как: главная, каталог, корзина, избранное, страница товара, а также вспомогательных информационных веб-страниц: о нас, оплата, доставка, контакты, форма заказа. Каждая веб-страница должна придерживаться данной структуры: верхний колонтитул, главный контент, нижний колонтитул.

Для реализации интеренет-сервиса необходимо применить современные технологии верстки. Верстка интернет-сервиса должна быть адаптивной для устройств с любой шириной экранов, а также быть кроссбраузерной, поддерживаясь современных браузерах.

# 2. Проектирование веб-сайта

## 2.1 Обзор аналогичных решений

Веб-сайт – это набор взаимосвязанных веб-страниц, которые представляют собой информацию, предназначенную для просмотра в браузере. Веб-сайт представляет собой совокупность веб-страниц, которые могут быть связаны между собой посредством гипертекстовых ссылок.

Для создания быстрого и динамического веб-сайта необходимо использовать современные технологии, которые позволяют создавать веб-сайт с минимальными затратами времени и средств. Веб-сайт должен быть удобным для пользователя, а также должен быть адаптирован под мобильные устройства. Тема веб-сайта должна быть востребованной, а также должна быть актуальной. Поэтому было принято решение разработать интернет-магазин, который будет представлять собой веб-сайт, на котором можно будет купить различные товары. Прежде чем приступить к разработке веб-сайта, необходимо проанализировать аналоги, чтобы понять, какие технологии используются для создания веб-сайтов, а также какие есть недостатки и преимущества веб-сайтов аналогов. И самое главное найти в каждом аналоге свои уникальные особенности, которые будут отличать этот сайт от других. Для этого будет рассмотрены 3 аналога.

Аналог №1 – сайт autoset (https://www. autoset.by/).

Основным преимуществом данного сайта является его удобство и простота использования. На главной странице сайта можно увидеть разделы, которые помогут пользователю найти нужный товар. На рисунке 1.1 представлено оформление сайта. Также все товары разделены по категориям, что позволяет пользователю найти нужный товар быстрее. Веб-сайт очень простой и удобный для пользователя, а также имеет красивый дизайн. Шрифт читабельный. Сайт динамический, интерактивный. Также есть анимация при наведении на картинки. Сайт использует два основных цвета (черный и белый), что положительно сказывается на общем восприятии сайта. На этом сайте есть уникальная функция, которая позволяет пользователю сравнить товары, которые он хочет купить. Еще можно отметить, что на сайте есть режим для слабовидящих, который позволяет пользователю увеличить шрифт и контрастность. Также можно добавить товары в избранное, чтобы не потерять их, причем без регистрации.

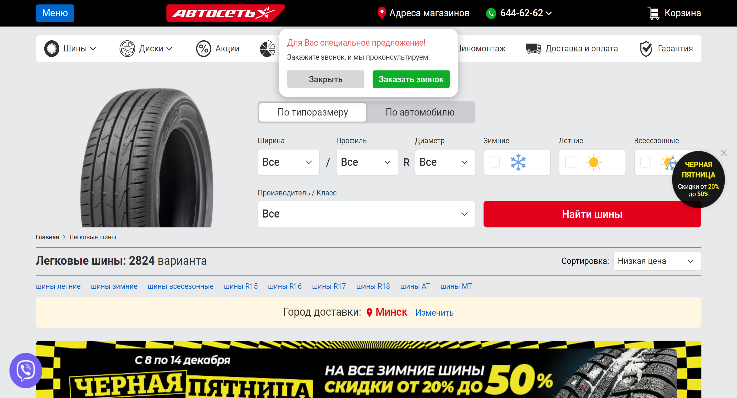


Рисунок 1.1 – Главная страница сайта autoset

Веб-сайт имеет адаптивный дизайн, который позволяет использовать сайт на мобильных устройствах, горизонтальное меню становится выезжающим, текст остаётся читабельным. Дизайн представлен на рисунке 1.2.

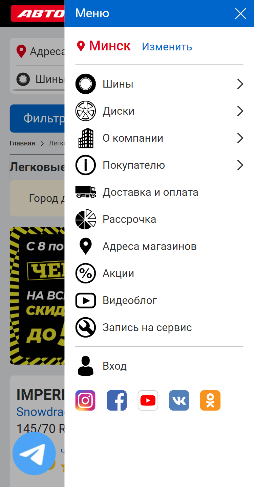


Рисунок 1.2 – Мобильная версия сайта autoset

Из минусов данного веб-сайта можно выделить то, что он долго загружается из-за большого количества товаров, которые находятся на сайте. Данный веб-сайт был разработан с помощью HTML, CSS, JavaScript, PHP, MySQL.

Аналог №2 – сайт Kolesa-darom (https://www.kolesa-darom.ru/).

Следующим аналогом будет сайт Kolesa-darom, который является одним из крупнейших интернет-сервисов по продаже автомобильной резины и дисков в России. На сайте есть разделы, в которых можно выбрать нужный товар. Веб-сайт очень удобен для пользователя, так как он имеет простой и понятный интерфейс. Также пользователь может быстро найти нужный товар, так как на сайте есть поиск по названию товара и по категориям. Все товары имеют подробное описание, также есть отзывы о товаре, которые позволяют пользователю сделать правильный выбор. Дизайн представлен на рисунке 1.3. Одной из особенностей сайта является то, что сайт будет показывать товары, которые пользователь просматривал ранее, таким образом пользователь сможет быстро найти нужный товар. Еще из особенностей данного сайта является то, что на сайте есть быстрый поиск по названию товара. Отличие данного поиска от обычного поиска в том, что он показывает товары вместе с ценой.

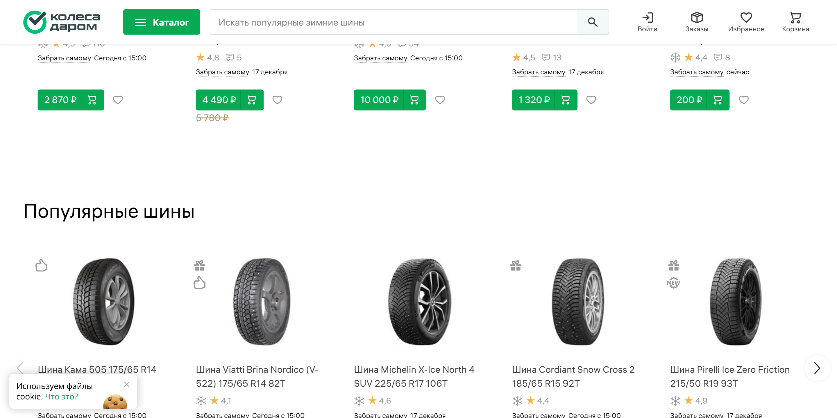


Рисунок 1.3 – Главная страница сайта Kolesa-darom

Также сайт имеет адаптивный дизайн, который позволяет просматривать сайт на мобильных устройствах. У данного сайта нет никаких проблем с адаптивностью, на разных устройствах сайт выглядит привлекательно. На рисунке 1.4 представлено оформление на мобильных устройствах.

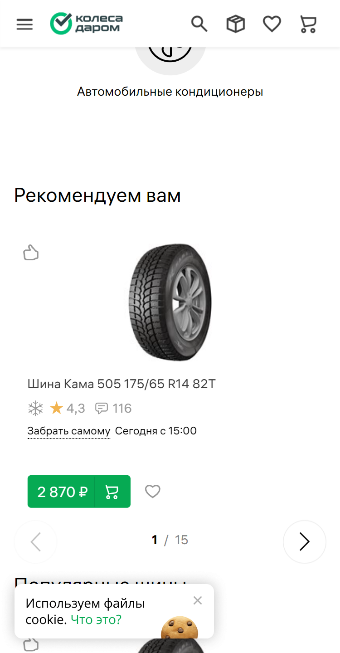


Рисунок 1.4 – Мобильная версия сайта Kolesa-darom

Из недостатков можно выделить то, что сайт имеет много рекламы, которая может отвлекать пользователя от основного контента. Данный сайт был разработан на таких технологи как HTML, CSS, JavaScript, PHP, MySQL.

Аналог №3 – сайт Michelin (https://www.michelinman.com/).

Michelin – сайт, который является крупнейшим интернет-сервисом в мире по продаже премиальной автомобильной резины. В основном на сайте используется белый цвет. Также данный сайт имеет минималистичный дизайн и простой интерфейс, данные стилистические решения предают премиальности продаваемому товару, бренду и сайту в целом.

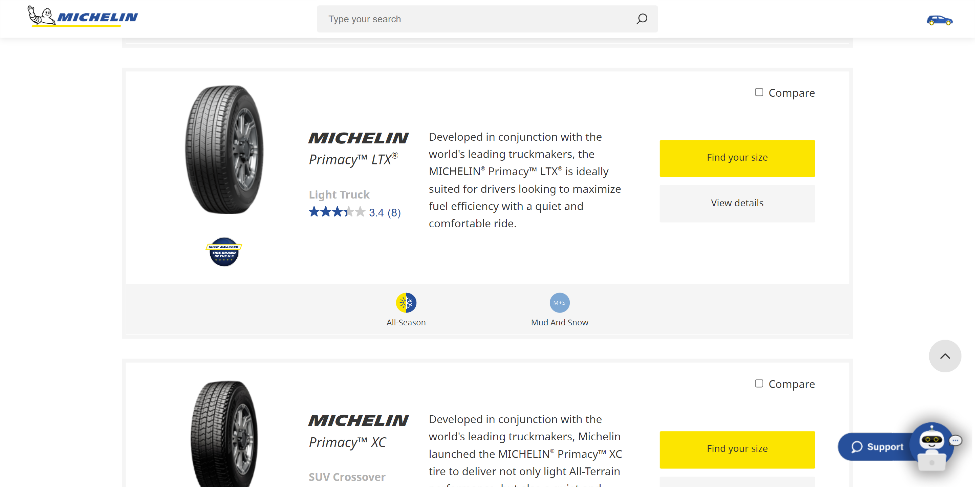


Рисунок 1.5 – Главная страница сайта Michelin

Также данный сайт кроссплатформенный, а также адаптирован под мобильные устройства. На рисунке 1.6 представлено оформление на мобильных устройствах.

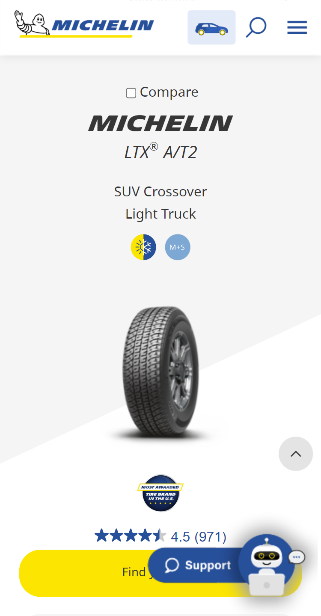


Рисунок 1.6 – Мобильная версия сайта Michelin

И таким образом, было проанализировано 3 сайта, которые являются аналогами нашего будущего сайта. Основные преимущества и недостатки этих сайтов были выделены, а также были выделены технологии, которые используются для создания сайтов. Можно сделать вывод, что сайт, который будет создан в рамках данной работы, будет иметь следующие преимущества:

1. Быстрая загрузка сайта
2. Простой и удобный дизайн
3. Спокойные постельные цвета
4. Кроссплатформенность
5. Адаптивность под мобильные устройства
6. Удобная навигация по сайту

## 2.2 Выбор средств реализации программного продукта

Главной задачей сайта является предоставление информации о товарах, которые можно купить. Данный интернет-магазин будет предназначен для людей, которые хотят купить автомобильную резину и диски, а также товары, связанные с обслуживанием вышеперечисленных товаров. Сайт будет адаптирован под мобильные устройства, а также будет кроссплатформенным для удобства его использования широко круга пользователей.

Интернет-магазин будет разработан в текстовом редакторе Visual Studio Code. Страницы сайта могут просматриваться в таких браузерах как Microsoft Edge, Safari, Opera, Mozilla Firefox, Google Chrome.

Visual Studio Code — редактор исходного кода, разработанный Microsoft для Windows, Linux и macOS. Позиционируется как «лёгкий» редактор кода для кроссплатформенной разработки веб- и облачных приложений. Включает в себя отладчик, инструменты для работы с Git, подсветку синтаксиса и средства для рефакторинга, а также широкий выбор плагинов для оптимизации разработки.

Визуальная часть интернет-сервиса, в данном случае это внешний вид сайта, его дизайн, а также навигация по сайту, является одной из важнейшей частей интернет-сервиса, так как она является лицом системы, и при неправильно реализованной реализации, может отвлечь от интернет-сервиса потенциальных клиентов. В данном случае для реализации визуальной части, будут использованы следующие технологии: React JS, SASS, Ant Design, React Router, Redux Toolkit.

JavaScript – это язык программирования, который используется для создания динамических веб-страниц. JavaScript позволяет создавать интерактивные веб-страницы, которые могут реагировать на действия пользователя. Также JavaScript является основным языком для разработки функциональных React компонентов и налаживания их взаимодействия и создания функционала.

React JS — это библиотека JavaScript, предназначенная для разработки пользовательских интерфейсов. В отличие от других библиотек JavaScript React JS использует компонентный подход, который позволяет разрабатывать сложные интерфейсы из простых компонентов. С помощью React разработчики создают веб-приложения, которые изменяют отображение без перезагрузки страницы. Благодаря этому приложения быстро реагируют на действия пользователя, например, заполнение форм, применение фильтров, добавление товаров в корзину и так далее. React применяют для отрисовки компонентов пользовательского интерфейса. Также библиотека может полностью управлять фронтендом. В этом случае React используют с библиотеками для управления состоянием и роутинга, например, Redux и React Router.

SASS — это расширение CSS, которое добавляет в CSS мощные возможности, такие как переменные, вложенные правила, импорт и многое другое. Но самое главное, что SASS позволяет использовать препроцессоры, которые позволяют писать более краткий и лаконичный код.

Redux — это библиотека для управления состоянием приложения.

Redux Toolkit — это набор инструментов для упрощения работы с Redux. Redux Toolkit позволяет создавать более простые и понятные редьюсеры, а также упрощает работу с асинхронными запросами.

React Router — это библиотека для маршрутизации в React-приложениях. React Router позволяет создавать маршруты, которые будут отображаться в зависимости от URL-адреса. Благодаря React Router можно создавать SPA (Single Page Application), которые не перезагружают страницу при переходе по ссылкам.

## 2.3 Проектирование страниц

Для реализации шаблонов верстки страницы будет использована Grid верстка. Она представляет собой двумерную систему сеток в CSS. Гриды подойдут и для верстки основных областей страницы, и небольших элементов пользовательского интерфейса. Также будет использована Flexbox-верстка (Flexbox), так как она позволяет создавать адаптивные веб-страницы, которые будут отображаться корректно на всех устройствах. 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 и CSS Grid поддерживается всеми используемые на сегодняшний момент современными браузерами (с использованием префиксов: 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 шаблоны страниц. Для того чтобы более детально продумать расположение основных элементов и функциональных элементов интернет-сервиса.

Для того чтобы более детально продумать дизайн отдельных элементов было принято создать макет сайта на основе рассмотренных аналогов, сделав макет удобным и легким. По макету будет понятно, где будут располагаться основные блоки, картинки, текст.

На сайте будут представлены следующие страницы:

1. Main (Главная страница);

2. Catalog (Продукты);

3. Basket (Корзина);

4. Favorites (Избранное)

5. Item (Полное описание продукта);

6. About (Описание магазина)

На странице Main будет расположены слайдеры с изображениями, категории товаров, информация о компании, контакты, форма обратной связи.

На странице Catalog будут расположены категории товары, фильтры, карточки товаров, пагинация. Фильтр будет реализован при помощи расположения его составных частей в виде аккордиона. При нажатии на карточку товары, пользователь будет перенаправлен на страницу Item.

На странице Basket будут расположены карточки товары, форма оформления заказа, информация о доставке, информация о оплате.

На странице Favorites будут расположены карточки товары, выбранные пользователем. В данном разделе пользователь может отслеживать цены и наличие выбранных им товары.

На странице Item будут расположены: изображения товара, его цена, описание и характеристики. Также этой странице пользователь сможет более детально ознакомиться с товаром, оставить отзыв и посмотреть отзывы других пользователей.

На странице About будет расположена информация о интернет магазине и его владельце, что поможет пользователю узнать необходимую ему информацию.

Также будет фиксированное навигационное меню с ссылками на главную страницу, страницу продуктов, страницу корзины, страницу профиля, страницу входа, страницу регистрации. Навигационное-меню будет расположено вверху страницы.

## 2.4 Выбор стилевого оформления

Для реализации стилевого оформления сайта было выбрано минималистическое направление с элементами классического веб-дизайна. Такой стиль позволяет привлечь внимание новых пользователей. Сайт не загромождается лишними деталями (боковыми панелями и т.п.), большим количеством спецэффектов. Акцент на сайте ориентируется на основной контент.

Сайт будет состоять из 2-х основных цветов - белого и чёрного. Белый цвет будет использоваться для фона, а чёрный для текста и некоторых элементов дизайна. Эти цвета были подобраны так, потому что они будут приятны для глаз и это цвета, на которое пользователь сразу обратит внимания на сайт.

В данном проекте будут использован один основной шрифт: Monserrat. Шрифт подключался с помощью библиотеки Google Fonts.

Шрифт Monserrat является одним из самых популярных шрифтов в 2019 году. Он был разработан в 2011 году, но в последние годы его популярность растет с каждым днем. Он был вдохновлен плакатами, вывесками и расписными окнами первой половины двадцатого века, которые можно увидеть в историческом районе Монтсеррат в Буэнос-Айресе. Этот шрифт отличается своей читабельностью и простотой. Он подходит для любых типов дизайна, в том числе и для веб-дизайна.

## 2.5 Разработка пользовательских элементов

Разработка логотипа осуществлялась в графическом редакторе Adobe Ilustrator. Дизайн довольно минималистичен. Было принято решение не загромождать его множеством декоративных элементов. Логотип представляет из себя название совмещение первой буквы названия индивидуального предпринимателя и колеса, как отсылающего к продаваемой продукции, представлена на рисунке 1.7.



Рисунок 1.7 – Логотип сайта

В проекте будет использована иконка сайта, которая будет отображаться во вкладке браузера, она представляет собой колесо. Иконка будет иметь формат .ico и будет иметь размер 32x32 пикселя. Иконка разаботана в программе Adobe Photoshop. Иконка представлена на рисунке 1.8.



Рисунок 1.8 – Иконка сайта

В проекте будут пользовательские элементы, которые будут отвечать за взаимодействие пользователя с сайтом. Это будут следующие элементы: кнопки, модальные окна, табы, пагинация, формы и т.д. Пример кнопки будет на рисунке 1.9.



Рисунок 1.9 – Кнопка сайта

Также было разработано адаптивное меню, которое позволяет осуществлять удобную навигацию пользователя по разделам интернет-сервиса. Представлено на рисунке 2.0.

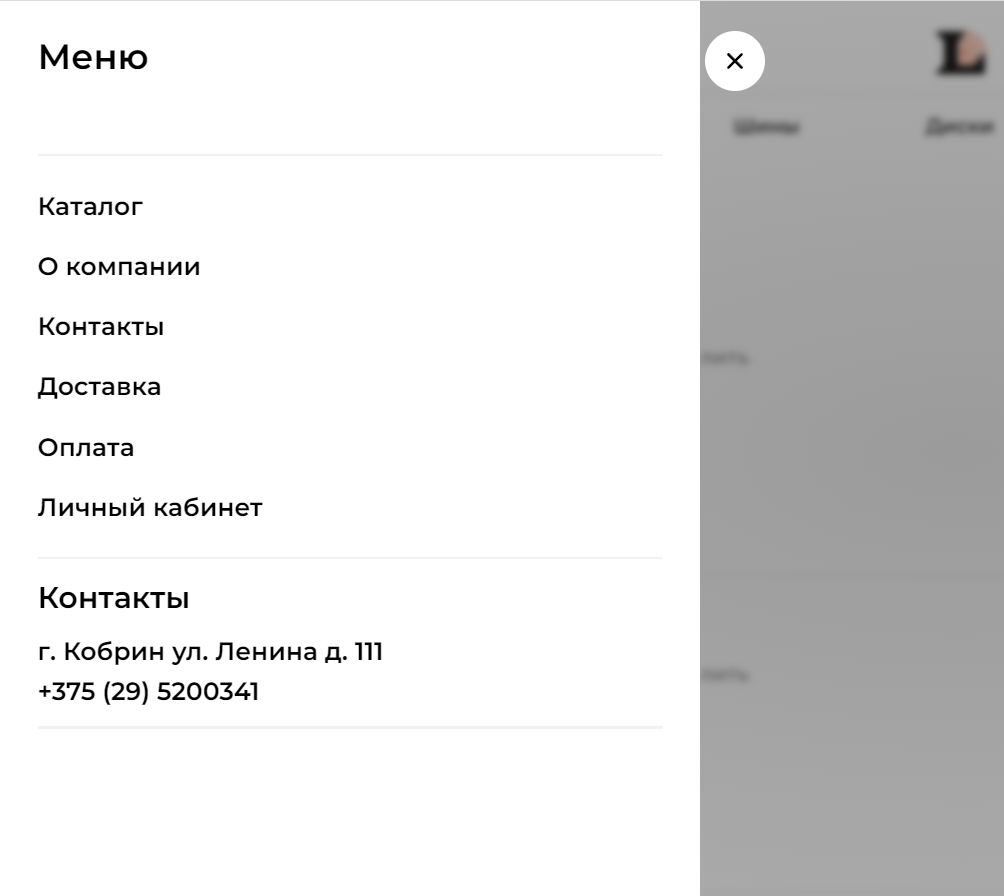


Рисунок 2.0 –адаптивное меню сайта

На сайте будут карточки товара, которые будут состоять из картинки и кнопок «Добавить в корзину» и «Добавить в избранное». Кнопка «Добавить в корзину» будет добавлять товар в корзину. Кнопка «Добавить в избранное» будет добавлять товар визбранное. Дизайн карточек товара будет простым и понятным. Пример карточки товара будет на рисунке 2.1.



Рисунок 2.1 –Карточка товара

Также на сайте, в разделе «Каталог» будет блок с параметрами поиска. Пользователь сможет искать продукты по категории, по цене, по уникальному полю. Пример блока фильтра товара будет рисунке 2.2.

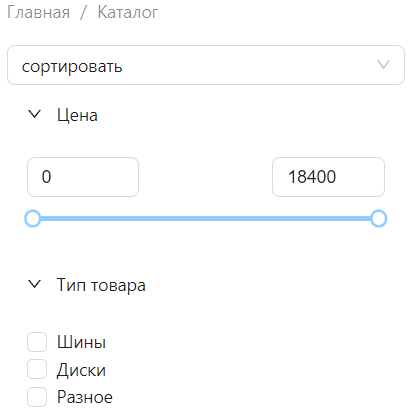


Рисунок 2.2 –Блок фильтра товаров

Следующим основным элементом сайта будет блок корзины. Пользователь сможет добавлять продукты в корзину, удалять их из корзины, оценивать стоимость заказа оформлять заказы. Пример блока корзины будет рисунке 2.3.

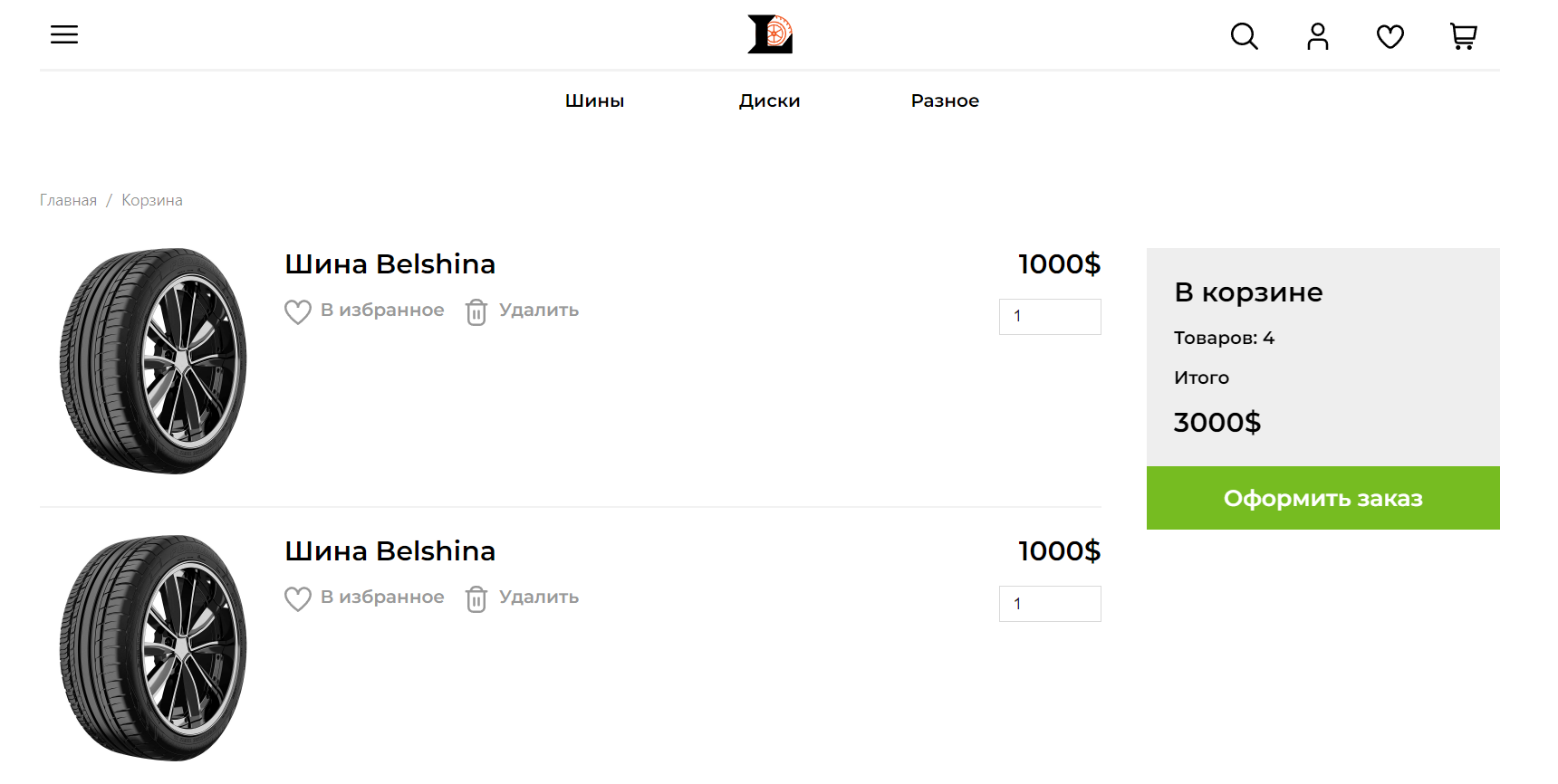


Рисунок 2.3 –Блок корзины

На сайте будут разработаны некоторые динамические эффекты и анимации, которые позволят придать сайту интерактивность и сделают его более привлекательным для пользователя. Анимация будет при появлении страницы на сайте. А также при наведении на кнопки.

Элемент Header, состоит из: логотипа сайта, навигационного меню и кнопки поиска. В навигационном меню 4 пункта: Поиск, Профиль, Избранное, Корзина. Над иконкой корзины и избранного будет висеть круг с количеством товаров, добавленных в соответствующий раздел. На рисунке 2.1 представлено оформление Header.



Рисунок 2.1 – Header сайта

## 2.6 Выводы

Подводя итоги этого раздела можно сказать, что были разработаны макеты сайта, которые будут использоваться для дальнейшей разработки сайта. Были выбраны основные цвета и шрифты, которые будут использоваться на сайте. Были разработаны некоторые динамические эффекты и анимации, которые позволят придать сайту интерактивность и сделают его более привлекательным для пользователя. Самое главное, чтобы сайт был удобным для пользователя и отвечал его потребностям.

# 3. Реализация структуры веб-сайта

## 3.1 Структура JSX-документа

Для реализации проекты было принято решение создавать разные компоненты. Компонент — это функция, которая возращает JSX формат.  JavaScript XML (JSX) — это расширение синтаксиса JavaScript, которое позволяет использовать HTML-подобный синтаксис для описания структуры интерфейса. Как правило, компоненты написаны с использованием JSX, но также есть возможность использования обычного JavaScript. Код примера JSX-формата показан в листинге 1.1.

|  |
| --- |
| import "./Footer.scss";  import { Link } from "react-router-dom";  export default function Footer(props) {    return (      <footer className="footerContainer">        <div className="footerContent">          <ul className="footerContentContainer">            <li>              <Link to="/catalog">Каталог</Link>            </li>            <li>              <Link to="/about">О компании</Link>            </li>            <li>              <Link to="/contacts">Контакты</Link>            </li>            <li>              <Link to="/delivery">Доставка</Link>            </li>            <li>              <Link to="/payment">Оплата</Link>            </li>            <li>              <Link to="/profile">Личный кабинет</Link>            </li>          </ul>        </div>      </footer>    );  } |

Листинг 1.1 – Структура JSX-формата

Самым главным компонентом будет с названием файлом и функции App.tsx. Он будет содержать основные маршруты по сайту. Для этого нужно использовать пакет React-router-dom. Почему лучше использовать его, чем обычный тег <a>? Потому что React это SPA. SPA(single page application) – это веб-приложение, работающее на одной странице. Оно подгружает все необходмые javascript и css файлы при первой загрузке страницы, а затем все общение между клиентом и сервером сводится к минимуму. Т.е. при таком подходе большая часть работы сайта производится на стороне клиента, а если нужно получить данные с сервера, то это обычно делается с помощью JSON. Код примера маршрутов будет показан в листинге 1.2.

|  |
| --- |
| <Router>          <Routes>            <Route path="/" element={<Main />} />            <Route path="/basket" element={<Basket />} />            <Route path="/favorites" element={<Favorites />} />            <Route path="/catalog" element={<Catalog />} />            <Route path="/catalog/item" element={<ItemPage />} />            <Route path="/contacts" element={<Contacts />} />            <Route path="/delivery" element={<Delivery />} />            <Route path="/login" element={<LoginPage />} />            <Route path="/registration" element={<RegistrationPage />} />            <Route path="/profile" element={<Profile />} />            <Route path="/payment" element={<Payment />} />            <Route path="/about" element={<AboutUs />} />            <Route path="/order" element={<OrderPage />} />            <Route path="/error" element={<ErrorPage />} />            <Route path="\*" element={<ErrorPage />} />          </Routes>        </Router> |

Листинг 1.2 – Структура маршрутов

Следующим компонентом будет Card компонент. В нем будет находится карточка товара. Карточка товара будет состоять из:

- фото товара;

- названия товара;

- описания товара;

- цены товара;

- кнопки добавления товара в корзину.

Кнопка добавления товара в корзину будет висеть поверх фото товара. При наведении на кнопку добавления товара в корзину, она будет менять цвет. При нажатии на кнопку добавления товара в корзину, она будет менять цвет и появится надпись, что товар добавлен в корзину. Код карточки товара будет показан в листинге 1.3.

|  |
| --- |
| export default function Card(props) {    return (      <div className="card">        <form className="cardContainer">          <div className="cardContainerContent">            <div className="productPhotoContainer">              <div className="favoriteIcon">                <img                  className="favoriteIconImg"                  src={favorites}                  alt="в избранное"                />              </div>              <div className="productPhoto">                <Link to="/catalog/item?009">                  <img src={koleso} alt="изображение товара" />                </Link>              </div>            </div>            <div className="productTitle">              <Link to="/catalog">Шина</Link>            </div>            <div className="productDescription">              Размеры 215/55 R16 97V Бренд Belshina            </div>            <div className="productBottom">              <div className="productBottomContainer">                <div className="productPrice">                  <span>100 Byn</span>                </div>                <div className="productControls">                  <button>                    <img src={shopCard} alt="добавить в корзину" />                  </button>                </div>              </div>            </div>          </div>        </form>      </div>    );  } |

Листинг 1.3 – Структура карточки товара

Следующим компонентом будет Basket компонент. В нем будет находится товары, которые были добавлены в корзину. Также будет находится блок с общей суммой товаров в корзине и кнопкой оформления заказа. На товаре будут кнопки добавить в корзину и удалить из корзины. Код корзины будет показан в листинге 1.4.

|  |
| --- |
| <div className="mainSection">        <div className="mainZone">          <div className="mainSectionContainer">            <Breadcrumb>              <Breadcrumb.Item>                <Link to="/">Главная</Link>              </Breadcrumb.Item>              <Breadcrumb.Item>Корзина</Breadcrumb.Item>            </Breadcrumb>            <div className="mainSectionContainerContent">              {isGoods ? (                <form className="basketContent">                  <div className="items">{[items]}</div>                  <div className="controls">                    <div className="controlsContainer">                      <div className="counter">                        <div className="counterHeader">В корзине</div>                        <div className="counterInfo">                          Товаров: <span className="counterInfoCount">4</span>                        </div>                      </div>                      <div className="total">                        <div className="totalHeader">Итого</div>                        <div className="totalInfo">3000$</div>                      </div>                      <div className="SubmitButton">                        <Link to="/order" className="SubmitButtonLink">                          <div>Оформить заказ</div>                        </Link>                      </div>                    </div>                  </div>                </form>              ) : (                <div className="emptyContent">Ваща корзина пуста</div>              )}            </div>          </div>        </div>      </div> |

Листинг 1.4 – Структура кода компонента Basket

Таким образом, в этом разделе были рассмотрены основные компоненты приложения, которые будут использоваться в дальнейшем. Хотелось бы отметить, что JSX-формат очень удобный для написания кода, так как он позволяет писать код в HTML-формате, что упрощает его чтение и понимание.

## 3.2 Структура Redux-Toolkit

Для того, чтобы удобно работать с данными, был использован Redux-Toolkit. В нем будет хранится информация о веб страницах и передаваться информация между компонентами.

Благодаря CreateSlice был реализован редьюсер для получения данных о товарах, о пользователях, о заказах, о корзине. В CreateSlice будет находиться имя редьюсера, начальное состояние, редьюсеры для изменения состояния, асинхронные запросы к серверу. Для того, чтобы получить данные нужно использовать ExtraReducers, в котором будет находиться асинхронный запрос к серверу. Код CreateSlice будет показан в листинге 1.5.

|  |
| --- |
| import { createSlice } from "@reduxjs/toolkit";  const initialState = {    width: 1280,  };  export const WindowSizeSlice = createSlice({    name: "windowSize",    initialState,    reducers: {      changeSize: (state, action) => {        state.width = action.payload;      },    },  });  export const { changeSize } = WindowSizeSlice.actions;  export default WindowSizeSlice.reducer; |

Листинг 1.5 – Структура кода CreatSlice

Затем все редьюсеры нужно объединить в один редьюсер и поместить его в Store. В Store будет находиться все состояние приложения. Для того, чтобы получить данные из Store нужно использовать useSelector. Для того, чтобы изменить данные в Store нужно использовать useDispatch. Код структуры Store будет показан в листинге 1.6.

|  |
| --- |
| import { configureStore } from "@reduxjs/toolkit";  import WindowSizeSlice from "../Slices/WindowSizeSlice.js";  import MainPageSlice from "../components/MainPage/Slices/MainPageSlice.js";  export const store = configureStore({    reducer: { windowSize: WindowSizeSlice, mainPage: MainPageSlice },  }); |

Листинг 1.6 – Структура кода Store

В этом разделе были рассмотрены осноыные положения по созданию глобального хранилища данных с помощью Redux-Toolkit. Были рассмотрены основные понятия по созданию редьюсеров, асинхронных запросов к серверу, получения данных из Store.

## 3.3 Добавление таблиц стилей Sass

Для стилизации был использован SASS, который позволяет использовать переменные, вложенность, миксины, наследование, импорт и экспорт. Код использования SASS показан в листинге 2.5.

|  |
| --- |
| .main{      width: 100%;      height: fit-content;      font-family: 'Montserrat';      .mainZone{          width: 84vw;          height: 100%;          padding: 0vh 8vw 0vh 8vw;          .mainContainer{              height: inherit;              width: inherit;               .mainContainerTitleContainer{                  width: 100%;                  .mainContainerTitle{                      font-size: 1rem;                      width: fit-content;                      padding: 4px 0;                      &::after{                          content: '';                          display: block;                          height: 1px;                          background-color: black;                          margin-top: 2px;                      }                  }              }              .mainContainerContent{                  display: grid;                  grid-template-columns: repeat(auto-fill, minmax(min(180px, 100%), 1fr));                  grid-template-rows: auto;                  grid-row-gap: 2rem;                  grid-column-gap: 2rem;                  margin-top: 2rem;              } |

Листинг 2.5 – Структура кода SASS

В этом разделе была рассмотрена структура SASS. Что такое SASS и как он используется в проекте.

## 3.4 Выводы

Для реализации проекта будут использованы современные методы и подходы к разработке веб-страниц, которые помогут ускорить процесс разработки а также сделать веб-страницы кроссбраузерными и адаптивными.

# 4. Тестирование веб-сайта

## 4.1 Адаптивный дизайн веб-сайта

Сегодня количество пользователей, которые заходят в Интернет с мобильных устройств очень велико, и с каждым днем их становится все больше. А потому адаптация веб-сайта под маленькие экраны – актуальная задача и для курсового проекта.

Передо мной стояла задача: создать сайт, который будет сам подстраиваться под разные типы экранов. Главными условиями адаптивности является не фиксированный размер элементов страницы.

Для создания адаптивности веб-сайта были использованы следующие инструменты: медиа-запросы (@media), единицы измерения, которые связаны непосредственно с параметрами области просмотра (представляют собой процентные величины от текущей области просмотра браузера).

После написания основной структуры страницы на JSX и внешнего стилевого оформления появился вопрос о тестировании. Веб-сайт был открыт при помощи различных браузеров. После этого было выявлено, что во всех браузерах сайт выглядит одинаково.

При отображении сайта на мобильных устройствах навигационное меню в верхнем колонтитуле меняется на выпадающее меню, открывающееся при нажатии на кнопку меню. На рисунке 3.1 можно увидеть пример адаптивного дизайна для мобильных устройств.

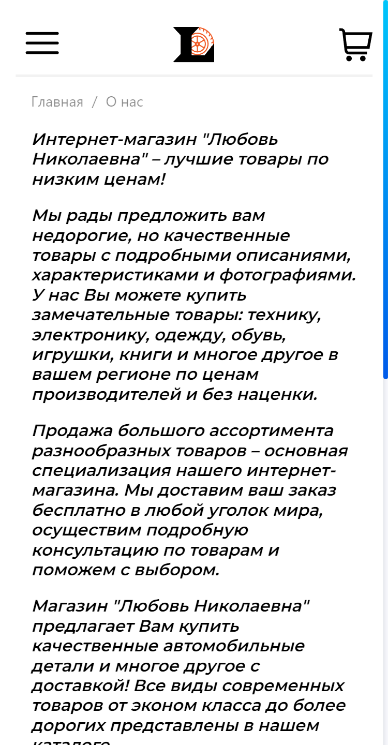


Рисунок 3.1 – Адаптивный главной страницы для телефонов

Также при использовании мобильного устройства блоки с продуктами на странице для отображения располагаются друг под другом. На рисунке 3.2 можно увидеть пример.

****

Рисунок 3.2 – Блоки с продуктами на мобильных устройствах

И таким образом можно сказать, что все блоки располагаются друг за друг, сделано это для того, чтобы пользователю было все хорошо видно на его мобильном устройстве.

## 4.2 Кроссбраузерность веб-сайта

Очень важно, чтобы наш сайт был кроссбраузерным. 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.

У каждого браузера есть свои встроенные, экспериментальные или нестандартные свойства и для того, чтобы они корректно работали было принято решения о внедрении вендорных префиксов. Они были автоматически прописаны после компиляции файла с расширением sass.

Веб-сайт будет тестироваться в следующих браузерах: Google Chrome, Opera и Яндекс.

При проверке веб-сайта в Google Chrome никаких искажений текста, анимации, переходов не было выявлено. На рисунке 3.3 можно увидеть пример веб-страницы в Google Chrome.

|  |  |
| --- | --- |
|  |  |
| а) | б) |

Рисунок 3.3 – Вид веб-страницы в Google Chrome а) мобильная версия; б) компьютерная версия

В Opera сайт показал схожий результат: никаких отклонений как для мобильной версии сайта, так и для компьютерной не было выявлено, текст не съехал, не произошло его наложения на картинки. На рисунке 3.4 можно увидеть пример веб-страницы в Opera.

|  |  |
| --- | --- |
|  |  |
| а) | б) |

Рис 3.4 – Вид веб-страницы в Opera а) мобильная версия; б) компьютерная версия

Также для тестирования сайта было использовано Linter. Linter - это инструмент, который проверяет код на наличие ошибок. В нашем случае он проверяет код на наличие ошибок в HTML и CSS.

При просмотре веб-сайта с помощью таких браузеров как Google Chrome, Яндекс и Opera общий вид веб-страницы не изменился.

Главной целью такой тщательной проверки сайта является грамотная настройка всех показателей, одной или всех страниц сайта, оперативное обнаружение и устранение всех дефектных мест на сайте, а также его адаптация под различные устройства.

Динамика сайта на разных устройствах – наиболее важная часть в написании сайта, поэтому, на данном этапе было проанализировано поведение веб-сайта при использовании с разных устройств, а также веб-ресурс был протестирован в большом количестве браузеров. По итогу выполнения данного этапа для веб-сайта был разработан адаптивный дизайн и кроссбраузерность. Работа, проделанная с сайтом, позволяет пользователям ориентироваться по сайту. Лишних элементов нет, что упрощает пользование сайтом. При пользовании сайтом при помощи мобильного устройства некоторые элементы меняют свое положение, для более удобного пользования. В ходе тестирования были проверены все ссылки (включая модальные окна), проверены формы регистрации пользователя. Была выполнена проверка контента. Также в ходе тестирования была проверена совместимость с браузерами. (Google Chrome, Opera, Яндекс).

## 4.3 Тестирование функционала сайта

Тестирование функционала сайта – это проверка всех функций сайта, которые должны работать корректно, а также проверка всех элементов, которые должны быть доступны для пользователя. Основными функциями сайта являются фильтрация, поиск, добавление в корзину, оформление заказа, а также просмотр и редактирование личных данных. Тестирование функционала сайта было выполнено вручную с использованием следующие браузеров Google Chrome (42.0.2311.90), Opera (29.0.1795.47) и Яндекс (22.1.0.2510)

В браузере Google Chrome были проверены все функции сайта. Можно сказать, что все функции сайта работают корректно.

В браузере Opera также были проверены все функции сайта. Хочется отметить, что в браузере Opera работает все функции сайта, кроме функции добавления в корзину.

Также было проведено тестирование функционала сайта в браузере Яндекс.

И как в других браузерах, так и в браузере Яндекс все функции сайта работают корректно.

Подводя итоги тестирования функционала сайта, можно сказать, что все функции сайта работают корректно. В ходе тестирования были проверены все ссылки (включая модальные окна), проверены формы регистрации пользователя. Была выполнена проверка контента.

## 4.4 Выводы

В этом разделе было проведено тестирование функционала сайта. Были проверены все функции сайта, которые должны работать корректно, а также были проверены все элементы, которые должны быть доступны для пользователя. Не было обнаружено никаких ошибок, которые могли бы повлиять на работу сайта.

# Заключение

В процессе реализации курсового проекта был проделан полный путь создания веб-сайта, начинающийся с постановки задач и проектирования веб-страниц, и, заканчивающийся его тестированием. В ходе создания веб-страниц были использованы современные методы проектирования, методы создания веб-страниц, добавление адаптивности.

В ходе проектирования были найдены способы размещения и представления контента на веб-страницах, продумано стилевое оформление, выбраны шрифты, созданы логотипы веб-сайта, разработано взаимодействие пользователя с функциональными элементами веб-страниц. Также были продуманны пользовательские элементы.

При создании веб-страниц были использованы методы адаптивной верстки: flex-верстки, grid-верстки, резиновой верстки и применения медиа-запросов. Благодаря адаптивной верстке пользователь будет иметь удобный доступ к контенту веб-страниц без потери информации. Также весь контент был расположен в семантических тегах, которые облегчат браузеру восприятие веб-страниц. Работа с каскадными таблицами стилей значительно упростилась и ускорилась благодаря применению мета языка Scss.

В процессе тестирования была проверена адаптивность с сайта к различным устройствам с помощью панели инструментов устройств в браузере Google Chrome. Также было протестировано поведение веб-страниц в разных браузерах, результатом которого являлась кроссбраузерность веб-страниц веб-сайта. В завершении была проведена валидация кода веб-страниц, результатом которой стала его валидность.

По итогу веб-сайт получился красивым и удобным для использования. В ходе его создания были применены современные методы разработки веб-страниц. На веб-сайте можно найти всю необходимую пользователю информацию и удобный инфтерфейс для взаимодействия с веб-страницами. Веб-сайт имеет кроссбраузерность и аддаптивность.

# Список использованных источников

1. Документация по React JS [Электронный ресурс]. - Режим доступа : <https://reactjs.org>/. – Дата обращения 21.11.2022.
2. Документация по Figma [Электронный ресурс]. - Режим доступа: <https://help.figma.com>. – Дата обращения 21.11.2022.
3. Документация по Redux/Redux-tolkit [Электронный ресурс]. - Режим доступа : <https://redux-toolkit.js.org/> . – Дата обращения 21.11.2022.
4. Документация по Sass/Scss [Электронный ресурс]. - Режим доступа : <https://sass-scss.ru/guide/> – Дата обращения 21.11.2022.

# Приложение

## Приложение А Макет структуры веб-сайта

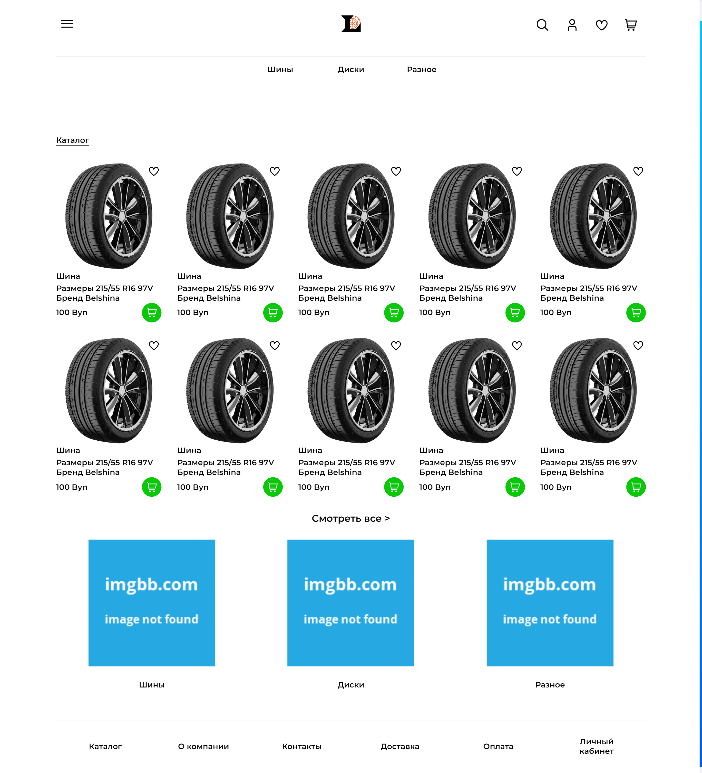
 

Рисунок 1-макет страницы “Main”

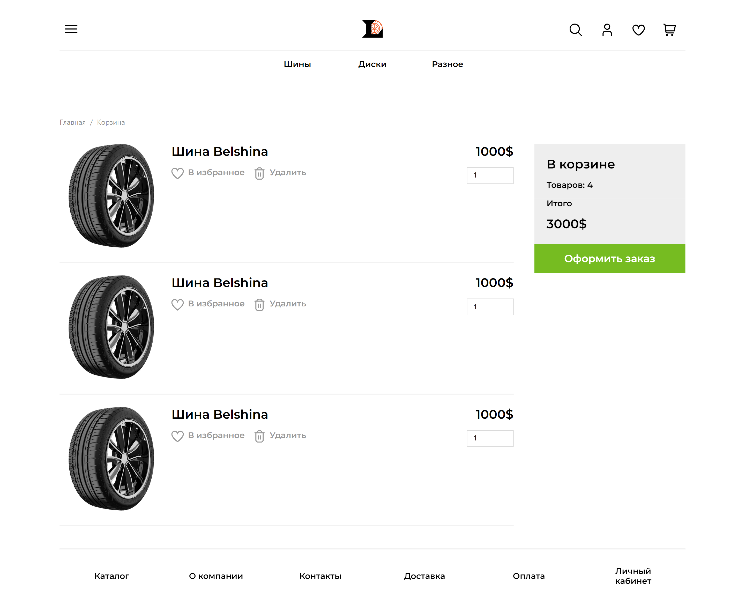
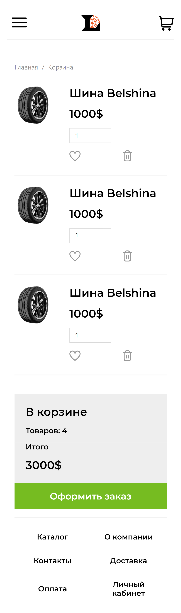
 

Рисунок 2-макет страницы “Basket”

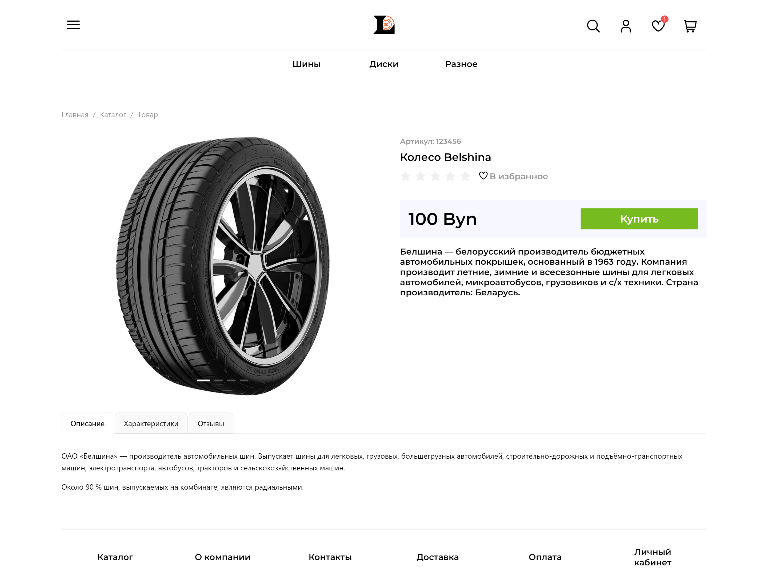
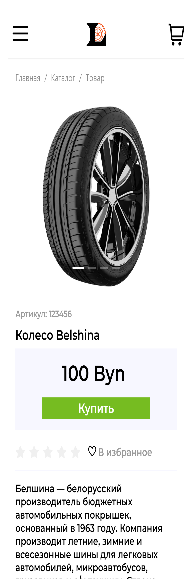
 

Рисунок 3-макет страницы “Item”

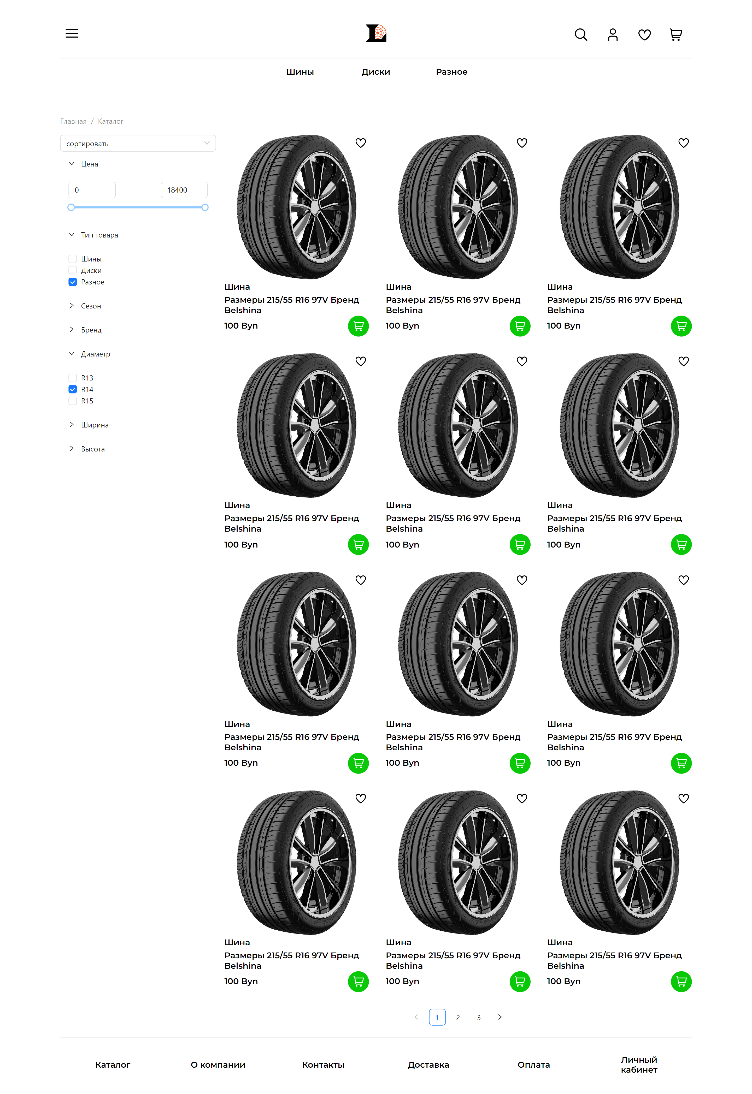
 

Рисунок 4-макет страницы “Catalog”

## Приложение Б Листинг JSX-документа

|  |
| --- |
| import React from "react";  import "./MainSection.scss";  import {    Breadcrumb,    Carousel,    Form,    Rate,    Tabs,    Input,    Button,    Collapse,  } from "antd";  import { Link } from "react-router-dom";  import { useState } from "react";  import { useSelector } from "react-redux";  import heartImg from "./files/heart.svg";  import koleso from "./files/koleso.jpg";  import FormItem from "antd/es/form/FormItem";  const validateMessages = {    required: "${label} is required!",    types: {      email: "${label} is not a valid email!",    },  };  export default function MainSection() {    return (      <div className="mainSection">        <div className="mainZone">          <Breadcrumb>            <Breadcrumb.Item>              <Link to="/">Главная</Link>            </Breadcrumb.Item>            <Breadcrumb.Item>              <Link to="/catalog">Каталог</Link>            </Breadcrumb.Item>            <Breadcrumb.Item>Товар</Breadcrumb.Item>          </Breadcrumb>          <div className="mainSectionContainer">            <form className="product">              <div className="productImages">                <Carousel>                  <div className="productImage">                    <img src={koleso} alt="товар" />                  </div>                  <div className="productImage">                    <img src={koleso} alt="товар" />                  </div>                  <div className="productImage">                    <img src={koleso} alt="товар" />                  </div>                  <div className="productImage">                    <img src={koleso} alt="товар" />                  </div>                </Carousel>              </div>              <div className="productSkuContainer">                <div className="productSku">Артикул: 123456</div>              </div>              <div className="productTitleContainer">                <h1 className="productTitle">Колесо Belshina</h1>              </div>              <div className="productControlsContainer">                <div className="productControls">                  <div className="productRaiting">                    <Rate className="rate" />                  </div>                  <div className="productFavorites">                    <button className="Button">                      <span className="controlImg">                        <img src={heartImg} alt="избр" />                      </span>                      <span className="controlText">В избранное</span>                    </button>                  </div>                </div>              </div>              <div className="productVariantsContainer"></div>              <div className="productBuyContainer">                <div className="productBuy">                  <div className="productPrice">                    <span>100 Byn</span>                  </div>                  <div className="productBuyButtonContainer">                    <button className="productBuyButton">                      <span>Купить</span>                    </button>                  </div>                </div>              </div>              <div className="productDescriptionContainer">                <div className="productDescription">                  Белшина — белорусский производитель бюджетных автомобильных                  покрышек, основанный в 1963 году. Компания производит летние,                  зимние и всесезонные шины для легковых автомобилей,                  микроавтобусов, грузовиков и с/х техники. Страна производитель:                  Беларусь.                </div>              </div>            </form>            <div className="tabs">              {useSelector((state) =>                state.windowSize.width > 550 ? (                  <Tabs                    type="card"                    tabBarStyle={{ color: "black !important" }}                    style={{ color: "black" }}                  >                    <Tabs.TabPane tab="Описание" key="1">                      <p>                        ОАО «Белшина» — производитель автомобильных шин. Выпускает                        шины для легковых, грузовых, большегрузных автомобилей,                        строительно-дорожных и подъёмно-транспортных машин,                        электротранспорта, автобусов, тракторов и                        сельскохозяйственных машин.                      </p>                      <p>                        Около 90 % шин, выпускаемых на комбинате, являются                        радиальными.                      </p>                    </Tabs.TabPane>                    <Tabs.TabPane tab="Характеристики" key="2">                      <div className="propertiesItems">                        <div className="propertiesItem">                          <div className="propertiesItemTitle">Ширина</div>                          <div className="propertiesDel"></div>                          <div className="propertiesItemValue">205</div>                        </div>                        <div className="propertiesItem">                          <div className="propertiesItemTitle">Размер</div>                          <div className="propertiesDel"></div>                          <div className="propertiesItemValue">205</div>                        </div>                        <div className="propertiesItem">                          <div className="propertiesItemTitle">Высота</div>                          <div className="propertiesDel"></div>                          <div className="propertiesItemValue">205</div>                        </div>                      </div>                    </Tabs.TabPane>                    <Tabs.TabPane tab="Отзывы" key="3">                      <Form                        labelCol={{                          span: 6,                        }}                        wrapperCol={{                          span: 12,                        }}                        validateMessages={validateMessages}                      >                        <Form.Item                          name={["user", "email"]}                          label="Email"                          rules={[                            {                              required: true,                              type: "email",                            },                          ]}                        >                          <Input></Input>                        </Form.Item>                        <FormItem                          name={["user", "name"]}                          label="Имя"                          rules={[                            {                              required: true,                            },                          ]}                        >                          <Input></Input>                        </FormItem>                        <Form.Item                          label="Отзыв"                          name={["user", "message"]}                          rules={[                            {                              required: true,                            },                          ]}                        >                          <Input.TextArea />                        </Form.Item>                        <Form.Item                          wrapperCol={{                            offset: 6,                            span: 12,                          }}                        >                          <Button type="primary" htmlType="submit">                            Отправить                          </Button>                        </Form.Item>                      </Form>                      <div className="reviewsContainer">                        <div className="reviewItem">                          <div className="reviewItemHeader">                            <div className="reviewItemName">Имя</div>                            <div className="reviewItemEmail">@idwkd</div>                            <div className="reviewItemDate">12.12.2022 14:56</div>                            <div className="reviewItemComment">                              Lorem Ipsum is simply dummy text of the printing and                              typesetting industry. Lorem Ipsum has been the                              industry's standard dummy text ever since the 1500s,                              when an unknown printer took a galley of type and                              scrambled it to make a type specimen book. It has                              survived not only five centuries, but also the leap                              into electronic typesetting, remaining essentially                              unchanged. It was popularised in the 1960s with the                              release of Letraset sheets containing Lorem Ipsum                              passages, and more recently with desktop publishing                              software like Aldus PageMaker including versions of                              Lo                            </div>                          </div>                        </div>                      </div>                    </Tabs.TabPane>                  </Tabs>                ) : (                  <Collapse bordered={false} className="collapse">                    <Collapse.Panel header="Описание" key="1">                      <p>                        ОАО «Белшина» — производитель автомобильных шин. Выпускает                        шины для легковых, грузовых, большегрузных автомобилей,                        строительно-дорожных и подъёмно-транспортных машин,                        электротранспорта, автобусов, тракторов и                        сельскохозяйственных машин.                      </p>                      <p>                        Около 90 % шин, выпускаемых на комбинате, являются                        радиальными.                      </p>                    </Collapse.Panel>                    <Collapse.Panel header="Характеристики" key="2">                      <div className="propertiesItems">                        <div className="propertiesItem">                          <div className="propertiesItemTitle">Ширина</div>                          <div className="propertiesDel"></div>                          <div className="propertiesItemValue">205</div>                        </div>                        <div className="propertiesItem">                          <div className="propertiesItemTitle">Размер</div>                          <div className="propertiesDel"></div>                          <div className="propertiesItemValue">205</div>                        </div>                        <div className="propertiesItem">                          <div className="propertiesItemTitle">Высота</div>                          <div className="propertiesDel"></div>                          <div className="propertiesItemValue">205</div>                        </div>                      </div>                    </Collapse.Panel>                    <Collapse.Panel header="Отзывы" key="3">                      <Form                        labelCol={{                          span: 6,                        }}                        wrapperCol={{                          span: 12,                        }}                        validateMessages={validateMessages}                      >                        <Form.Item                          name={["user", "email"]}                          label="Email"                          rules={[                            {                              required: true,                              type: "email",                            },                          ]}                        >                          <Input></Input>                        </Form.Item>                        <FormItem                          name={["user", "name"]}                          label="Имя"                          rules={[                            {                              required: true,                            },                          ]}                        >                          <Input></Input>                        </FormItem>                        <Form.Item                          label="Отзыв"                          name={["user", "message"]}                          rules={[                            {                              required: true,                            },                          ]}                        >                          <Input.TextArea />                        </Form.Item>                        <Form.Item                          wrapperCol={{                            offset: 6,                            span: 12,                          }}                        >                          <Button type="primary" htmlType="submit">                            Отправить                          </Button>                        </Form.Item>                      </Form>                      <div className="reviewsContainer">                        <div className="reviewItem">                          <div className="reviewItemHeader">                            <div className="reviewItemName">Имя</div>                            <div className="reviewItemEmail">@idwkd</div>                            <div className="reviewItemDate">12.12.2022 14:56</div>                            <div className="reviewItemComment">                              Lorem Ipsum is simply dummy text of the printing and                              typesetting industry. Lorem Ipsum has been the                              industry's standard dummy text ever since the 1500s,                              when an unknown printer took a galley of type and                              scrambled it to make a type specimen book. It has                              survived not only five centuries, but also the leap                              into electronic typesetting, remaining essentially                              unchanged. It was popularised in the 1960s with the                              release of Letraset sheets containing Lorem Ipsum                              passages, and more recently with desktop publishing                              software like Aldus PageMaker including versions of                              Lo                            </div>                          </div>                        </div>                      </div>                    </Collapse.Panel>                  </Collapse>                )              )}            </div>          </div>        </div>      </div>    );  } |

Структура JSX—документа

## Приложение В Листинг (SASS)

|  |
| --- |
| .mainSection{      height: fit-content;      font-family: 'Montserrat';      .mainZone{          width: 84vw;          height: 100%;          padding: 0vh 8vw 0vh 8vw;          min-height: 60vh;          .mainSectionContainer{              padding-top: 2rem;              .product{                  display: grid;                  grid-template-areas:                      "photo sku"                      "photo title"                      "photo user-controls"                      "photo variants"                      "photo buy"                      "photo description"                      "photo . ";                  grid-template-columns: 50% 1fr;                  grid-template-rows: repeat(6, auto) 1fr;                  grid-column-gap: 30px;                  .productImages{                      grid-area: photo;                      max-width: 84vw;                      justify-self: center;                      width: 100%;                      height: 100%;                      div.productImage{                          width: 100%;                          display: flex !important;                          justify-content: center;                          align-items: center;                          img{                              display: block;                              max-width: 500px;                              max-height: 500px;                              width: 80%;                              height: 80%;                          }                      }                  }                  .productSkuContainer{                      grid-area: sku;                      .productSku{                          color: #999999;                          font-size: 0.85rem;                          margin-bottom: 10px                      }                  }                  .productTitleContainer{                      grid-area: title;                      .productTitle{                          line-height: 1.2;                          margin-bottom: 10px;                          font-size: 1.25rem;                          margin: 0;                          margin-bottom: 8px;                      }                  }                  .productControlsContainer{                      grid-area: user-controls;                      padding-bottom: 1rem;                      border-bottom: 1px solid #f7f7ff;                      margin-bottom: 1rem;                        .productControls{                          display: flex;                          flex-direction: row;                          flex-wrap: wrap;                          width: fit-content;                          gap: 1rem;                          .productFavorites{                              display: flex;                              flex-direction: row;                              justify-content: center;                              align-items: flex-end;                              .Button{                                  all: unset;                                  cursor: pointer;                                  display: flex;                                  flex-direction: row;                                  gap: 4px;                                  color: #999999;                                  span{                                      white-space: nowrap;                                      img{                                          width: 1rem;                                          height: 1rem;                                      }                                  }                              }                          }                          .productRaiting{                              .rate{                              }                          }                      }                  }                  .productVariantsContainer{                      grid-area: variants;                      display: flex;                      flex-wrap: wrap;                  }                  .productBuyContainer{                      grid-area: buy;                      .productBuy{                          display: grid;                          grid-template-columns: 1fr minmax(220px, max-content);                          grid-column-gap: 1rem;                          grid-template-rows: auto;                          grid-auto-flow: column;                          // display: flex;                          // flex-direction: row;                          // flex-wrap: wrap;                          align-items: center;                          margin-bottom: 1rem;                          padding: 1rem;                          background-color: #f7f7ff;                          @media screen and (max-width: 767px){                              & {                                  display: flex;                                 flex-direction: column;                                  align-items: center;                                  justify-content: center;                                  gap: 16px;                              }                          }                          .productPrice{                              span{                                  font-size: 2rem;                                  font-weight: 600;                                  margin-right: 10px;                                  white-space: nowrap;                              }                          }                          .productBuyButtonContainer{                              width: 100%;                              height: 100%;                              display: flex;                              flex-direction: row;                              justify-content: center;                              .productBuyButton{                                  all: unset;                                  cursor: pointer;                                  background-color: #76bc21;                                  width: 100%;                                  height: 100%;                                  min-height: 2rem;                                  max-width: 220px;                                  display: flex;                                  justify-content: center;                                  align-items: center;                                  &:hover{                                    background-color:  #83bf3a;                                  }                                  span{                                      color: #fff;                                      font-size: 1.2rem;                                  }                              }                          }                      }                  }                  .productDescriptionContainer{                      grid-area: description;                       min-width: 0;                       .productDescription{                          margin-bottom: 10px;                          word-wrap: break-word;                       }                  }                  @media screen and (max-width: 1024px){                      & {                          display: grid;                          grid-template-areas:                              "photo"                              "sku"                              "title"                              "variants"                              "buy"                              "user-controls"                              "description"                              "calculate-delivery";                          grid-template-columns: auto;                          grid-template-rows: auto;                          grid-column-gap: 0;                          .productImages{                              margin-bottom: 2rem;                          }                      }                  }              }              .tabs{                  margin:2rem 0;                  div[role="tab"]{                      color: black;                  }                  .propertiesItems{                      display: grid;                      grid-template-columns: repeat(auto-fit, minmax(min(450px, 100%), 1fr));                      gap: 1rem 3rem;                      .propertiesItem{                          display: grid;                          grid-template-columns: auto 1fr 200px;                          align-items: end;                          gap: 1rem;                          .propertiesItemTitle{                              max-width: 200px;                          }                          .propertiesItemValue{                            }                          .propertiesDel{                              width: 100%;                              height: 2px;                              border-bottom: 1px dotted #999999;                              margin-bottom: 8px;                          }                        }                  }                  .reviewsContainer{                      width: 100%;                      height: 100%;                      max-width: 540px;                      display: flex;                      flex-direction: column;                      .reviewItemDate{                          color: #999999;                      }                      .reviewItemComment{                          padding-top:1rem ;                      }                    }                  .collapse{                      background-color: #f7f7ff64;                  }              }          }      }  } |

Структура SASS—документа

## Приложение Г Листинг Redux

|  |
| --- |
| import { configureStore } from "@reduxjs/toolkit";  import WindowSizeSlice from "../Slices/WindowSizeSlice.js";  import MainPageSlice from "../components/MainPage/Slices/MainPageSlice.js";  export const store = configureStore({    reducer: { windowSize: WindowSizeSlice, mainPage: MainPageSlice },  }); |

|  |
| --- |
| import { createSlice } from "@reduxjs/toolkit";  const initialState = {    width: 1280,  };  export const WindowSizeSlice = createSlice({    name: "windowSize",    initialState,    reducers: {      changeSize: (state, action) => {        state.width = action.payload;      },    },  });  export const { changeSize } = WindowSizeSlice.actions;  export default WindowSizeSlice.reducer; |