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 «Салон культовых авто»

Исполнитель

студент 2 курса 1 группы Вакуленчик Владислав Сергеевич

(Ф.И.О.)

Руководитель Жиляк Н. А.

(учен. степень, звание, должность, подпись, Ф.И.О.)

Курсовой проект защищен с оценкой ­

Руководитель Жиляк Н.А.

(подпись)

**Оглавление**

[**Введение** 3](#_Toc121646457)

[**1. Обзор технических методов и программных средств разработки веб-сайта** 4](#_Toc121646458)

[**1.1 Обзор аналогичных решений** 4](#_Toc121646459)

[**1.2 Формирование требований к программному продукту** 5](#_Toc121646460)

[**1.3 Постановка задач** 5](#_Toc121646461)

[**1.4 Выбор средств реализации программного средства** 6](#_Toc121646462)

[**1.5 Вывод** 6](#_Toc121646463)

[**2. Последовательность разработки содержания и структуры веб-сайта** 7](#_Toc121646464)

[**2.1 Техническое задание** 7](#_Toc121646465)

[**2.2 Построение прототипов интерфейсов и макетирование страниц** 7](#_Toc121646466)

[**2.3 Разработка дизайна элементов программного продукта** 7](#_Toc121646467)

[**2.4 Выбор и обоснование используемых специальных эффектов** 10](#_Toc121646468)

[**2.5 Разработка решений по общесистемным вопросам** 10](#_Toc121646469)

[**2.6 Составление структуры программного продукта** 11](#_Toc121646470)

[**2.7 Вывод** 12](#_Toc121646471)

[**3. Программная реализация проекта** 13](#_Toc121646472)

[**3.1 Верстка программного продукта** 13](#_Toc121646473)

[**3.3 Вывод** 15](#_Toc121646474)

[**4. Описание хода работы программного средства с копиями экрана пользовательского интерфейса основных элементов системы, а также ключевые функции разработанного программного кода** 16](#_Toc121646475)

[**4.1 Руководство пользователя** 16](#_Toc121646476)

[**4.2 Кроссбраузерное тестирование** 17](#_Toc121646477)

[**4.3 Вывод** 18](#_Toc121646478)

[**Заключение** 19](#_Toc121646479)

[**Список использованных литературных источников** 20](#_Toc121646480)

[**Приложение** 21](#_Toc121646481)

[**Приложение А. Макет веб-страниц** 21](#_Toc121646482)

[**Приложение Б. Прототип веб-страниц** 25](#_Toc121646483)

# **Введение**

React — это JavaScript-библиотека для создания пользовательских интерфейсов. Он показывает на странице компонент интерфейса, синхронизируя его с данными приложения. Для полной разработки приложений, также используются другие библиотеки, совместно с React.

React-разработчики создают приложения на React, используя дополнительные инструменты: например, Redux, TypeScript или Jest. Это востребованная работа: на React.JS написаны Яндекс, Netflix, Facebook и другие известные сервисы.

Задача курсовой работы состоит в создании веб-сайта с удобным интерфейсом, используя востребованные и часто используемые технические средства, и компоненты.

Сам проект был разбит на несколько разделов, в каждом из которых он приобретал все более адаптивный под его пользователей вид. Несмотря на стандартные средства React, были подключены библиотеки для роутинга, редукс и многое другое.

Задачей проекта было создание макета и прототипов, соответствующих последним правилам дизайна. А целью - создание адаптивного и кроссбраузерного приложения.

# **1. Обзор технических методов и программных средств разработки веб-сайта**

# **1.1 Обзор аналогичных решений**

Курсовой проект представляет из себя приложение салона культовых автомобилей. Так как в интернет-пространстве существуют множество разнообразных веб-сайтов салонов автомобилей с уникальным дизайном, то перед написанием своего собственного, я изучил подобные аналоги.

Первым аналогом является сайт bugatti.com, в котором ограничено использование текста, визуальных эффектов, а также он охватывает пустое пространство. Стоит отметить отлично проработанную адаптацию под мобильные устройства. Дизайн сайта представлен на рисунке 1.1

|  |
| --- |
|  |

Рисунок 1.1 – Веб-сайт https://www.bugatti.com/

Следующее аналогичное решение nissanusa.com. Он содежрит несколько разделов с множественными автомобилями. Дизайн сайта представлен на рисунке 1.2.

|  |
| --- |
|  |

Рисунок 1.2 – Веб-сайт https://www.nissanusa.com/vehicles/sports-cars/gt-r.html

# **1.2 Формирование требований к программному продукту**

Конечный продукт должен главным образом соответствовать требованию адаптивности, а также кроссбраузерности. Должна быть проведена работа с дополнительными пакетами, для увеличения возможности использования продукта.

# **1.3 Постановка задач**

Основными задачами курсового проекта являются:

* создание уникального дизайна
* подключение хуков и дополнительных пакетов
* создание адаптивного и кроссбаузерного продукта

# **1.4 Выбор средств реализации программного средства**

Для реализации проекта были выбраны: React JS c установкой дополнительных компонентов: Redux Toolkit, React Router DOM, Material Core, а для дизайна - CSS/Sass. Курсовой проект выполняется в редакторе кода VS Code.

# **1.5 Вывод**

В первом разделе осуществлен обзор технических методов и программных средств для реализации курсовой работы. Были определены задачи веб-сайта, а также рассмотрены аналоговые решения, их преимущества и недостатки.

# **2. Последовательность разработки содержания и структуры веб-сайта**

# **2.1 Техническое задание**

В данном проекте необходимо создать веб-сайт салона культовых автомобилей размерностью в 1 страницу, но с множественным рендерингом компонентов. Он будет содержать в себе информацию об автомобилях, салоне, контактах связи с ним.

# **2.2 Построение прототипов интерфейсов и макетирование страниц**

Обязательным этапом создания приложения, является разработка прототипов и макетов всех страниц. Это нужно для того, чтобы корректно разместить все элементы по всем компонентам, и чтобы они не конфликтовали друг с другом. Макет и прототип показаны в приложении 1 и 2 соответственно.

# **2.3 Разработка дизайна элементов программного продукта**

Поскольку по тематике, на заднем фоне салона находится изображение космоса, то и основные цвета соответственно синие, серые и белые. Также стоит отметить, что в приложении будут 3Д-модели продаваемых автомобилей. Хотя и нужно достаточно времени, чтобы их прогрузить для пользователя при первом заходе на веб-сайт, это будет являться отличительной чертой приложения, которая поможет выделиться от конкурентов.

В проекте будет применен шрифт Montserrat. Шрифт Montserrat-Bold будет применяться для заголовков, Montserrat-Light - блока меню, a Montserrat-Regular – для текса абзацев.

Разработка логотипа для веб-сайта проводилась в приложении Adobe Illustrator 2020. Он представлен на рисунке 2.1.

|  |
| --- |
|  |

Рисунок 2.1 – Логотип приложения.

Основными пользовательскими элементами являются кнопки, выполненные в одном гармоничном стиле. Те, что расположены в главном меню ссылают на внутренние элементы страницы, остальные - ссылки на элементы других компонентов. При наведении на отдельные кнопки меню, срабатывают псевдо-классы.

# **2.4 Выбор и обоснование используемых специальных эффектов**

Основные элементы, которые можно отнести к специальным эффектам, является 3Д-модели, которые можно будет вращать курсором. Они установлены с помощью библиотеки Three.js.

# **2.5 Разработка решений по общесистемным вопросам**

Так как приложение должно поддерживать один общий дизайн, то помимо выбора основных цветов элементов, они также все имеют закругленные углы (свойство border-radius), как представлено на рисунке 2.2.

|  |
| --- |
| а    б |

Рисунок 2.2 – примеры элементов приложения: а - навигация, header; б - форма заказа

# **2.6 Составление структуры программного продукта**

Поскольку основным требованием было создание адаптивного продукта, то для разработки были применены технологии FlexBox верстки, которые слегкостью помогают перенести элемент контейнера на новую строку при уменьшении ширины экрана (свойство flex-wrap: wrap) и многих других технологий.

# **2.7 Вывод**

В ходе работы в данном разделе была проделана большая работа в выборе способа верстки, а также стилевого оформления. Были подобраны основные цветовые оттенки и стили шрифтов, и проведена работа над логотипом продукта. Немаловажным оказалась создание макета и прототипа, по которому будет проводиться верстка, и разработка динамических эффектов в нем.

# **3. Программная реализация проекта**

Для реализации приложения, как говорилось ранее, была выбрана библиотека React JS, которая расширяет возможности обычного JavaScript. Также были установлены дополнительные компоненты: Three.js (для 3Д-моделей), React-Router-DOM (для замены обычному тегу <a> и реализации рендеринга выборочных компонентов), React-Redux-Toolkit (позволяет работать с обычным React, добавляя несколько инструментов, для удобной работы с ним).

# **3.1 Верстка программного продукта**

Верстка была сделана с пощью языка разметки данных HTML, с использованием JSX выражений, а также было сделано редактирование кода для корректной работы с ReactJS. Для примера, на рисунке 3.1 приведен код для навигации.

|  |
| --- |
| <nav className="main-nav">  <img className="nav-logo" src={nav\_logo} />  <div className="search-container2">  <img className="search\_icon" src={search\_icon} />  <input  className="search-input"  type="text"  placeholder="Поиск авто"  onChange={(event) => setSearchInput(event.target.value)}  onClick={searchBLockHide}  onBlur={searchBLockShow}  />  </div>  <div className="nav-content-container">  <Link to="/"><h2 className="nav-content">Главная</h2></Link>  <Link to="/auto"><h2 className="nav-content">Автомобили</h2></Link>  <a href="#contacts"><h2 className="nav-content">Контакты</h2></a>  </div>  </nav> |
| а |
| <div id="search-block" style={{display: searchShow ? 'none' : 'block'}} className="search-resault-container">  <h1 className="search-res-h1">Результат поиска: </h1>  <div className="search-resault-container2">  {CarsData.filter((val) => {  if (searchInput == "") {  return 0;  }  else if (val.name.toLowerCase().includes(searchInput.toLowerCase())) {  return val  }  }).map((val) => {  return (  <>  <div key={val.id} className="single-car22">  <Link to={val.linkto}><img className="search-car-img" src={val.img\_src} /></Link>  <h3 className="search-car-h3">{val.name}</h3>  </div>  </>  )})}  </div>  </div> |
| б |
| <Routes>  <Route path="/" element={<Mainpage />}></Route>  <Route path="/auto" element={<Auto />}></Route>  <Route path="/" element={<Mainpage />}></Route>    <Route path="/supra" element={<Supra />}></Route>  <Route path="/impala" element={<Impala />}></Route>  <Route path="/charger" element={<Charger />}></Route>  <Route path="/bugatti" element={<Bugatti />}></Route>  <Route path="/astonmartin" element={<AstonMartin />}></Route>  <Route path="/delorean" element={<Delorean />}></Route>  <Route path="/bmwnft" element={<BMWNFT />}></Route>  <Route path="/nfsbmw" element={<BMWNFS />}></Route>  <Route path="/nissangtr" element={<NissanGTR />}></Route>  </Routes> |
| в |
| <footer id="contacts" className="main-footer">  <p className="footer-p">  <strong>TradeCenter</strong> - хотя выглядит бесподобно, что я бы и сам вложил большие деньги в его развитие  и рекламу, является некоммерческим, сделанным специально в рамках курсовой работы  в 3 семестре учебы в учреждении БГТУ, проектом.  </p>  <div className="author">  <div className="design">Дизайн и реализация:</div>  <div className="me">Вакуленчик Владислав ИСиТ 2-1</div>  <div className="socials">  <a href="https://www.instagram.com/mega.vlad0n/"><img src={insta} /></a>  <a href="https://vk.com/itsvladiatorbeach"><img src={vk} /></a>  <a href="https://ladya2003.github.io/campfire/"><img src={campfire} /></a>  </div>  </div>  <hr className="hr-footer"/>  <p className="footer-copyright">  Copyright © 2022 Vladislav Vakulenchik. All rights reserved.  </p>  </footer> |
| г |

Рисунок 3.1 – примеры кода в приложении: а – навигация; б – блок поиска; в – роутинг, в котором расположены все основные компоненты

Код приложения тестировался с помощью интернет-сервиса Linter и успешно прошел проверку:

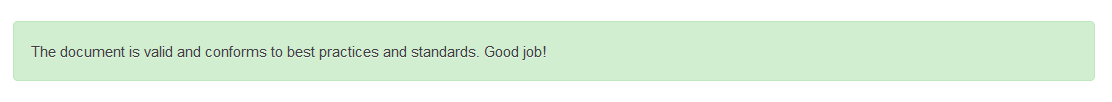


Рисунок 3.2 – Проверка на валидность через интернет-сервис Linter

# **3.3 Вывод**

В данном разделе была реализована структура курсового проекта на языке HTML, были подключены таблицы стилей CSS, которые были получены с помощью препроцессора Sass. Проект был протестирован на наличие синтаксических ошибок и успешно прошел все проверки.

# **4. Описание хода работы программного средства с копиями экрана пользовательского интерфейса основных элементов системы, а также ключевые функции разработанного программного кода**

**4.1 Руководство пользователя**

На главной странице приложения, сверху, находится меню, которое может перенести пользователя на любую страницу, которую он пожелает. Также в меню есть компонент с поиском для быстрого доступа к любому автомобилю. В заголовке приложения находится его название и самый знаменитый автомобиль салона Toyota Supra.

|  |
| --- |
|  |

Рисунок 4.1 – компонент навигации

Если спустится чуть ниже по главной странице, то можно увидеть информационный блок, который кратно описывает назначение и главные функции салона.

|  |
| --- |
|  |

Рисунок 4.2 – компонент информации о салоне

Далее идет блок с самыми популярными авто в салоне. При нажатии на каждую из которых, запускается ее 3Д-модель.

|  |
| --- |
|  |

Рисунок4.3 – популярные авто в салоне

Предпоследним элементом на главной странице, является форма заказа, любого понравившегося авто. Для ее отправки, пользователю необходимо заполнить поля Фамилия, Почта и выбрать модель автомобиля. Слева от формы находится 3Д-модель Айфона 13 версии.

|  |
| --- |
|  |

Рисунок 4.4 – форма заказа автомобиля

Последним компонентом на главной и всех остальных страницах является footer, в котором находится вся соответствующая информация про создателя и разработчика приложения, а также о способах с ним связаться.

|  |
| --- |
|  |

Рисунок 4.5 – footer приложения

Страница всех автомобилей вмещает в себе коллекцию всех доступных автомобилей, при наведении курсора на соответствующие картинки, появится текст «Больше информации», а при нажатии откроется страницы полной информации этой машины.

|  |
| --- |
| а    б |

Рисунок 4.6 – страница всех авто: а – пример без наведения курсора мыши; б – с наведением

Также на этой странице, есть блок с комментариями. В элемент input можно ввести информацию, которую пользователь хочет оставить на веб-сайте, а по нажатию на кнопку «Добавить», написанный комментарий разместиться в блоке ниже.

|  |
| --- |
| а  б |

Рисунок 4.7 – блок комментариев: а – блок без добавления комментариев; б – блок после добавления комментариев

Как говорилось ранее, по нажатию на изображение авто, идет переадресация на компонент полной информации о нем.

|  |
| --- |
| а    б |

Рисунок 4.8 – страница автомобиля Toyota Supra: a – ее верхняя часть; б - нижняя

# **4.2 Кроссбраузерное тестирование**

Приложение было протестировано на крассбраузерность в нескольких браузерах.

|  |
| --- |
| а    б  в |

Рисунок 4.9 – отображение приложения в разных браузерах: а – Google Chrome; б – Yandex Browser; в – MS Edge

# **4.3 Вывод**

В ходе работы данного раздела, была разработана подробная пользовательская инструкция, в которой описаны все элементы, с которыми он может взаимодействовать. Вместе с тем, было проведено тестирование приложения на кроссбраузерность. По итогам курсового проекта, получили адаптивное и удобное для пользования приложение.

# **Заключение**

Целью курсового проекта являлось создание приложения салона культовых авто с уникальным дизайном. Вначале разработки был поставлен ряд задач:

1. Просмотреть веб-сайты с аналогичным решением и похожей тематикой
2. Разработать макет и прототип в приложении Figma
3. Реализовать структуру веб-сайта на языке разметки данных HTML5 и JSX выражений
4. Реализовать стилевое оформление и, с помощью препроцессора Sass, сформировать CSS-файлы
5. Подключить css-модули к созданным HTML-файлам
6. Подключить пакеты Redux Toolkit, React Router DOM, Material Core, Three.js
7. С помощью media-запросов и FlexBox-верстки реализовать адаптивность веб-сайта для любых устройств
8. Протестировать веб-сайт

После проделанной работы, можно заявить, что все технические требования были выполнены и, в результате, получился много компонентное приложение, которое соответствует всем качествам современного автосалона: доступность, качество, компактность и удобство использования. Более того, приложение является полностью адаптивным как для мобильных, так и десктопных устройств и корректно отображается во всех браузерах.

# **Список использованных литературных источников**

1. Справочник по HTML [Электронный ресурс] – Режим доступа: [http://htmlbook.ru/html](http://htmlbook.ru/html/) – Дата доступа: 08.11.2022.
2. Sass [Электронный ресурс] – Режим доступа: [http://caniuse.com](https://vk.com/away.php?to=http%3A%2F%2Fcaniuse.com&cc_key=) – Дата доступа: 2.12.2022.
3. Шпаргалки [Электронный ресурс] – Режим доступа: [https://ruseller.com/htmlshpora.php](https://vk.com/away.php?to=https%3A%2F%2Fruseller.com%2Fhtmlshpora.php&cc_key=) – Дата доступа: 22.10.2022.

**Приложение**

**Приложение 1. Макет веб-страниц**

|  |
| --- |
| Рисунок 1.1 – Макет главной страницы  Рисунок 1.2 – Макет страница коллекции всех авто  **C:\Users\my_as\AppData\Local\Microsoft\Windows\INetCache\Content.Word\5.jpg**  Рисунок 1.3 – Макет страницы авто |

**Приложение 2. Прототип веб-страниц**

|  |
| --- |
| Рисунок 2.1 – Прототип главной страницы  Рисунок 2.2 – Прототип коллекции всех авто |
|  |

Рисунок 2.3 – Прототип страницы авто