|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**КУРСОВАЯ РАБОТА**

по дисциплине: Разработка клиентских частей интернет-ресурсов

по профилю:

направления профессиональной подготовки: 09.03.04 «Программная инженерия»

Тема: Клиентская часть интернет-ресурса «Мир автогонок»

Студент: Смирнов Андрей Юрьевич

Группа: ИКБО-30-21

Работа представлена к защите\_\_\_\_\_\_\_\_\_\_\_(дата)\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

(подпись и ф.и.о. студента)

Руководитель:

Работа допущена к защите\_\_\_\_\_\_\_\_\_\_\_(дата)\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

(подпись и ф.и.о. рук-ля)

Оценка по итогам защиты: \_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /

(подписи, дата, ф.и.о., должность, звание, уч. степень двух преподавателей, принявших защиту)

М. РТУ МИРЭА. 2022 г.

# Аннотация

В данной курсовой работе содержится:

9 изображений схем и скриншотов сайта;

использовано 15 информационных источников;

- содержится 37 страниц;

- содержится 13 листингов.

Перед началом разработки клиентской части интернет-ресурсов было проведено исследование похожих сайтов. Далее применялись современные технологии для разработки клиентской части интернет-ресурсов. Такие как: HTML, CSS и JavaScript.

Продукт был оптимизирован под экраны любых размеров в том числе и мобильных устройств.

# Содержание

# Перечень сокращений

РКЧИР – Разработка клиентских частей интернет ресурсов

HTML — Hypertext Markup Language.

CSS — Cascading Style Sheets.

Инициализация — присвоение переменой какого-либо значения.

Функция — фрагмент кода, к которому можно обратится из другой части программы.

Метод — код, содержащий инструкции для программы.

Массив — структура данных, хранящая значение у каждого из которых есть индекс.

События в JavaScript — определенное действие, которые было вызвано пользователем или браузером.

<div> — Тег <div> определяет любой отдельный блок содержимого и предназначен для деления веб-страницы на фрагменты. Вы можете объединить любой набор логически связанных элементов в единственном блоке <div>.

# Введение

Гоночная серия Formula 1 – самый популярный автоспорт в мире. За гонками следят миллионы людей со всего мира. Каждый год проходит больше 20 гонок, они проходят на всех континентах, кроме Антарктиды. В них участвуют 20 гонщиков. Это высококвалифицированные гонщики, они лучшие из лучших, чтобы попасть в Формулу, они с самого детства занимались картингом и другими гоночными сериями формулы(формула-2,3,4).

Первый чемпионат среди гонщиков прошёл в 1950 году. За эти 72 года очень многое поменялось. Регламент менялся столько раз, что не редки случаи, когда об нарушение зрители и команды узнают только по завершению гонки. Количество великих личностей, которые являются гонщиками данной серии впечатляет. Это Прост, Сенна, Феттель, Хэммильтон. Ну и конечно Шумахер, его фамилия стала нарицательной(означает быструю езду).

Этому популярному автоспорту и посвящен данный интернет-ресурс

# 1 Общие сведения

## Наименование клиентской части интернет-ресурса

В ходе курсовой работы разрабатывается клиентская часть интернет-ресурса «Мир автогонок» с использованием HTML, JavaScript, CSS

## Функциональное назначение как совокупность свойств клиентской части интернет-ресурса

?????

## Прикладное программное обеспечение, необходимое для разработки и функционирования клиентской части интернет-ресурса.

Visual Studio Code необходим для написание клиентской части. Он обладает всем необходимым функционалом для разработки проекта с использованием технологий HTML, JavaScript, CSS.

Необходимы браузеры для проверки корректного отображения клиентской части интернет-ресурса. Используется Google Chrome, а так же для контрольных проверок используется Firefox.

# 2.1 Анализ предметной области разрабатываемой клиентской части интернет-ресурса

Для написание работы необходимо изучить уже существующие решения, а конкретно примеры новостных сайтов по автогонкам. Мы рассмотрим лишь часть из них, а конкретно такие медиа-ресурсы как:

1. [https://www.championat.com](https://www.championat.com/auto/_f1.html)
2. [https://www.sports.ru](https://www.sports.ru/f1-championship/)
3. https://ru.motorsport.com
4. <https://www.f1news.ru/>
5. <https://www.f1-world.ru/?locale=ru>

## Подход больших медиаресурсов к контенту связанному с гоночным сериям.( championat.com, sports.ru)

Выделим небольшие подгруппы, такие сайты как championat.com и sports.ru являются большими медиаресурсами посвящённые разным видам спорта, имеющие сегменты посвящённые мотоспорту. Список гоночных серий ограничивается основными гоночными сериями. В основном такие серии как Formula-1 и младшие серии, Formula-Е, MotoGP, Дакар

Новости могут создавать сами пользователи и администрация сайта, имеется возможность оставить комментарий на данную новость или оставить реакцию с помощью эмодзи или up/dawn vote. Контент регулярно обновляется и редактируется администрацией.

Имеются блоги посвященные командам, гонщикам и руководителям команд, это реализуется с помощью тэга в самой новости т.е. наличие данного тэга позволяет новости попасть в блог знаменитости. Никакой подробной информации позволяющей вкратце ознакомится с гонщиками и командами нет, весь контент сконцентрирован в небольших новостных публикациях.

## Подход более профильных сайтов.(motorsport.com)

Главным отличием от больших медиаресурсов можно выделить наличие практически всех мировых гоночных серий, в некоторых из них русскоговорящая аудитория слабо заинтересована. Присутствуют W SERIES, INDYCAR, Дакар и другие.

Аналогично большим медиаресурсам, тут контент предоставляется авторами с помощью статей. Так же по тэгам можно найти интересующую вас информацию о команде, гонщике и другое.

Имеются разделы с фотографиями и видеофрагментами. Это позволяет фанатам насладиться лучшими моментами гонок.

## Подход нишевых изданий(f1-world.ru и f1news.ru)

Это маленькие ресурсы с меньшим трафиком людей, но более специфичным. На данные сайты люди заходят с целью узнать конкретную информацию об конкретной гоночной серии. Цель таких сайтов быть более качественными на фоне больших сайтов с помощью более качественной работе с конкретной гоночной серией.

Так на главной странице можно узнать практически всю имеющуюся информацию о следующем гран при т.е. даты, время, место, точное расписание, текущую таблицу личного и командного зачета и прочее. Так же имеются итоги прошедшего гран при, которое можно довольно подробно изучать.

Имеются исторические статьи и отдельные колонки, так же имеются форумы для любителей автогонок и прочий интерактив, который может заинтересовать пользователя.

## Общие особенности

* Взаимодействие пользователя с сайтом, таким как создание контента или же комментирования его.
* Предоставление информации в виде новостных публикаций на узкую тему.
* Наличие админов или же редакторов по контенту ведущих данные колонки или же новостную страницу.
* Общая информация о текущем гоночном сезоне, его результаты и прочее.
* Наличие контента вокруг самих гонок, информация об звездах и закулисье.

## Выводы.

Сайт будет посвящён конкретно гоночной серии Formula-1. Иметь информационные страницы про гонщиков и автоспорт в целом. Это позволит пользователю лучше познакомиться с Формулой.

Пользователю могут отслеживать интересующую их информацию, и описывать себя. Выбирать любимых гонщиков и команды.

Будет известна дата следующего Гран-При

# 2.2 Выбор технологий разработки клиентской части интернет-ресурса

В ходе разработки ресурса используются технологии HTML, JavaScript и CSS. Данные технологии были пройдены в ходе курса РКЧИР.

HTML - стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере. Его возможности достаточны для создания страниц с минимальной визуализацией и минимальным откликом на действия пользователя. Способы его применения достаточно обширны, инструмент используется повсеместно. В курсовой работе используется для создания веб-страниц и частично их функционирования, а также для того, чтобы связать результат применения указанных ниже инструментов.

CSS - формальный язык описания внешнего вида документов, написанного с использованием языка разметки. Он отвечает за визуальное оформление сайта, что в современной парадигме создания клиентских частей интернет-ресурсов необходимо. Используется в данной курсовой работе, так как с помощью него планируется придать сайту современный внешний вид (язык удобен, современен и поддерживается большинством браузеров). Его инструментария достаточно, чтобы создать задуманную внешнюю оболочку интернет-ресурсу.

JavaScript – охватывающий несколько парадигм язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Этот язык является стандартом для реализации программного кода клиентской части сайтов. Он удобен, практичен, распространён и позволяет реализовывать практически всё, что потребуется для функционирования страницы. При использовании дополнительных фреймворков (например, node.js) на нём возможно написать и серверную часть проекта. Используется в данной курсовой работе, так как удобен, является стандартом и идеально подходит под реализацию задуманного (в частности, благодаря поддержке объектно-ориентированной парадигмы программирования).

На языке HTML будет визуализированы страницы, с помощью CSS будет описан внешний вид страниц. События и прочее будет выполнено с помощью языка JavaScript

# 2.3 Создание веб-страниц клиентской части интернет-ресурса с использованием технологий HTML5, CSS3 и JavaScript

Данный интернет-ресурс имеет 5 веб-страниц, каждая из которых предоставляют пользователю информацию и взаимодействовать с нею

У всех страниц имеется общий элемент header(шапка сайта, рисунок 1) и общий элемент footer(рисунок 2)



Рисунок 1 — Вид элемента «header»

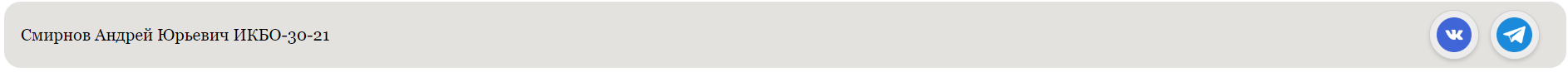


Рисунок 2 — Вид элемента «footer»

Веб-сайт имеет следующие веб-страницы:

1. Главная страница (На ней расположен таймер до следующей гонки, таблица лидеров командного и личного зачетов и различные новости или статьи на которые можно перейти)
2. Галерея (На ней отображаются фотографии связанные с тематикой сайта)
3. Гонщики(Позволяет познакомится с гонщиками, их именами, возрасто и прочее)
4. Статья(На ней хранится некоторая информация, с возможностью навигации по ней)
5. Топ(определенный топ с возможностью навигации)

# 2.4 Создание межстраничной навигации

С помощью ссылок на страницы в языке HTML5(т.е. тег <a>) реализовано навигационное меню в верхней части каждой из страниц. В случае маленького разрешения экрана ( рисунок 1 ), навигационное меню реализовано через выплывающий с боку блок ( рисунок 3).



Рисунок 3 – Навигационная панель при маленьком разрешении экрана

Благодаря навигационному меню возможен переход от каждой страницы к каждой, кроме статей. Что позволяет пользователю свободно перемещаться по сайту. Это так же является интуитивно понятно.

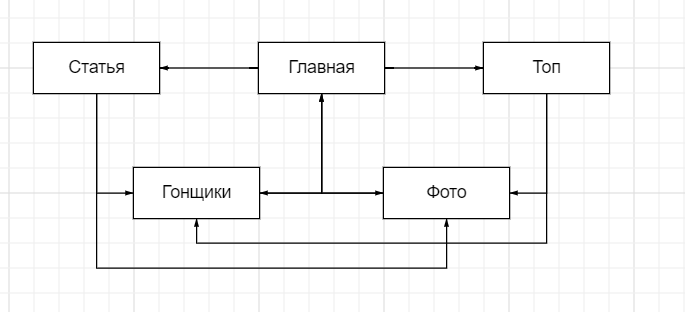


Рисунок 5 – Схема навигации интернет-ресурса

# 2.5 Реализация слоя клиентской логики веб-страниц с применением технологии JavaScript

В данном интернет-ресурсе JavaScript код используется для обработки данных и создание элементов HTML5 в веб-страницах. А так же с помощью событий, позволяет пользователю взаимодействовать с сайтом. Выбирать гонщика для ознакомления с ним(Рисунок 6). Реализация Листинг 1, 2

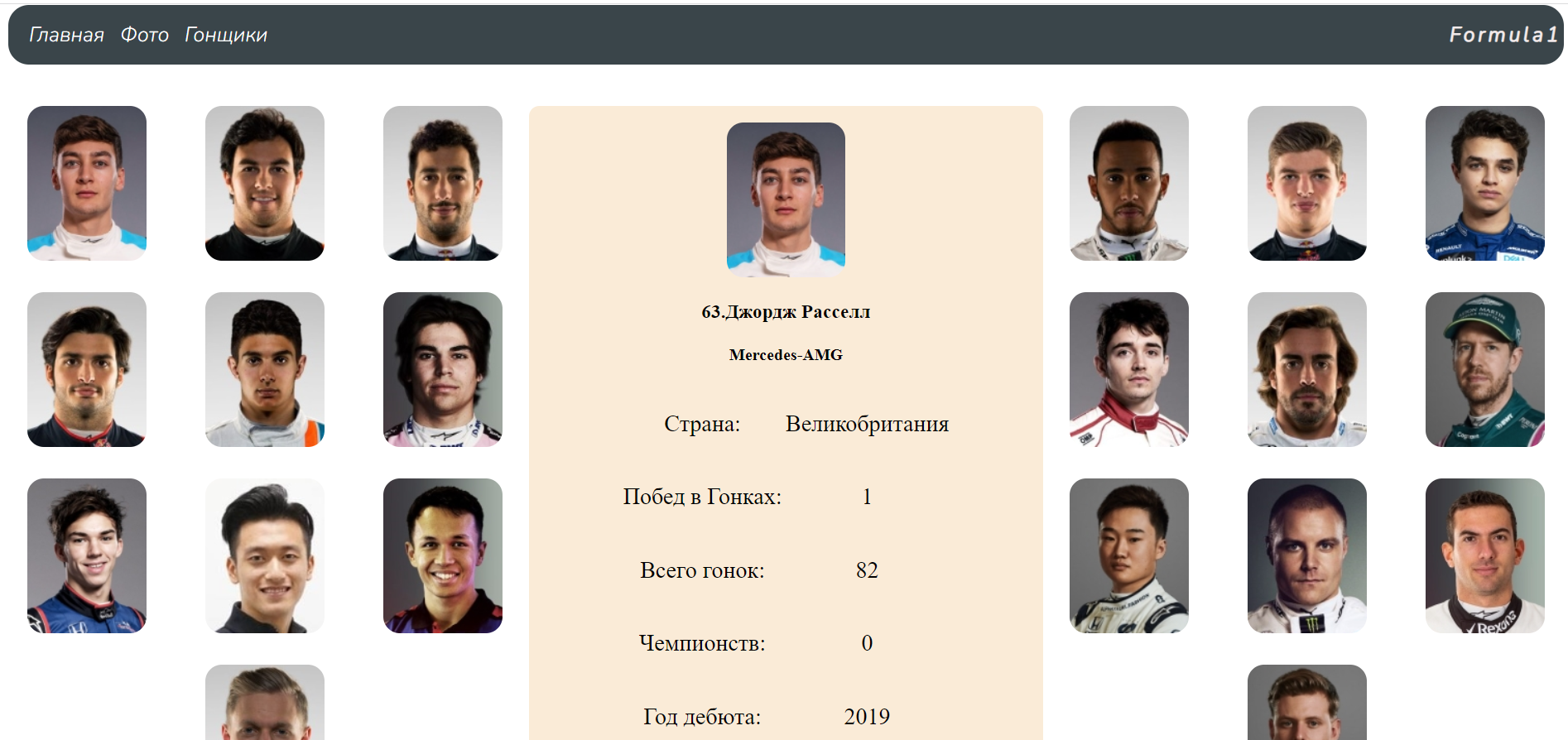


Рисунок 6 – Реализация ознакомления с гонщиками

Так же JavaScript используется для загрузки изображений на веб страницу и заполнение её новыми элементами.

Для реализации обратного отсчета так же используется JavaScript (Листинг 3). Результат продемонстрирован на рисунке 7.



Рисунок 7 – Таймер обратного отсчета

Вычисление таблицы лидеров в индивидуальном или командном зачете так же реализовано с помощью JavaScript (Листинг 4, 5). Результат продемонстрирован на рисунке 8.



Рисунок 8 – Топ гонщиков

# 3.1 Оптимизация клиентской части интернет-ресурса для основных браузеров и мобильных устройств

Веб-страницы оптимизированы под различные браузеры с помощью CSS стилей, мата тегами HTML или изменением содержания HTML5 страниц.

Сайт выглядит идентично как в браузере Google Chrome ,так и в Firefox (Рисунок 9).



Рисунок 9 – Вид сайта в браузере

Так же в зависимости от разрешения экрана меняется и сам сайт. Это позволяет не слишком уменьшать размеры на устройствах с маленьким экраном. Пример изображен на рисунке 10.



Рисунок 10 – вид сайта при меньшем разрешении экрана

Для совсем маленьких экранов или мобильных устройств размеры и расположение элементов меняется еще раз, а навигационная панель меняется на боковую панель. Пример приведен на рисунке 11.

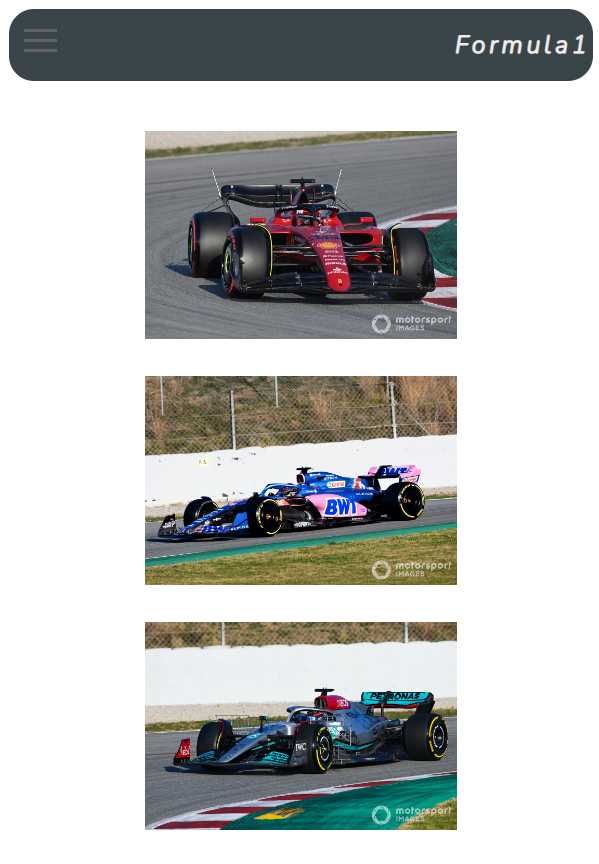


Рисунок 11 – Вид сайта при наименьшем размере

Изменение веб-страниц осуществляется с помощью медиа тегов(Листинг 5). Ключевыми размерами страницы являются: 1000 и 700 пикселей.

Для совсем малых экранов уместно убрать лишние элементы, чтобы увеличить полезное пространство. Такие примеры на рисунке 12 и 13.



Рисунок 12 – Пример веб-страницы для малых экранов

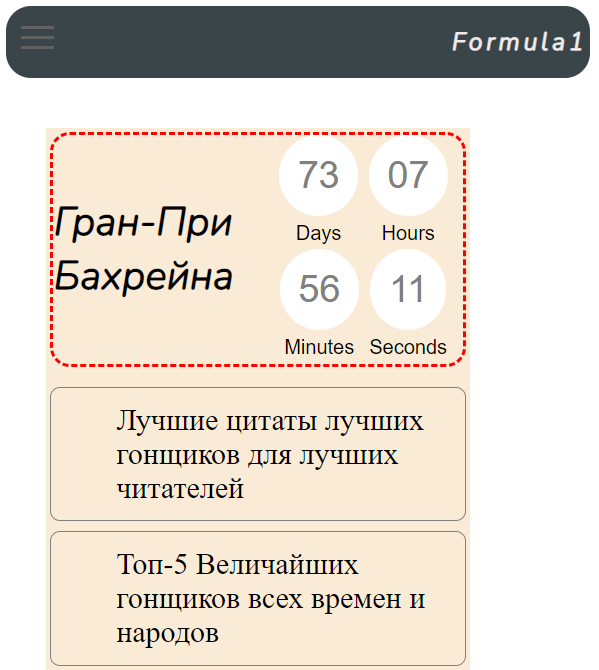


Рисунок 13 - пример главной страницы для маленьких экранов

Оптимизация под самый малый размер так же осуществлена с помощью медиа тегов. Так же были убраны боковые панели, а точнее их ширена изменена на 0, а параметр visible стал равен hidden т.е. элементы не видны.

Реализация приведена в листингах 6 и 7.

# Заключение

# Список источников

# Приложение

Листинг 1 – JavaScript файл для создания с веб-страницей “Гонщики”

const imageOut1 = document.querySelector('.drivers\_img1');

const imageOut2 = document.querySelector('.drivers\_img2');

const imageOut3 = document.querySelector('.drivers\_img3');

const driverName = document.querySelector('.name\_info');

const driverTeam = document.querySelector('.team\_info');

const driverCountry = document.querySelector('.country\_info');

const driverWins = document.querySelector('.wins\_info');

const driverExpirience = document.querySelector('.experience\_info');

const driverChampionships = document.querySelector('.championships\_info')

const driverDebut = document.querySelector('.debut\_info')

// объявлены все необходимые переменны, в которые записаны данные

let k = 1;

// оформление сайта, добавление новых элементов(изображений) на веб-страницу

for (let key in driver\_images){

    let img = document.createElement('img');

    img.setAttribute('data-key', key);

    img.classList = 'photo\_drivers'

    img.src = 'media/images/drivers2022/'+key+'.jpg';

    if(k%2 == 0){

        imageOut1.append(img)

    }

    else{

        imageOut2.append(img)

    }

    k++;

}

Листинг 2 – JavaScript файл для обработки событий на странице “Гонщики”

// обработчик события клика на картинку

imageOut1.addEventListener('click', showInfo);

imageOut2.addEventListener('click', showInfo);

let l = true;

function showInfo(){

    const key = event.target.dataset['key']

    if (key == undefined){

        return true;

    }

    let img = document.createElement('img');

    img.setAttribute('data-key', key);

    img.classList = 'photo\_drivers'

    img.id = 'center'

    img.src = 'media/images/drivers2022/'+key+'.jpg';

    if(l){

        // 1 добавление элемента

        img.id = 'center'

        imageOut3.append(img)

        let img\_block = img;

        l = false;

        document.getElementById('p1').style = "visibility: visible;"

        document.getElementById('p2').style = "visibility: visible;"

        document.getElementById('p3').style = "visibility: visible;"

        document.getElementById('p4').style = "visibility: visible;"

        document.getElementById('p5').style = "visibility: visible;"

    }

    else{

        //последующие добовления

        document.getElementById('center').src = 'media/images/drivers2022/'+key+'.jpg';

    }

    // заполнение данных

    driverName.textContent = driver\_images[key]['name']

    driverTeam.textContent = driver\_images[key]['team']

    driverCountry.textContent = driver\_images[key]['country']

    driverWins.textContent = driver\_images[key]['win']

    driverExpirience.textContent = driver\_images[key]['experience']

    driverChampionships.textContent = driver\_images[key]['championships']

    driverDebut.textContent = driver\_images[key]['Debut']

}

Листинг 3 – Реализация таймера

function getTimeRemaining(endtime) {

    var t = Date.parse(endtime) - Date.parse(new Date());

    var seconds = Math.floor((t / 1000) % 60);

    var minutes = Math.floor((t / 1000 / 60) % 60);

    var hours = Math.floor((t / (1000 \* 60 \* 60)) % 24);

    var days = Math.floor(t / (1000 \* 60 \* 60 \* 24));

    return {

      'total': t,

      'days': days,

      'hours': hours,

      'minutes': minutes,

      'seconds': seconds

    };

  }

  function initializeClock(id, endtime) {

    var clock = document.getElementById(id);

    var daysSpan = clock.querySelector('.days');

    var hoursSpan = clock.querySelector('.hours');

    var minutesSpan = clock.querySelector('.minutes');

    var secondsSpan = clock.querySelector('.seconds');

    function updateClock() {

      var t = getTimeRemaining(endtime);

      daysSpan.innerHTML = t.days;

      hoursSpan.innerHTML = ('0' + t.hours).slice(-2);

      minutesSpan.innerHTML = ('0' + t.minutes).slice(-2);

      secondsSpan.innerHTML = ('0' + t.seconds).slice(-2);

      if (t.total <= 0) {

        clearInterval(timeinterval);

      }

    }

    updateClock();

    var timeinterval = setInterval(updateClock, 1000);

  }

  var deadline="March 05 2023 20:00:00 GMT+0400";

  initializeClock('countdown', deadline);

Листинг 4 – Расчет и создание топа в индивидуальном зачете

let prev\_max = 10000

for(let i = 1; i <= 10; i++){

    let max = 0;

    let name = "";

    for(let key in driver\_images){

        if (max < parseInt(driver\_images[key]["points\_2022"]) && parseInt(driver\_images[key]["points\_2022"]) < prev\_max){

            max = parseInt(driver\_images[key]["points\_2022"]);

            name = driver\_images[key]['name'];

        }

    }

    prev\_max = max;

    document.getElementById(String(i)+'points').textContent = max

    document.getElementById(String(i)+'driver').textContent = name;

}

Листинг 5 – Расчет и создание топа в общем зачете

let prev\_max2 = 10000

for(let i = 1; i <= 10; i++){

    let max = 0;

    let name = "";

    for(let key in images){

        if (max < parseInt(images[key]["points\_2022"]) && parseInt(images[key]["points\_2022"]) < prev\_max2){

            max = parseInt(images[key]["points\_2022"]);

            name = images[key]['Name'];

        }

    }

    prev\_max2 = max;

    document.getElementById(String(i)+'team\_points').textContent = max

    document.getElementById(String(i)+'team').textContent = name;

}

Листинг 5 – Реализация медиа тэгов для 700px и 1000px и более

.photo\_gallery{

    max-width: 30%;

    min-width: none;

}

@media screen and (max-width: 1000px) {

    .photo\_gallery{

        width: 40%;

        min-width: 250px;

    }

}

@media screen and (max-width: 700px) {

    .photo\_gallery{

        width: 70%;

        min-width: 250px;

    }

}

Листинг 6 – @media для 700px на веб-странице “главная”

@media screen and (max-width: 700px) {

    .side\_info{

        width: 0%;

        min-width: 0;

        visibility: hidden;

    }

    .news\_block{

        width: 100%;

    }

}

Листинг 7 – @media для 750px на веб-странице с статьями

@media screen and (max-width: 750px) {

    .nav\_aside{

        visibility: hidden;

    }

    .content-600{

        width: 100%;

    }

    .col-3-4{

        width: 100%;

    }

    .col-1-4{

        width: 0%;

    }

}