

Pau Siquier del Rey
DAW MP9 Diseño de interfaces web
Actividad de desarrollo UF2
Fecha de entrega: 20 de abril 2022

Índice de contenido

| | |
|--|----|
| Comentario del archivo index.html..... | 3 |
| Fondo del juego..... | 3 |
| Coches..... | 3 |
| Botones..... | 4 |
| Menú desplegable..... | 4 |
| Tabla de resultados..... | 4 |
| Etiqueta script..... | 5 |
| Comentario del archivo styles.css..... | 5 |
| Fuente y efectos..... | 5 |
| Página principal..... | 6 |
| Efectos del loading | 7 |
| Menú desplegable..... | 8 |
| Botones..... | 10 |
| Carretera, línea de meta y coches..... | 11 |
| Clase hide..... | 11 |
| Tabla..... | 12 |
| Comentario del archivo app.js..... | 13 |
| Página principal..... | 13 |
| Funcionamiento del juego..... | 14 |
| Funcionalidad de los botones..... | 15 |
| Método animate y la tabla..... | 16 |
| Resumen..... | 17 |

Comentario del archivo index.html

En este apartado vamos a proceder a hacer los comentarios necesarios para la aclaración del código escrito en el archivo index.html. En este archivo se crea la estructura del juego y contiene elementos tales como: imágenes de coches, botones, menú desplegable, tabla, entre otros, según se haya pedido en el enunciado de la tarea por realizar.

Fondo del juego

Para crear un fondo de pantalla atractivo, antes de que se cargue el juego, introducimos unos gifs. Así creamos un efecto dinámico antes de que comience la carrera. Al mismo tiempo, aparecerá una barra que enseñará el progreso de cargar la página.

```
<body>
  <!--Fondo de pantalla-->
  <div class="fondo">
    
    <div class="intro">
      
      <div class="textoLoading">Loading...</div>
      <div id='barraLoading'>
        <div id='InternoLoading'></div>
      </div>
    </div>
  </div>
```

Coches

A continuación se importan las imágenes de los coches que sean participantes potenciales en el juego. Hemos decidido optar por el formato gif y así los coches no solo se desplazarán de izquierda a derecha sino también tendrán una animación propia.

```
</div>
<!--las imagenes de los coches-->
<div class="pista">
  <div class="carretera">
    <div class="coches">
      
      
      
      
      
      
      
      
      
    </div>
  </div>
</div>
</div>
<!--los botones-->
```

Botones

Creamos dos botones: iniciar y reiniciar. También añadimos otro gif que aparecerá junto a los resultados al final del juego.

```
<!--los botones-->
<div class="juego">
  <div class="menu">
    <div class="contenedorBotones">
      <div id="patan"></div>
      <button id="reiniciar" class="hide">Reiniciar</button>
    </div>
    <button id="inicio">Inicio</button>
  </div>
</div>
```

Menú desplegable

Para crear un menú desplegable utilizamos la etiqueta `<select>`. Todas las opciones que se pueden elegir tendrán un `value` que nos dirá que número se ha elegido.

```
51 <!--menu desplegable-->
52 <label id="etiqueta"><span>Numero de participantes</span>
53 <select id="participantes">
54   <option value="1">1</option>
55   <option value="2">2</option>
56   <option value="3">3</option>
57   <option value="4">4</option>
58   <option value="5">5</option>
59   <option value="6">6</option>
60   <option value="7">7</option>
61   <option value="8">8</option>
62   <option value="9">9</option>
63 </select>
64 </label>
65 </div>
66 </div>
```

Tabla de resultados

Para crear la tabla que almacenará los resultado utilizamos la etiqueta `<table>`. Contendrá dos columnas: Posición y Coche. El interior de la tabla se llenará de datos dinámicamente mientras los coches vayan llegando a la línea de meta.

```
66 <!--la tabla-->
67 <div id="tabla" class="hide">
68   <table id="posiciones">
69     <thead>
70       <tr>
71         <th>Posicion</th>
72         <th>Coche</th>
73       </tr>
74     </thead>
75     <tbody>
76     </tbody>
77   </table>
78 </div>
79 </div>
80 </div>
81 <!--El archivo js-->
82 <script src="app.js"></script>
83 </body>
84
85 </html>
```

Etiqueta script

Al final del documento, antes de cerrar la etiqueta <body>, añadimos la etiqueta <script> para enlazar el archivo de JavaScript.

```
0      <!--El archivo js-->
1      <script src="app.js"></script>
2  </body>
```

Comentario del archivo styles.css

En este apartado vamos a proceder a hacer los comentarios necesarios para la aclaración del código escrito en el archivo styles.css. En este archivo se trabaja la parte visual del juego. Se añaden colores, animaciones y se posicionan los elementos.

Fuente y efectos

Primero que todo importamos una fuente específica. Lo hacemos con dos archivos para que pueda estar soportada por distintos navegadores. Decidimos los estilos de la fuente implementada.

A continuación definimos los efectos que se utilizarán en diferentes imágenes del juego como pueden ser el avance de la barra del loading o el parpadeo del logo principal.

```
1  /*Importar archivo de la fuente desde la carpeta. Hay dos para que funcione con cualquier navegador*/
2  @font-face {
3      font-family: Autoslocos;
4      src: url('assets/font/RodfellowsWacky.woff') format('woff'),
5           url('assets/font/RodfellowsWacky.woff2') format('woff2');
6      font-weight: normal;
7      font-style: normal;
8      font-display: swap;
9  }
10 /*Keyframes para nuestros efectos*/
11 @keyframes progreso {
12     100% {
13         width: 300px;
14     }
15 }
16 /*El logo parpadea*/
17 @keyframes blink {
18     to {
19         visibility: hidden;
20     }
21 }
22 @-webkit-keyframes blink {
23     to {
24         visibility: hidden;
25     }
26 }
```

Creamos un efecto especial para un coche que se irá desplazando en la página principal de izquierda a derecha y lo hará al mismo ritmo que el avance de la barra de loading.

```

27  /*El coche de Pier se desplaza en la pagina de loading*/
28  ∨ @-webkit-keyframes piercorre {
29  ∨    0% {
30      left:0;
31      -webkit-transform:rotateY(180deg);
32  }
33  ∨    50% {
34      left:75%;
35      -webkit-transform:rotateY(180deg);
36  }
37  ∨    100% {
38      left:0;
39      -webkit-transform:rotateY(0deg);
40  }
41  }

```

Página principal

Para comenzar a darle estilos a la página principal reseteamos los márgenes y el padding y definimos el tamaño de la páginas como el 100% de altura.

Importamos la imagen que nos servirá de fondo de pantalla. Su tamaño va a ocupar el tamaño exacto de la ventana en la que se visualice. La imagen se quedará en el mismo lugar y se extenderá para ajustarse a la pantalla, sin repetirse y siempre quedando centrada.

```

50  /*-----Web-----*/
51
52  body, html {
53      height: 100%;
54      margin: 0px;
55      padding: 0px;
56      border: 0px;
57  }
58  .fondo {
59      background-image: url('assets/img/autoslocos_fondo.jpg');
60      position: fixed;
61      width: 100vw;
62      height: 100vh;
63      background-position: center center;
64      background-repeat: no-repeat;
65      background-size: cover;
66  }
67

```

Efectos del loading

Procedemos aplicándole estilos a los demás elementos que se visualizan mientras esperamos a que comience el juego. Primero nos ocupamos del logo que tendrá una animación de parpadeo que ya hemos mencionado más arriba. Posicionamos el elemento en la página.

```
68  /*-----Loading-----*/
69
70  .logoLoading {
71      margin: 0;
72      position: absolute;
73      top: 22%;
74      left: 47%;
75      transform: translate(-50%, -50%);
76      animation: blink 2s steps(2, start) infinite;
77      -moz-animation: blink 2s steps(2, start) infinite;
78      -o-animation: blink 2s steps(2, start) infinite;
79      -webkit-animation: blink 2s steps(2, start) infinite;
80  }
81
```

Además, posicionamos un coche en la parte inferior de la página, donde irá desplazándose utilizando otra de las animaciones mencionadas.

```
75  ✓ .logoPierro {
76      margin: 0;
77      position: absolute;
78      bottom: -5px;
79      width: 30%;
80      -webkit-animation: linear infinite;
81      -webkit-animation-name: piercorre;
82      -webkit-animation-duration: 11s ;
83  }
```

Las propiedades del texto y de la barra del loading se definen a continuación. El texto recibe el color, el tamaño y la posiciones que hemos considerado adecuados. Se le aplica también una sombra para que quede más integrado con el resto de los elementos que lo rodean.

En cuanto a la barra de progreso, aparte de definir su color y tamaño, cabe destacar la sombra que se aplica a su interior. Con la propiedad `box-shadow: inset` conseguimos que el interior quede más oscurecido y opaco y así le damos una apariencia más creíble.

```

92  .textoLoading {
93      font-family: Autoslocos;
94      color: ■ #f0e000;
95      font-size: 40px;
96      position: absolute;
97      top: 43%;
98      left: 40%;
99      text-shadow: 0 0 20px □ rgb(0, 0, 0);
100 }
101
102 #barraLoading {
103     width: 300px;
104     height: 30px;
105     border: 7px solid ■ #f0e000;
106     border-radius: 3px;
107     position: absolute;
108     top: 50%;
109     left: 40%;
110     box-shadow: 0 0 50px □ rgb(0, 0, 0) ;
111     box-shadow: inset 0px 0px 50px □ rgba(0, 0, 0, 0.501);
112 }
113

```

La parte dinámica de la barra de progreso se presenta a continuación. El progreso es el nombre de la animación definida al principio del documento. La utilizamos ahora para visualizar el avance de la carga.

```

114 #InternoLoading {
115     width: 1px;
116     height: 30px;
117     background-color: ■ rgb(108, 0, 196);
118     animation: progreso 5s linear;
119     top: 50%;
120     left: 40%;
121 }

```

Menú desplegable

Seguidamente definimos los estilos que aplicaremos al menú desplegable. Posicionamos todos los elementos que lo constituyen y aplicamos la fuente que previamente importamos.


```

123  /*-----inicio del juego-----*/
124
125  label {
126      position: absolute;
127      top: 50%;
128      left: 25%;
129      justify-content: center;
130      align-items: center;
131      display: block;
132  }
133
134  select {
135      display: block;
136      left: 41.8%;
137      position: absolute;
138  }
139
140  #etiqueta {
141      font-family: Autoslocos;
142      font-size: 40px;
143      letter-spacing: 2px;
144      color: white;
145      text-shadow: 0 0 20px rgb(0, 0, 0);
146      left: 33%;
147  }
148

```

Cuando el menú se despliegue, el fondo del selector permanecerá de un color definido. Entre otras, seguimos utilizando la propiedad border-radius y box-shadow para que quede más integrado.

```

149  #participantes {
150      font-family: Autoslocos;
151      background-color: rgb(198, 129, 255);
152      border: solid 4px rgb(81, 0, 147);
153      border-radius: 10px;
154      width: 150px;
155      height: 50px;
156      color: white;
157      font-size: 30px;
158      box-shadow: 0 0 20px rgb(0, 0, 0);
159      left: 40.5%;
160  }
161

```

Botones

Los dos botones que creamos van a compartir muchas de las propiedades como el tipo, el tamaño y el color de la letra, el espacio entre ellas, el color del fondo, el estilo del borde y otras. Por esta razón se las definimos de forma conjunta agrupando los selectores y así evitamos repetir el código. El cursor cambia a pointer cuando pasa por encima del botón. De esta manera el usuario queda avisado que es un elemento que se puede pulsar.

```
162  #inicio, #reiniciar {
163      font-family: Autoslocos;
164      letter-spacing: 2px;
165      background-color: #c09cf2;
166      border: solid 4px #800080;
167      border-radius: 10px;
168      font-size: 40px;
169      margin-top: 30px;
170      color: white;
171      box-shadow: 0 0 20px #000000;
172      text-align: center;
173      transition: all .35s;
174      cursor: pointer;
175  }
176
```

Sin embargo, algunos valores son diferentes entre los dos botones. Sus tamaños y posiciones varían. Lo mostramos con la siguiente captura de pantalla.

```
177  #inicio {
178      width: 150px;
179      height: 80px;
180      top: 60%;
181      left: 44.8%;
182      position: absolute;
183  }
184
185  #reiniciar {
186      width: 250px;
187      height: 80px;
188      display: block;
189      position: fixed;
190      top: 60%;
191      left: 48%;
192      transform: translate(-50%, -50%);
193  }
```

Al mismo tiempo, aplicamos una pseudoclase :hover. Cuando el usuario pase por encima del botón con el cursor, se producirán cambios visuales. Aquí también agrupamos los selectores.



Carretera, línea de meta y coches

Con la propiedad de background-image importamos una imagen que será nuestra carretera. Esta ocupará el 95% del ancho de la pantalla. De igual manera, importamos otra imagen para el espacio restante que será el que se corresponda al lugar donde los coches quedarán parados una vez terminada la carrera.

Por otro lado, los coches se ubican uno por debajo del otro y se modifica su tamaño.

```
202  ∨ .pista {
203      background-image: url(assets/img/meta.gif);
204      position: absolute;
205      width: 100%;
206      box-shadow: 0 0 20px rgb(0, 0, 0);
207  }
208
209  ∨ .coche {
210      width: 79.5px;
211      display: block;
212  }
213
214  ∨ .carretera {
215      border-right: 15px solid rgb(132, 0, 147);
216      width: 95%;
217      background-image: url(assets/img/desierto.jpg);
218      background-repeat: no-repeat;
219      background-size: cover;
220  }
221
```

Clase hide

En el archivo html hemos creado la clase hide que aplicamos a varios elementos. Esta clase nos sirve para mostrarlos o esconderlos en distintos momentos del juego según las pautas del enunciado de la tarea. Para conseguir este efecto, aplicamos la propiedad display con el valor none a dicha clase. Le añadimos una anotación adicional !important y así conseguimos que ningún otro estilo más adelante pueda sobrescribirse sobre ella. Dicho de otra manera, los elementos con la clase hide permanecerán ocultos aunque en el archivo css en otro lugar se defina de otra manera.

```

222     .hide {
223     |     display: none !important;
224     }

```

Tabla

Continuamos dándole estilo a la tabla. Utilizamos los mismos colores y las mismas fuentes que en los elementos anteriores. La situamos en el lado inferior derecho de la página. Redondeamos sus bordes y quitamos los bordes que podrían aparecer dobles con la propiedad border-collapse. Cualquier elemento que sobresalga, se quedará oculto. Los datos que contenga la tabla más adelante se centrarán.

```

232  #tabla {
233  |     font-family: Autoslocos;
234  |     color: ■ #fff;
235  |     font-size: 20px;
236  |     letter-spacing: 3px;
237  |     margin: 0 auto;
238  |     margin-left: 80%;
239  |     position: absolute;
240  |     bottom: 2%;
241  |     border-collapse: collapse;
242  |     border-spacing: 1;
243  |     border-radius: 10px;
244  |     background: ■ rgb(132, 0, 147);
245  |     overflow: hidden;
246  | }
247
248  th {
249  |     padding: 10px;
250  | }
251
252  td {
253  |     text-align: center;
254  |     background: ■ rgb(198, 129, 255);
255  | }
256

```



| Posicion | Coche |
|----------|--------|
| 1 | coche4 |
| 2 | coche8 |
| 3 | coche7 |
| 4 | coche1 |
| 5 | coche3 |
| 6 | coche6 |
| 7 | coche9 |
| 8 | coche2 |
| 9 | coche5 |

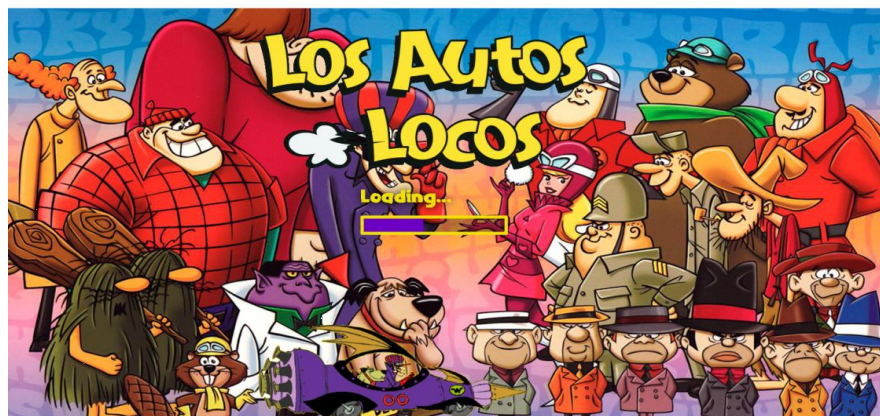
Comentario del archivo app.js

En este apartado vamos a proceder a hacer los comentarios necesarios para la aclaración del código escrito en el archivo app.js. En este archivo se trabaja la parte lógica y dinámica del juego. Aquí es donde conseguimos que se enlace el número de coches elegidos con el número de coches mostrados en la pantalla. Creamos las funciones necesarias para que los coches avancen a una velocidad aleatoria y llenamos la tabla dinámicamente con los resultados de la carrera. A continuación, explicaremos estas funcionalidades en más detalle.

Página principal

Antes de que aparezca la página principal con el menú desplegable, durante 5 segundos tenemos una página que simula la carga del juego. Cuando finaliza este tiempo, pasamos a la pantalla del juego en sí.

```
1  /*-----Loading----- */
2
3  //En 5 segundos aparece el juego
4  $("#patan").addClass("hide");
5  $(".juego").addClass("hide");
6  setTimeout(function () {
7      $(".intro").addClass("hide");
8  }, 5000);
9  setTimeout(function () {
10     $(".juego").removeClass("hide");
11 }, 5000);
12
```



Funcionamiento del juego

Una vez en la página principal, el jugador elije el número de participantes. El valor correspondiente a la elección que haya hecho se almacenará y se utilizará en un bucle. Al recorrer el bucle, almacenamos en un array los coches con sus correspondientes velocidades. La velocidad es un valor aleatorio de entre 1-10. Al pulsar en el menú desplegable, aparte de lo mencionado ya, ocultamos el logo del juego. Cada vez que se realice una elección nueva los coches desaparecen y vuelve a aparecer las cantidad seleccionada.

```
16  /*Al apretar en el numero de participantes recibimos el valor
17  del numero de coches que se usara en el bucle, apareceran en la pantalla
18  y se almacenaran en el array junto a sus velocidades*/
19  var coches = [];
20  $("#participantes").change(function () {
21      coches = [];
22      $(".coche").addClass("hide");
23      $(".logoLoading").addClass("hide");
24      var numeroDeCoches = $("#participantes").val();
25      for (var i = 1; i <= numeroDeCoches; i++) {
26          var actual = "coche" + i;
27          $(".coche[data-id='" + actual + "']").removeClass("hide");
28          coches.push({
29              "cocheActual": actual,
30              "velocidad": Math.floor(Math.random() * (11 - 1) + 1)
31          });
32      }
33  })
```



Funcionalidad de los botones

La principal funcionalidad de los botones consiste en que al pulsarlos desaparezcan o aparezcan los elementos que necesitamos. De este modo cuando apretamos en el botón de inicio, este mismo junto al menú desplegable desaparecen. Al mismo tiempo aparecen el botón de reinicio, la tabla y, a pocos segundos, un gif.

```
35  /* Al presionar el boton de inicio se ocultan o aparecen los elementos que necesitamos*/
36  $("#inicio").click(function () {
37      $("#reiniciar").removeClass("hide");
38      $("#tabla").removeClass("hide");
39      setTimeout(function () {
40          $("#patan").removeClass("hide");
41      }, 6500);
42      $("#etiqueta").addClass("hide");
43      $("#inicio").addClass("hide");
44  });
```



Al apretar el botón de reinicio, este mismo junto a la tabla y el gif desaparecen. El contenido de la tabla se vacía. Aparecen de nuevo el botón de inicio y el menú desplegable. Usando el método `animate` devolvemos los coches a la línea del principio de la carrera.

```
60  /* Al presionar el boton de reinicio se ocultan o aparecen los elementos que necesitamos
61  Los coches vuelven al punto de salida y la tabla se vacia*/
62  $("#reiniciar").click(function () {
63      $("#reiniciar").addClass("hide");
64      $("#tabla").addClass("hide");
65      $(".losResultados").addClass("hide");
66      $("#patan").addClass("hide");
67      $("#inicio").removeClass("hide");
68      $("#etiqueta").removeClass("hide");
69      $("#inicio").removeClass("hide");
70      $(".coche").animate({
71          marginLeft: "0"
72      });
73      $("#posiciones > tbody").html("");
74  });
```

Método `animate` y la tabla

Para conseguir que los coches se desplacen e imitar una carrera utilizamos el método `animate`. Cada coche avanzará hasta alcanzar el margen derecho de la ventana. Almacenamos los resultados de la carrera en una tabla según vayan llegando los coches a la meta. Añadimos la posición que iremos incrementando con cada coche y el nombre del coche.

```
45  /* Se anima a los coches y una vez llegan a la meta se introduce su posicion en la tabla*/
46  var posicion = 1;
47  coches.forEach(function (coche) {
48      $(".coche[data-id='" + coche["cocheActual"] + "']").animate({
49          marginLeft: "100%"
50      }, coche["velocidad"] * 1000, function () {
51          $('#posiciones tbody').append('<tr><td>' + posicion + '</td><td>' + coche["cocheActual"] + '</td></tr>');
52          posicion = posicion + 1;
53      });
54  });
55  });
```




Resumen

Con el presente escrito hemos procurado explicar detalladamente el funcionamiento de un juego de carreras de coches. Hemos cumplido con todos los requisitos técnicos y hemos implementado todas las funcionalidades que se solicitaban haciéndonos servir de HTML, CSS y JavaScript. Asimismo, nuestro objetivo ha sido conseguir un efecto visual agradable y funcional para los usuarios.

Webgrafia:

www.codecademy.com

www.stackoverflow.com

www.w3schools.com

<https://www.youtube.com/c/HolaMundoDev>