

**JOHAN ALEJANDRO MARTÍNEZ MARTÍNEZ**  
OCTUBRE DE 2020

# 1 CONTENIDO

---

<a href="#">1</a>	<a href="#">CONTENIDO</a>	<a href="#">1</a>
<a href="#">2</a>	<a href="#">PRESENTACIÓN</a>	<a href="#">2</a>
<a href="#">3</a>	<a href="#">FASE 1: Dibujar y mover una bola</a>	<a href="#">3</a>
<a href="#">4</a>	<a href="#">FASE 2: Rebotando en las paredes</a>	<a href="#">6</a>
<a href="#">5</a>	<a href="#">FASE 3: Control de la pala y el teclado</a>	<a href="#">9</a>
<a href="#">6</a>	<a href="#">FASE 4: Fin del juego</a>	<a href="#">18</a>
<a href="#">7</a>	<a href="#">FASE 5: Muro de ladrillos</a>	<a href="#">22</a>
<a href="#">8</a>	<a href="#">FASE 6: Detección de colisiones</a>	<a href="#">28</a>
<a href="#">9</a>	<a href="#">FASE 7: Contar puntos y ganar</a>	<a href="#">33</a>
<a href="#">10</a>	<a href="#">FASE 8: Controlando el ratón</a>	<a href="#">39</a>
<a href="#">11</a>	<a href="#">FASE 9: Finalizando el juego</a>	<a href="#">45</a>
<a href="#">12</a>	<a href="#">CONCLUSIONES</a>	<a href="#">51</a>
<a href="#">13</a>	<a href="#">BIBLIOGRAFÍA</a>	<a href="#">52</a>

## 2 PRESENTACIÓN

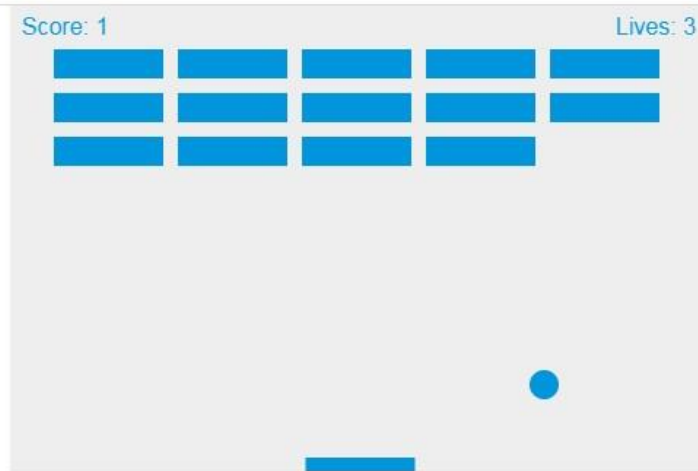
---

La presente monografía describe el desarrollo metódico de un juego 2D elaborado utilizando HTML5, CSS, CANVAS y JavaScript.

El juego elaborado se crea con JavaScript puro, utilizando un enfoque metódico en el cual se avanza de versión en versión, de modo que cada nuevo programa abarca un aspecto adicional del juego.

Cada una de las fases se cubre en un apartado diferente. Se plantea el alcance de cada una de ellas, se explican las instrucciones o conceptos que son necesarios para entender el significado del trabajo realizado, se agrega el código, y finalmente se presentan fotos de la ejecución del programa

Una vez cubiertas todas las fases, se dispondrá de un clásico juego 2D que servirá como base e inspiración para desarrollar otros programas aplicados en la Web.



### Gráfica 1. Juego 2D en JavaScript

El documento web que sirve como referencia para el desarrollo del juego está en el siguiente enlace:

[https://developer.mozilla.org/es/docs/Games/Workflows/Famoso\\_juego\\_2D\\_usando\\_JavaScript\\_puro](https://developer.mozilla.org/es/docs/Games/Workflows/Famoso_juego_2D_usando_JavaScript_puro)

**AUTOR: JOHAN ALEJANDRO MARTÍNEZ**

### 3 FASE 1: DIBUJAR Y MOVER UNA BOLA

En esta fase básicamente se crea una página HTML normal, pero en esta se crea el elemento **CANVAS**, el cual será clave para desarrollar el juego. Dentro del canvas que se separa en `<script>` `</script>` irá las funciones principales como sería la de **dibujar la bola()**, la que imprime sobre la pantalla la bola, la de **dibujar()**, que es la que se encarga de ir limpiando el canvas e ir actualizando las coordenadas donde se ubicara la bola, y por último **setInterval**, esta regula la velocidad con la que se desplazara la bola por medio de los milisegundos con los que se dibujara la pelota.

Sin olvidar que podríamos modificar más cosas como, colores tanto del canvas como de bola y el tamaño del canvas.

A continuación, el código:

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4.     <meta charset="utf-8" />
5.     <title>Juego-Fase_1</title>
6.
7.     <!-- Define los estilos de la interfaz visual
8.         padding es la distancia de un objeto en relación con
9.         el marco que lo contiene
10.        margin es la distancia que separa a un objeto de
11.        otro
12.        background es el color de fondo
13.        display: block; Estos elementos fluyen hacia
14.        abajo
15.        margin: 0 auto; Centra el canvas en la
16.        pantalla -->
17.     <style>
18.         * {
19.             padding: 0;
20.             margin: 0;
21.         }
22.         canvas {
23.             background: #EB4A0E;
24.             display: block;
25.             margin: 0 auto;
26.         }
27.     </style>
28. </head>
29. <body>
30.     <canvas id="miCanvas" width="600" height="450"></canvas>
```

## 30. &lt;script&gt;

```
31.     var canvas = document.getElementById("miCanvas");
32.     var ctx = canvas.getContext("2d");
33.
34.
35.     var x = canvas.width/2;
36.
37.
38.     var y = canvas.height-30;
39.
40.
41.     var dx = 30;
42.     var dy = -30;
43.
44.     function dibujarBola() {
45.
46.         ctx.beginPath();
47.
48.         ctx.arc(x, y, , 10, Math.PI*2);
49.
50.
51.         ctx.fillStyle = "#47EB0E";
52.
53.
54.         ctx.fill();
55.
56.
57.         ctx.closePath();
58.     }
59.
60.
61.     function dibujar() {
62.
63.         ctx.clearRect(0, 0, canvas.width, canvas.height)
64.     ;
65.
66.         dibujarBola();
67.
68.         x = x + dx;
69.
70.
71.         y = y + dy;
72.
73.
74.
75.         setInterval(dibujar, 300);
76.     </script>
77.
78. </body>
79. </html>
```

Al ejecutarse el juego, la interfaz sería algo tal que así:



## 4 FASE 2: REBOTANDO EN LAS PAREDES

---

Esta fase es similar a la anterior, pero lo que se le agregan los límites los cuales la bola no pueda pasar, haciendo entonces que rebote y que se mantenga dentro del canvas, pero para que esto suceda se agrega una nueva variable llamada **ballRadius()**, que es el tamaño de la bola, esto nos servirá crear dos condiciones para que se cumple la función antes mencionada.

En la primera condición se crea el límite de movimiento de la bola en vertical:

```
if(y + dy > canvas.height-ballRadius || y + dy < ballRadius) {dy = -dy;}
```

Y en la última condición se crea el límite de movimiento de la bola en horizontal:

if(x + dx > canvas.width-ballRadius || x + dx < ballRadius) { dx = -dx;}

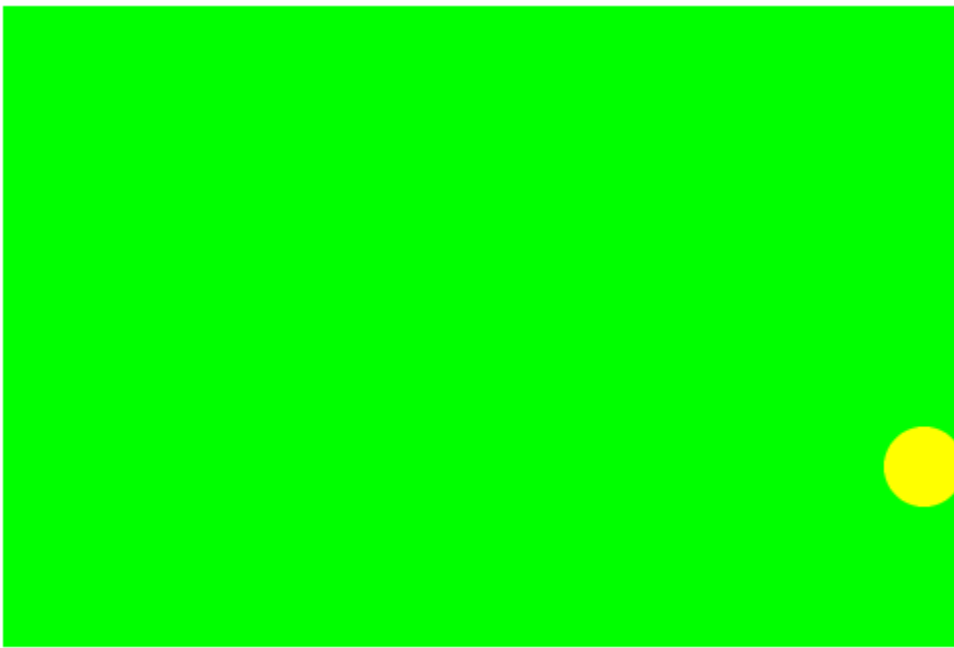
```
1. <!DOCTYPE html>
2.
3. <html>
4.
5. <head>
6.
7. <meta charset="utf-8" />
8.
9.
10.    <title>JUEGO-FASE 2</title>
11.
12.
13.    <style>
14.        * { padding: 0; margin: 0; }
15.
16.
17.        canvas { background: #00ff00; display: block; margin: 0 auto; }
18.
19.
20.    </style>
21. </head>
22. <body>
23.
24.
25.
26.                                     <canvas
id="miCanvas" width="480" height="320"> </canvas>
27.
28.
29.
30.
31.    <script>
32.
33.
34.
35.
36.        var canvas = document.getElementById("miCanvas");
37.
38.        var ctx = canvas.getContext("2d");
39.
40.
41.
42.
```

```
43.
44.         var ballRadius = 20;
45.
46.         var x = canvas.width / 2;
47.
48.         var y = canvas.height - 30;
49.
50.
51.
52.         var dx = 2;
53.
54.         var dy = -2;
55.
56.         function dibujarBola() {
57.
58.
59.             ctx.beginPath();
60.
61.             ctx.arc(x, y, ballRadius, 0, Math.PI*2);
62.
63.             ctx.fillStyle = "#ffff00";
64.
65.             ctx.fill();
66.
67.
68.
69.             ctx.closePath();
70.         }
71.
72.
73.         function dibujar() {
74.
75.
76.             ctx.clearRect(0, 0, canvas.width, canvas.height);
77.
78.
79.             dibujarBola();
80.
81.             if (x + dx > canvas.width -
ballRadius || x + dx < ballRadius) { /
82.                 dx = dx * (-1);
83.             }
84.
85.
86.             if(y + dy > canvas.height - ballRadius |
| y + dy < ballRadius) {
87.                 dy = dy * (-1);
88.             }
89.
90.
91.             x = x + dx;
92.             y = y + dy;
93.         }
```



```
94.  
95.  
96.     setInterval(dibujar, 10);  
97.  
98.  
99.     </script>  
100.  
101.  
102.  
103.  
104.     </body>  
105.  
106.  
107.     </html>
```

Al ejecutar el programa de vería algo así:



## 5 FASE 3: CONTROL DE LA PALA Y EL TECLAD

---

En la esta fase si comienza a tomar algo de sentido el juego, añadiéndole como sería una paleta con la que la pelota pueda interactuar de manera que, cuando la bola toque a la barra esta se cambiara de orientación.

Para esto se deben crear dos variables que serán las que controlen el movimiento de la paleta por medio del teclado; estas tendrán el nombre de: flechaDerechaPulsada y flechalzquierdaPulsada. Después de crea una función que va regular cuando se presiona o no la tecla.

El código a continuación:

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4.     <meta charset="utf-8" />
5.     <title>Juego-fase 3</title>
6.     <style>
7.         * { padding: 0; margin: 0; }
8.         canvas { background: #0E50BC; display: block; margin:
0 auto; }
9.     </style>
10. </head>
11. <body>
12.
13.     <canvas id="miCanvas" width="500" height="350"></canvas>
14.
15.     <script>
16.         var canvas = document.getElementById("miCanvas");
17.         var ctx = canvas.getContext("2d");
18.
19.
20.         var radioBola = 8;
21.
22.         var x = canvas.width/2;
23.         var y = canvas.height-30;
24.         var dx = 2;
25.         var dy = -2;
26.
27.
```

```

28.
29.     var alturaPaleta = 12;
30.     var anchuraPaleta = 60;
31.     var paletaPosX = (canvas.width - anchuraPaleta) / 2;

32.
33.     var flechaDerechaPulsada = false;
34.     var flechaIzquierdaPulsada = false;
35.
36.
37.     document.addEventListener("keydown", manejadorTeclaP
resionada, false);
38.     document.addEventListener("keyup", manejadorTeclaLib
erada, false);
39.
40.
41.     function manejadorTeclaPresionada(e) {
42.         if(e.keyCode == 39) {
43.
44.             flechaDerechaPulsada = true;
45.         }
46.         else if(e.keyCode == 37) {
47.
48.             flechaIzquierdaPulsada = true;
49.         }
50.     }
51.
52.
53.     function manejadorTeclaLiberada(e) {
54.         if(e.keyCode == 39) {
55.
56.             flechaDerechaPulsada = false;
57.         }
58.         else if(e.keyCode == 37) {
59.
60.             flechaIzquierdaPulsada = false;
61.         }
62.     }
63.
64.
65.     function dibujarBola() {
66.         ctx.beginPath();
67.         ctx.arc(x, y, radioBola, 0, Math.PI*2);
68.         ctx.fillStyle = "#EE650B";
69.         ctx.fill();
70.         ctx.closePath();
71.     }
72.
73.     function dibujarPaleta() {

```

```

74.         ctx.beginPath();
75.
76.         ctx.rect(paletaPosX, canvas.height-
alturaPaleta, anchuraPaleta, alturaPaleta);
77.         ctx.fillStyle = "#331E1E";
78.         ctx.fill();
79.
80.         ctx.closePath();
81.     }
82.
83.
84.     function dibujar() {
85.         ctx.clearRect(0, 0, canvas.width, canvas.height)
;
86.
87.
88.         dibujarBola();
89.
90.
91.         dibujarPaleta();
92.
93.
94.
95.
96.         if(x + dx > canvas.width-
radioBola || x + dx < radioBola) {
97.             dx = -dx;
98.         }
99.         if(y + dy > canvas.height-
radioBola || y + dy < radioBola) {
100.             dy = -dy;
101.         }
102.
103.
104.         if(flechaDerechaPulsada && paletaPosX < canvas.w
idth-anchuraPaleta) {
105.             paletaPosX += 7;
106.         }
107.         else if(flechaIzquierdaPulsada && paletaPosX > 0
) {
108.             paletaPosX -= 7;
109.         }
110.
111.         x += dx;
112.         y += dy;
113.     }
114.
115.
116.     setInterval(dibujar, 5);

```

```
117.     </script>
118.
119.     </body>
120.     </html>
```

Y así se vería una vez ejecutado:



## 6 FASE 4: FIN DEL JUEGO

---

En esta fase lo que se hará es que el programa detecte cuando la bola toca la parte inferior, para que cuando lo haga salte la alerta de que has perdido; esto se logra verificando si algunos datos son ciertos y poder saltar la alerta, por eso nos fijaremos en dos códigos:  $(y + dy > \text{canvas.height} - \text{radioBola})$ ,  $(x > \text{paletaPosX} \ \&\& \ x < \text{paletaPosX} + \text{anchuraPaleta})$ ; el primero es para saber si la bola toca la parte inferior y la segunda es para saber si la paleta interfirió con la bola. Así sabremos cuando la bola toca lo inferior sin que la paleta la toque y se podrá saltar el aviso de que has perdido.

El código se muestra a continuación:

```

1.<!DOCTYPE html>
2.<html>
3.<head>
4.    <meta charset="utf-8" />
5.    <title>Juego 2D - #04 - Game Over</title>
6.    <style>* { padding: 0; margin: 0; } canvas { bac
kground: #154AC4; display: block; margin: 0 auto; }<
/style>
7.</head>
8.<body>
9.
10.    <canvas
id="miCanvas" width="500" height="350"></canvas>
11.
12.    <script>
13.
14.
15.        var canvas = document.getElementById("miCan
vas");
16.        var ctx = canvas.getContext("2d");
17.
18.        var radioBola = 9;
19.        var x = canvas.width/2;
20.        var y = canvas.height-30;
21.        var dx = 2;

```

```

22.         var dy = -2;
23.
24.         var alturaPaleta = 12;
25.         var anchuraPaleta = 60;
26.         var paletaPosX = (canvas.width-
           anchuraPaleta)/2;
27.
28.         var flechaDerechaPresionada = false;
29.         var flechaIzquierdaPresionada = false;
30.
31.         document.addEventListener("keydown", manejadorTeclaPresionada, false);
32.         document.addEventListener("keyup", manejadorTeclaLiberada, false);
33.
34.         function manejadorTeclaPresionada(e) {
35.             if(e.keyCode == 39) {
36.                 flechaDerechaPresionada = true;
37.             }
38.             else if(e.keyCode == 37) {
39.                 flechaIzquierdaPresionada = true;
40.             }
41.         }
42.         function manejadorTeclaLiberada(e) {
43.             if(e.keyCode == 39) {
44.                 flechaDerechaPresionada = false;
45.             }
46.             else if(e.keyCode == 37) {
47.                 flechaIzquierdaPresionada = false;
48.             }
49.         }
50.
51.         function dibujarBola() {
52.             ctx.beginPath();
53.             ctx.arc(x, y, radioBola, 0, Math.PI*2);
54.             ctx.fillStyle = "#F92615";
55.             ctx.fill();
56.             ctx.closePath();
57.         }
58.         function dibujarPaleta() {
59.             ctx.beginPath();

```

```

60.         ctx.rect(paletaPosX, canvas.height-
    alturaPaleta, anchuraPaleta, alturaPaleta);
61.         ctx.fillStyle = "#050202";
62.         ctx.fill();
63.         ctx.closePath();
64.     }
65.
66.     function dibujar() {
67.         ctx.clearRect(0, 0, canvas.width, canva
    s.height);
68.
69.         dibujarBola();
70.         dibujarPaleta();
71.
72.         if(x + dx > canvas.width-
    radioBola || x + dx < radioBola) {
73.             dx = -dx;
74.         }
75.         if(y + dy < radioBola) {
76.             dy = -dy;
77.         }
78.
79.
80.         else if(y + dy > canvas.height-
    radioBola) {
81.             if(x > paletaPosX && x < paletaPosX
    + anchuraPaleta) {
82.                 dy = -dy;
83.             }
84.             else {
85.                 clearInterval(juego);
86.                 alert("has perdido");
87.                 document.location.reload();
88.             }
89.         }
90.
91.         if(flechaDerechaPresionada && paletaPos
    X < canvas.width-anchuraPaleta) {
92.             paletaPosX += 7;
93.         }

```

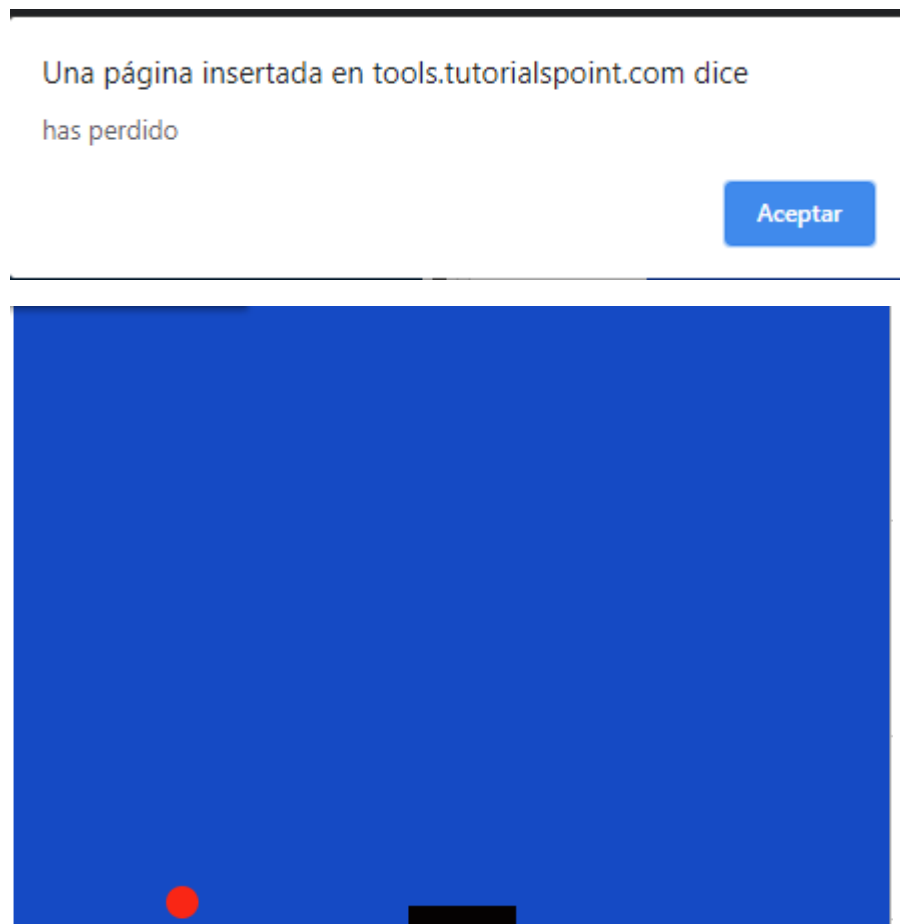


```

94.         else if(flechaIzquierdaPresionada && pa
    letaPosX > 0) {
95.             paletaPosX -= 7;
96.         }
97.
98.         x += dx;
99.         y += dy;
100.    }
101.
102.
103.    var juego = setInterval(dibujar, 8);
104. </script>
105.
106. </body>
107. </html>

```

Y así se ve el juego:



## 7 FASE 5: MURO DE LADRILLOS

---

En la fase 5 se incorporan muros los que la bola podrá interactuar, para esto se agregaran en el programa las variables que son básicamente:

El número de columnas, de filas, el ancho y la altura de los ladrillos, relleno ladrillo, vacío superior ladrillo y vacío izquierdo ladrillo; con esto y la función dibujar que será más adelante se podrá visualizar correctamente los ladrillos.

Este es el código:

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4.     <meta charset="utf-8" />
5.     <title>Juego-fase 5</title>
6.     <style>* { padding: 0; margin: 0; } canvas { background:
    #1A51BD; display: block; margin: 0 auto; }</style>
7. </head>
8. <body>
9.
10.     <canvas id="miCanvas" width="600" height="450"></canvas>
11.
12.     <script>
13.         var canvas = document.getElementById("miCanvas");
14.         var ctx = canvas.getContext("2d");
15.         var radioBola = 8;
16.         var x = canvas.width/2;
17.         var y = canvas.height-30;
18.         var dx = 2;
19.         var dy = -2;
20.         var alturaPaleta = 12;
21.         var anchuraPaleta = 60;
22.         var paletaPosX = (canvas.width-anchuraPaleta)/2;
23.         var flechaDerechaPresionada = false;
24.         var flechaIzquierdaPresionada = false;
```

```

25.
26.
27.     var nroFilasLadrillos = 8;
28.     var nroColumnasLadrillos = 8;
29.     var anchoLadrillo = 60;
30.     var alturaLadrillo = 17;
31.     var rellenoLadrillo = 10;
32.     var vacioSuperiorLadrillo = 30;
33.     var vacioIzquierdoLadrillo = 30;
34.
35.
36.     var ladrillos = [];
37.
38.
39.     for(var columna=0; columna<nroColumnasLadrillos; col
40.     umna++) {
41.         ladrillos[columna] = [];
42.
43.
44.         for(var fila=0; fila<nroFilasLadrillos; fila++)
45.         {
46.
47.
48.
49.             ladrillos[columna][fila] = { x: 0, y: 0 };
50.         }
51.     }
52.
53.     document.addEventListener("keydown", manejadorTeclaP
54.     resionada, false);
55.     document.addEventListener("keyup", manejadorTeclaLib
56.     erada, false);
57.
58.     function manejadorTeclaPresionada(e) {
59.         if(e.keyCode == 39) {
60.             flechaDerechaPresionada = true;
61.         }
62.         else if(e.keyCode == 37) {
63.             flechaIzquierdaPresionada = true;
64.         }
65.     }
66.
67.     function manejadorTeclaLiberada(e) {
68.         if(e.keyCode == 39) {
69.             flechaDerechaPresionada = false;
70.         }
71.         else if(e.keyCode == 37) {
72.             flechaIzquierdaPresionada = false;

```

```

70.         }
71.     }
72.
73.     function dibujarBola() {
74.         ctx.beginPath();
75.         ctx.arc(x, y, radioBola, 0, Math.PI*2);
76.         ctx.fillStyle = "#DF1C0C";
77.         ctx.fill();
78.         ctx.closePath();
79.     }
80.
81.     function dibujarPaleta() {
82.         ctx.beginPath();
83.         ctx.rect(paletaPosX, canvas.height-
alturaPaleta, anchuraPaleta, alturaPaleta);
84.         ctx.fillStyle = "#130301";
85.         ctx.fill();
86.         ctx.closePath();
87.     }
88.
89.
90.     function dibujarLadrillos() {
91.
92.         for(var columna=0; columna<nroColumnasLadrillos;
columna++) {
93.
94.             for(var fila=0; fila<nroFilasLadrillos; fila
++) {
95.
96.                 var brickX = (fila*(anchoLadrillo+rellen
oLadrillo))+vacioIzquierdoLadrillo;
97.
98.
99.                 var brickY = (columna*(alturaLadrillo+re
llenoLadrillo))+vacioSuperiorLadrillo;
100.
101.
102.                 ladrillos[columna][fila].x = brickX;
103.
104.
105.                 ladrillos[columna][fila].y = brickY;
106.
107.
108.                 ctx.beginPath();
109.                 ctx.rect(brickX, brickY, anchoLadrillo,
alturaLadrillo);
110.
111.                 ctx.fillStyle = "#FEFDFD";
112.                 ctx.fill();

```

```

113.             ctx.closePath();
114.
115.
116.         }
117.     }
118. }
119.
120.     function dibujar() {
121.         ctx.clearRect(0, 0, canvas.width, canvas.height)
122.         ;
123.
124.         dibujarLadrillos();
125.
126.         dibujarBola();
127.         dibujarPaleta();
128.
129.         if(x + dx > canvas.width-
radioBola || x + dx < radioBola) {
130.             dx = -dx;
131.         }
132.         if(y + dy < radioBola) {
133.             dy = -dy;
134.         }
135.         else if(y + dy > canvas.height-radioBola) {
136.             if(x > paletaPosX && x < paletaPosX + anchur
aPaleta) {
137.                 dy = -dy;
138.             }
139.             else {
140.                 clearInterval(juego);
141.                 alert("has perdido");
142.
143.
144.                 document.location.reload();
145.             }
146.         }
147.
148.         if(flechaDerechaPresionada && paletaPosX < canva
s.width-anchuraPaleta) {
149.             paletaPosX += 7;
150.         }
151.         else if(flechaIzquierdaPresionada && paletaPosX
> 0) {
152.             paletaPosX -= 7;
153.         }
154.
155.         x += dx;
156.         y += dy;

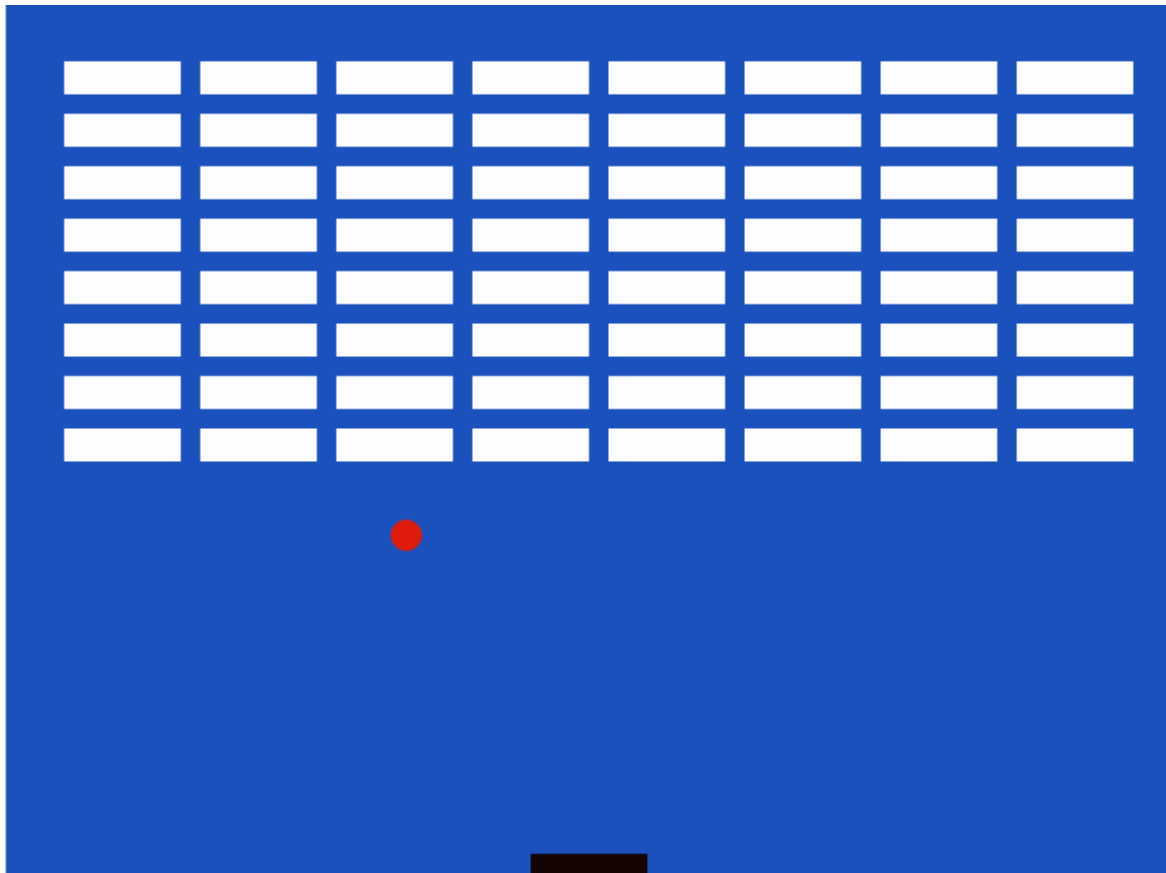
```

```

157.     }
158.
159.     var juego = setInterval(dibujar, 8);
160.   </script>
161.
162. </body>
163. </html>

```

Y al ejecutarlo se vería así:



## 8 FASE 6: DETECCIÓN DE COLISIONES

---

En esta parte del juego se empieza a ver las completo, y lo que sigue a continuación es hacer que cuando bola pase por uno de los ladrillos impresos este desaparezca detectando la colisión entre ambos; esto se puede hacer insertando en el código una función que sea capaz de detectar dicho suceso.

sin olvidar que también se deben crear algunas variables para que complemente aún más, y son básicamente la de va a regular el ciclo del juego, la alerta cuando pierde y por último la que se encarga de regresar al juego inicial una vez se pierda

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4.     <meta charset="utf-8" />
5.     <title>Juego-fase 6</title>
6.     <style>* { padding: 0; margin: 0; } canvas { background:
7.         #000000; display: block; margin: 0 auto; }</style>
8. </head>
9. <body>
10.     <canvas id="miCanvas" width="600" height="450"></canvas>
11.
12.     <script>
13.         var canvas = document.getElementById("miCanvas");
14.         var ctx = canvas.getContext("2d");
15.
16.         var radioBola = 8;
17.         var x = canvas.width/2;
18.         var y = canvas.height-30;
19.         var dx = 2;
20.         var dy = -2;
21.
22.         var alturaPaleta = 12;
23.         var anchuraPaleta = 60;
24.         var paletaPosX = (canvas.width-anchuraPaleta)/2;
25.
26.         var flechaDerechaPresionada = false;
27.         var flechaIzquierdaPresionada = false;
28.
29.         var nroFilasLadrillos = 8;
30.         var nroColumnasLadrillos = 8;
31.         var anchuraLadrillo = 60;
32.         var alturaLadrillo = 15;

```

```

33.     var rellenoLadrillo = 10;
34.     var vacioSuperiorLadrillo = 30;
35.     var vacioIzquierdoLadrillo = 30;
36.
37.     var ladrillos = [];
38.     for(var c=0; c<nroColumnasLadrillos; c++) {
39.         ladrillos[c] = [];
40.         for(var f=0; f<nroFilasLadrillos; f++) {
41.
42.             ladrillos[c][f] = { x: 0, y: 0, status: 1 };
43.         }
44.     }
45.
46.
47.     document.addEventListener("keydown", manejadorTeclaP
resionada, false);
48.     document.addEventListener("keyup", manejadorTeclaLib
erada, false);
49.
50.     function manejadorTeclaPresionada(e) {
51.         if(e.keyCode == 39) {
52.             flechaDerechaPresionada = true;
53.         }
54.         else if(e.keyCode == 37) {
55.             flechaIzquierdaPresionada = true;
56.         }
57.     }
58.
59.     function manejadorTeclaLiberada(e) {
60.         if(e.keyCode == 39) {
61.             flechaDerechaPresionada = false;
62.         }
63.         else if(e.keyCode == 37) {
64.             flechaIzquierdaPresionada = false;
65.         }
66.     }
67.
68.
69.
70.     function deteccionColision() {
71.
72.
73.         for(var c=0; c<nroColumnasLadrillos; c++) {
74.             for(var f=0; f<nroFilasLadrillos; f++) {
75.
76.
77.                 var b = ladrillos[c][f];
78.
79.

```



```

80.         if(b.status == 1) {
81.
82.
83.
84.             if(x > b.x && x < b.x+anchuraLadrillo
o && y > b.y && y < b.y+alturaLadrillo) {
85.                 dy = -dy;
86.                 b.status = 0;
87.             }
88.         }
89.     }
90. }
91.
92.
93.     function dibujarBola() {
94.         ctx.beginPath();
95.         ctx.arc(x, y, radioBola, 0, Math.PI*2);
96.         ctx.fillStyle = "#F70B0B";
97.         ctx.fill();
98.         ctx.closePath();
99.     }
100.
101.     function dibujarPaleta() {
102.         ctx.beginPath();
103.         ctx.rect(paletaPosX, canvas.height-
alturaPaleta, anchuraPaleta, alturaPaleta);
104.         ctx.fillStyle = "#FF00FF";
105.         ctx.fill();
106.         ctx.closePath();
107.     }
108.
109.     function dibujarLadrillos() {
110.         for(var c=0; c<nroColumnasLadrillos; c++) {
111.             for(var f=0; f<nroFilasLadrillos; f++) {
112.
113.
114.
115.                 if(ladrillos[c][f].status == 1) {
116.
117.
118.                     var brickX = (f*(anchuraLadrillo+rel
lenoLadrillo))+vacioIzquierdoLadrillo;
119.                     var brickY = (c*(alturaLadrillo+rell
enoLadrillo))+vacioSuperiorLadrillo;
120.                     ladrillos[c][f].x = brickX;
121.                     ladrillos[c][f].y = brickY;
122.                     ctx.beginPath();
123.                     ctx.rect(brickX, brickY, anchuraLadr
illo, alturaLadrillo);

```

```

124.         ctx.fillStyle = "#FD72D3";
125.         ctx.fill();
126.         ctx.closePath();
127.     }
128. }
129. }
130. }
131.
132.     function dibujar() {
133.         ctx.clearRect(0, 0, canvas.width, canvas.height)
134.         ;
135.         dibujarLadrillos();
136.         dibujarBola();
137.         dibujarPaleta();
138.         deteccionColision();
139.         if(x + dx > canvas.width-
radioBola || x + dx < radioBola) {
140.             dx = -dx;
141.         }
142.         if(y + dy < radioBola) {
143.             dy = -dy;
144.         }
145.         else if(y + dy > canvas.height-radioBola) {
146.             if(x > paletaPosX && x < paletaPosX + anchur
aPaleta) {
147.                 dy = -dy;
148.             }
149.             else {
150.
151.                 clearInterval(juego);
152.
153.                 alert("has perdido");
154.
155.                 document.location.reload();
156.             }
157.         }
158.
159.         if(flechaDerechaPresionada && paletaPosX < canva
s.width-anchuraPaleta) {
160.             paletaPosX += 7;
161.         }
162.         else if(flechaIzquierdaPresionada && paletaPosX
> 0) {
163.             paletaPosX -= 7;
164.         }
165.
166.         x += dx;
167.         y += dy;

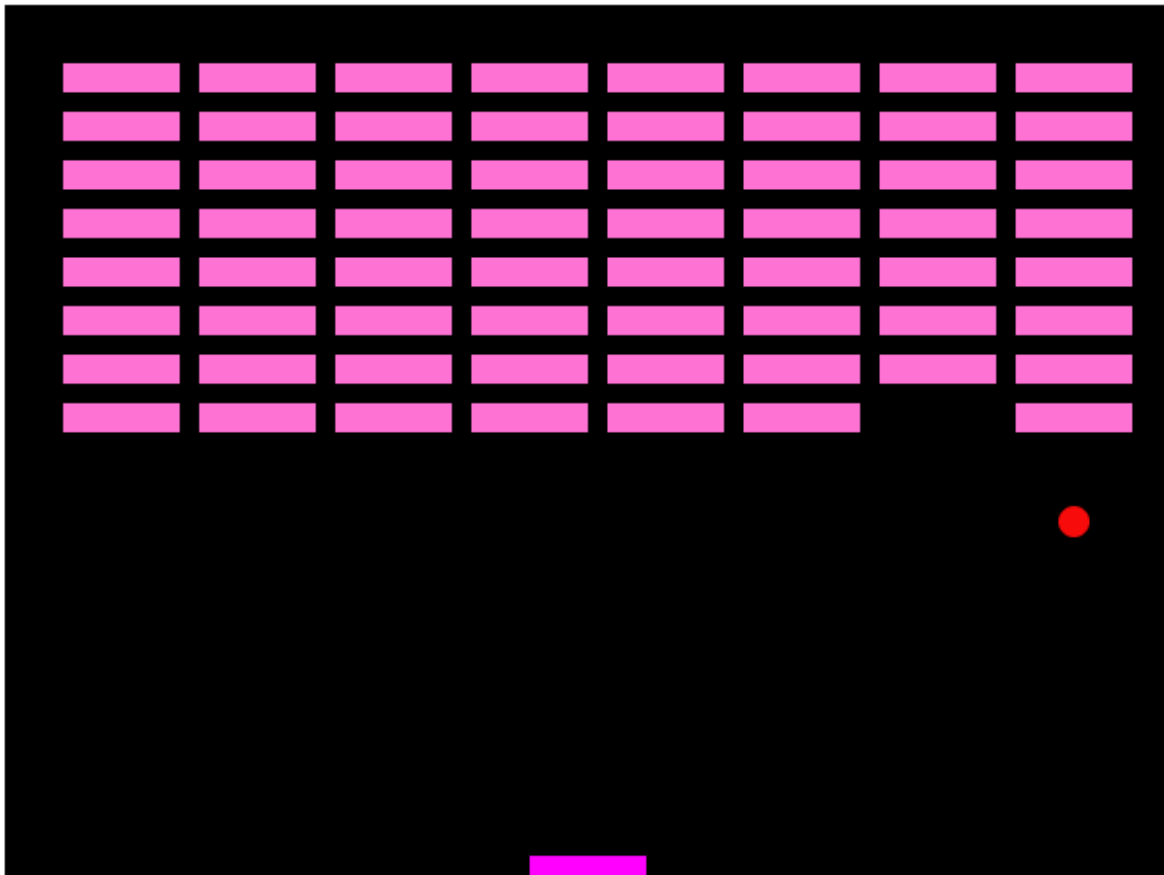
```

```

168.     }
169.
170.     var juego = setInterval(dibujar, 8);
171. </script>
172.
173. </body>
174. </html>

```

Y su respectiva interfaz al ejecutarlo:



## 9 FASE 7: CONTAR PUNTOS Y GANAR

Después de todo lo agregado hasta ahora sigue la parte del puntaje, esto se hace creando una variable que lo que hará es que cada vez que le demos a un ladrillo

con la bola el puntaje vaya aumentando teniendo en cuenta la cantidad de ladrillos que haya para que cuando llegue a esa cantidad el juego diga que gano.

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4.     <meta charset="utf-8" />
5.     <title>Juego 2D - #07 - Control de juego ganado</title>
6.
7.     <style>* { padding: 0; margin: 0; } canvas { background:
8.         #000000; display: block; margin: 0 auto; }</style>
9. </head>
10. <body>
11.     <canvas id="miCanvas" width="600" height="450"></canvas>
12.
13.     <script>
14.         var canvas = document.getElementById("miCanvas");
15.         var ctx = canvas.getContext("2d");
16.
17.         var radioBola = 8;
18.         var x = canvas.width/2;
19.         var y = canvas.height-30;
20.         var dx = 2;
21.         var dy = -2;
22.         var alturaPaleta = 12;
23.
24.
25.         var anchuraPaleta = 600;
26.
27.         var paletaPosX = (canvas.width-anchuraPaleta)/2;
28.         var flechaDerechaPresionada = false;
29.         var flechaIzquierdaPresionada = false;
30.
31.         var nroFilasLadrillos = 8;
32.         var nroColumnasLadrillos = 8;
33.         var anchuraLadrillos = 60;
34.         var alturaLadrillos = 15;
35.         var rellenoLadrillos = 10;
36.         var vacioSuperiorLadrillo = 30;
37.         var vacioIzquierdoLadrillo = 30;
38.
39.
40.         var puntaje = 0;
41.
42.         var ladrillos = [];
43.         for(var c=0; c<nroColumnasLadrillos; c++) {
44.             ladrillos[c] = [];

```

```

45.         for(var f=0; f<nroFilasLadrillos; f++) {
46.             ladrillos[c][f] = { x: 0, y: 0, estado: 1 };
47.         }
48.     }
49.
50.     document.addEventListener("keydown", manejadorTeclaP
resionada, false);
51.     document.addEventListener("keyup", manejadorTeclaLib
erada, false);
52.
53.     function manejadorTeclaPresionada(e) {
54.         if(e.keyCode == 39) {
55.             flechaDerechaPresionada = true;
56.         }
57.         else if(e.keyCode == 37) {
58.             flechaIzquierdaPresionada = true;
59.         }
60.     }
61.     function manejadorTeclaLiberada(e) {
62.         if(e.keyCode == 39) {
63.             flechaDerechaPresionada = false;
64.         }
65.         else if(e.keyCode == 37) {
66.             flechaIzquierdaPresionada = false;
67.         }
68.     }
69.     function detectarColision() {
70.         for(var c=0; c<nroColumnasLadrillos; c++) {
71.             for(var f=0; f<nroFilasLadrillos; f++) {
72.                 var b = ladrillos[c][f];
73.                 if(b.estado == 1) {
74.                     if(x > b.x && x < b.x+anchuraLadrill
os && y > b.y && y < b.y+alturaLadrillos) {
75.                         dy = -dy;
76.                         b.estado = 0;
77.
78.
79.
80.                         puntaje++;
81.                         if(puntaje == nroFilasLadrillos*
nroColumnasLadrillos) {
82.                             alert("ENHORABUENA, HAS
GANADO!!");
83.                             document.location.reload();
84.                         }
85.                     }
86.                 }
87.             }
88.         }

```

```

89.         }
90.     }
91.
92.     function dibujarBola() {
93.         ctx.beginPath();
94.         ctx.arc(x, y, radioBola, 0, Math.PI*2);
95.         ctx.fillStyle = "#0095DD";
96.         ctx.fill();
97.         ctx.closePath();
98.     }
99.
100.    function dibujarPaleta() {
101.        ctx.beginPath();
102.        ctx.rect(paletaPosX, canvas.height-
    alturaPaleta, anchuraPaleta, alturaPaleta);
103.        ctx.fillStyle = "#FF01BA";
104.        ctx.fill();
105.        ctx.closePath();
106.    }
107.
108.    function dibujarLadrillos() {
109.        for(var c=0; c<nroColumnasLadrillos; c++) {
110.            for(var r=0; r<nroFilasLadrillos; r++) {
111.                if(ladrillos[c][r].estado == 1) {
112.                    var posXLadrillo = (r*(anchuraLadril
    los+rellenoLadrillos))+vacioIzquierdoLadrillo;
113.                    var posYLadrillo = (c*(alturaLadrill
    os+rellenoLadrillos))+vacioSuperiorLadrillo;
114.                    ladrillos[c][r].x = posXLadrillo;
115.                    ladrillos[c][r].y = posYLadrillo;
116.                    ctx.beginPath();
117.                    ctx.rect(posXLadrillo, posYLadrillo,
    anchuraLadrillos, alturaLadrillos);
118.                    ctx.fillStyle = "#FF01BA";
119.                    ctx.fill();
120.                    ctx.closePath();
121.                }
122.            }
123.        }
124.    }
125.
126.    function dibujarPuntaje() {
127.        ctx.font = "15px timenewroman";
128.        ctx.fillStyle = "#FF01BA";
129.        ctx.fillText("puntaje: "+puntaje, 8, 20);
130.    }
131.
132.    function dibujar() {

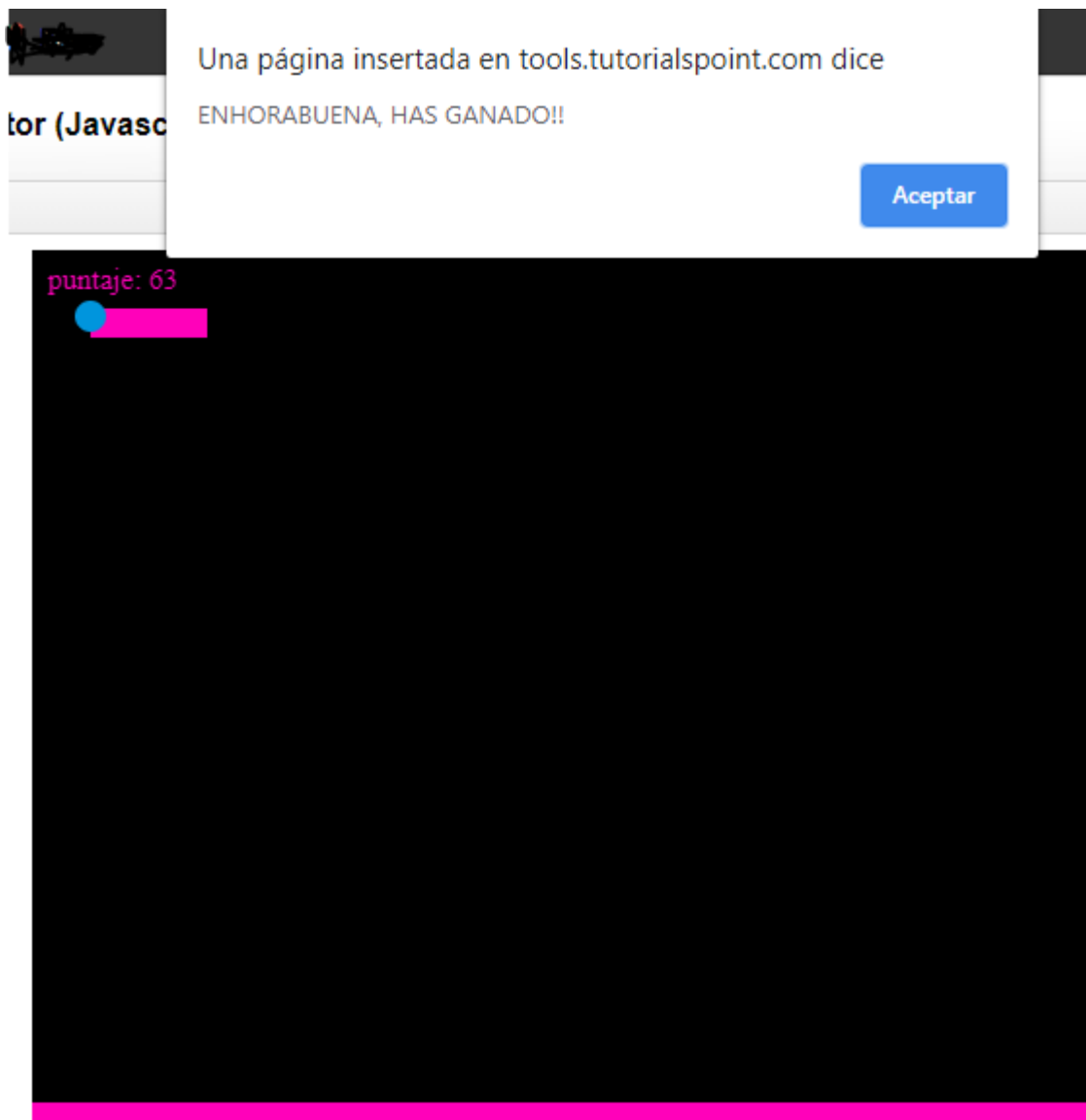
```

```

133.         ctx.clearRect(0, 0, canvas.width, canvas.height)
134.         ;
135.         dibujarLadrillos();
136.         dibujarBola();
137.         dibujarPaleta();
138.         dibujarPuntaje();
139.         detectarColision();
140.         if(x + dx > canvas.width-
radioBola || x + dx < radioBola) {
141.             dx = -dx;
142.         }
143.         if(y + dy < radioBola) {
144.             dy = -dy;
145.         }
146.         else if(y + dy > canvas.height-radioBola) {
147.             if(x > paletaPosX && x < paletaPosX + anchur
aPaleta) {
148.                 dy = -dy;
149.             }
150.             else {
151.                 clearInterval(juego);
152.                 alert("OH!! HAS PERDIDO, VUELVE A
INTENTARLO");
153.                 document.location.reload();
154.             }
155.         }
156.         if(flechaDerechaPresionada && paletaPosX < canva
s.width-anchuraPaleta) {
157.             paletaPosX += 7;
158.         }
159.         else if(flechaIzquierdaPresionada && paletaPosX
> 0) {
160.             paletaPosX -= 7;
161.         }
162.         x += dx;
163.         y += dy;
164.     }
165.     var juego = setInterval(dibujar, 8);
166. </script>
167. </body>
168. </html>

```

Al ejecutarse se vería así:





## 10 FASE 8: CONTROLANDO EL RATÓN

---

En esta fase se implementará que la paleta no solo se mueva por medio del teclado, sino que también por medio del ratón, y para conseguir esto se introduce al código una función, una variable y una condición que, en el caso de ser cumplida, se podrá mover la paleta con el ratón.

A continuación, en código:

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4.     <meta charset="utf-8" />
5.     <title>Juego 2D - fase 8</title>
6.     <style>* { padding: 0; margin: 0; } canvas { background:
7.         #000000; display: block; margin: 0 auto; }</style>
8. </head>
9. <body>
10.     <canvas id="miCanvas" width="600" height="450"></canvas>
11.
12.     <script>
13.         var canvas = document.getElementById("miCanvas");
14.         var ctx = canvas.getContext("2d");
15.
16.         var radioBola = 8;
17.         var x = canvas.width/2;
18.         var y = canvas.height-30;
19.         var dx = 2;
20.         var dy = -2;
21.
22.         var alturaPaleta = 12;
23.         var anchuraPaleta = 60;
24.         var paletaPosX = (canvas.width-anchuraPaleta)/2;
25.
26.         var flechaDerechaPresionada = false;
27.         var flechaIzquierdaPresionada = false;
28.
29.         var nroFilasLadrillos = 8;
30.         var nroColumnasLadrillos = 8;
31.         var anchuraLadrillo = 75;
```

```

32.     var alturaLadrillo = 20;
33.     var rellenoLadrillo = 10;
34.     var vacioSuperiorLadrillo = 30;
35.     var vacioIzquierdoLadrillo = 30;
36.
37.     var puntaje = 0;
38.
39.     var ladrillos = [];
40.     for(var c=0; c<nroColumnasLadrillos; c++) {
41.         ladrillos[c] = [];
42.         for(var f=0; f<nroFilasLadrillos; f++) {
43.             ladrillos[c][f] = { x: 0, y: 0, estado: 1 };
44.         }
45.     }
46.
47.     document.addEventListener("keydown", manejadorTeclaP
resionada, false);
48.     document.addEventListener("keyup", manejadorTeclaLib
erada, false);
49.
50.
51.     document.addEventListener("mousemove", manejadorRato
n, false);
52.
53.     function manejadorTeclaPresionada(e) {
54.         if(e.keyCode == 39) {
55.             flechaDerechaPresionada = true;
56.         }
57.         else if(e.keyCode == 37) {
58.             flechaIzquierdaPresionada = true;
59.         }
60.     }
61.
62.     function manejadorTeclaLiberada(e) {
63.         if(e.keyCode == 39) {
64.             flechaDerechaPresionada = false;
65.         }
66.         else if(e.keyCode == 37) {
67.             flechaIzquierdaPresionada = false;
68.         }
69.     }
70.
71.
72.     function manejadorRaton(e) {
73.         var posXRatonDentroDeCanvas = e.clientX - canvas
.offsetLeft;
74.
75.         if(posXRatonDentroDeCanvas > 0 && posXRatonDentr
oDeCanvas < canvas.width) {

```

```
76.
77.             paletaPosX = posXRatonDentroDeCanvas - anchu
raPaleta/2;
78.         }
79.     }
80.     function detectarColision() {
81.         for(var c=0; c<nroColumnasLadrillos; c++) {
82.             for(var r=0; r<nroFilasLadrillos; r++) {
83.                 var b = ladrillos[c][r];
84.                 if(b.estado == 1) {
85.                     if(x > b.x && x < b.x+anchuraLadrill
o && y > b.y && y < b.y+alturaLadrillo) {
86.                         dy = -dy;
87.                         b.estado = 0;
88.                         puntaje++;
89.                         if(puntaje == nroFilasLadrillos*
nroColumnasLadrillos) {
90.                             alert("ENHORABUENA, HAS
GANADO!!");
91.                             document.location.reload();
92.                         }
93.                     }
94.                 }
95.             }
96.         }
97.     }
98.
99.     function dibujarBola() {
100.         ctx.beginPath();
101.         ctx.arc(x, y, radioBola, 0, Math.PI*2);
102.         ctx.fillStyle = "#F72006";
103.         ctx.fill();
104.         ctx.closePath();
105.     }
106.     function dibujarPaleta() {
107.         ctx.beginPath();
108.         ctx.rect(paletaPosX, canvas.height-
alturaPaleta, anchuraPaleta, alturaPaleta);
109.         ctx.fillStyle = "#EB0ABF";
110.         ctx.fill();
111.         ctx.closePath();
112.     }
113.     function dibujarLadrillos() {
114.         for(var c=0; c<nroColumnasLadrillos; c++) {
115.             for(var r=0; r<nroFilasLadrillos; r++) {
116.                 if(ladrillos[c][r].estado == 1) {
117.                     var brickX = (r*(anchuraLadrillo+rel
lenoLadrillo))+vacioIzquierdoLadrillo;
```

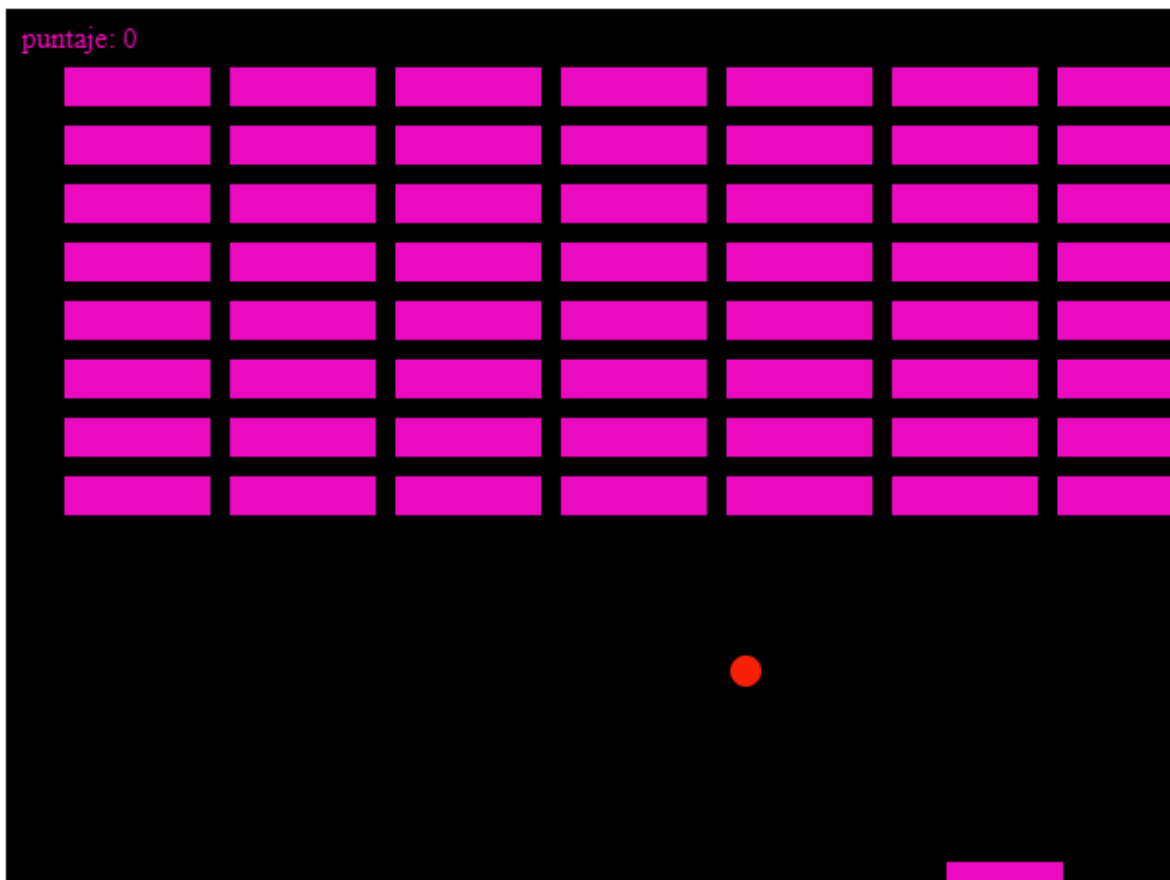
```

118.                var brickY = (c*(alturaLadrillo+rell
    enoLadrillo))+vacioSuperiorLadrillo;
119.                ladrillos[c][r].x = brickX;
120.                ladrillos[c][r].y = brickY;
121.                ctx.beginPath();
122.                ctx.rect(brickX, brickY, anchuraLadr
    illo, alturaLadrillo);
123.                ctx.fillStyle = "#EB0ABF";
124.                ctx.fill();
125.                ctx.closePath();
126.            }
127.        }
128.    }
129.
130.    function dibujarPuntaje() {
131.        ctx.font = "15px timesnewroman";
132.        ctx.fillStyle = "#F80CBF";
133.        ctx.fillText("puntaje: "+puntaje, 8, 20);
134.    }
135.
136.    function dibujar() {
137.        ctx.clearRect(0, 0, canvas.width, canvas.height)
    ;
138.        dibujarLadrillos();
139.        dibujarBola();
140.        dibujarPaleta();
141.        dibujarPuntaje();
142.        detectarColision();
143.
144.        if(x + dx > canvas.width-
    radioBola || x + dx < radioBola) {
145.            dx = -dx;
146.        }
147.        if(y + dy < radioBola) {
148.            dy = -dy;
149.        }
150.        else if(y + dy > canvas.height-radioBola) {
151.            if(x > paletaPosX && x < paletaPosX + anchur
    aPaleta) {
152.                dy = -dy;
153.            }
154.            else {
155.                clearInterval(juego);
156.                alert("OH, HAS PERDIDO");
157.                document.location.reload();
158.            }
159.        }
160.

```

```
161.         if(flechaDerechaPresionada && paletaPosX < canva
    s.width-anchuraPaleta) {
162.             paletaPosX += 7;
163.         }
164.         else if(flechaIzquierdaPresionada && paletaPosX
    > 0) {
165.             paletaPosX -= 7;
166.         }
167.
168.         x += dx;
169.         y += dy;
170.     }
171.
172.     var juego = setInterval(dibujar, 8);
173. </script>
174.
175. </body>
176. </html>
```

Y así se visualiza:



## 11 FASE 9: FINALIZANDO EL JUEGO

---

A esta altura del código ya podríamos decir que el juego estaría terminado, no obstante, sabemos que se le pueden agregar un sinfín de cosas. Ya para acabar se le agrega lo que sería la función de las vidas, una condición que regule cuando las vidas lleguen a cero el juego se reinicia, esto irá de la mano con un ciclo que también se pondrá, restaurando por completo el juego; y la otra parte sería la de ocultar el cursor mientras se está jugando, esto se logra con una variable que modifique el estilo del cursor eliminándolo.

La interfaz del juego sería:



## 12 CONCLUSIONES

---

Podemos ver que este juego esta hecho completamente por un código html citado a java script, y si, se puede llegar a ver algo tedioso ver toda esa fila de código, pero se demostró que siguiendo todas las pautas al pie de la letra se puede llegar a desarrollar el juego sin problemas.

Esto sirve básicamente como aprendizaje, porque puedes modificar tanto los datos básicos del juego como ir más allá añadiendo más cosas que hacen del juego mas completo.



## 13 BIBLIOGRAFÍA

---

[https://developer.mozilla.org/es/docs/Games/Workflows/Famoso\\_juego\\_2D\\_usando\\_JavaScript\\_puro/Construye\\_grupo\\_bloques](https://developer.mozilla.org/es/docs/Games/Workflows/Famoso_juego_2D_usando_JavaScript_puro/Construye_grupo_bloques)