



PAC de Desarrollo



Pruebas

La aplicación consiste en simular el juego de una máquina tragaperras, donde el usuario introduce en un input el número de monedas con las que quiere jugar, pulsa una palanca para iniciar la tirada, recibe una serie de premios según las combinaciones obtenidas aleatoriamente, y recupera las monedas al finalizar el juego.

Para el desarrollo de la aplicación he utilizado el IDE **VSC**, con las extensiones **Prettier** (ayuda con el formateo de código), y **ESLint** (ayuda a detectar errores), los apuntes y ejemplos de las clases, consultado la web <https://www.w3schools.com>, y también he utilizado bastante, desde el menú contextual del navegador chrome donde testeaba archivo index.html, la opción Inspeccionar - Console, para ir detectando errores durante el desarrollo.

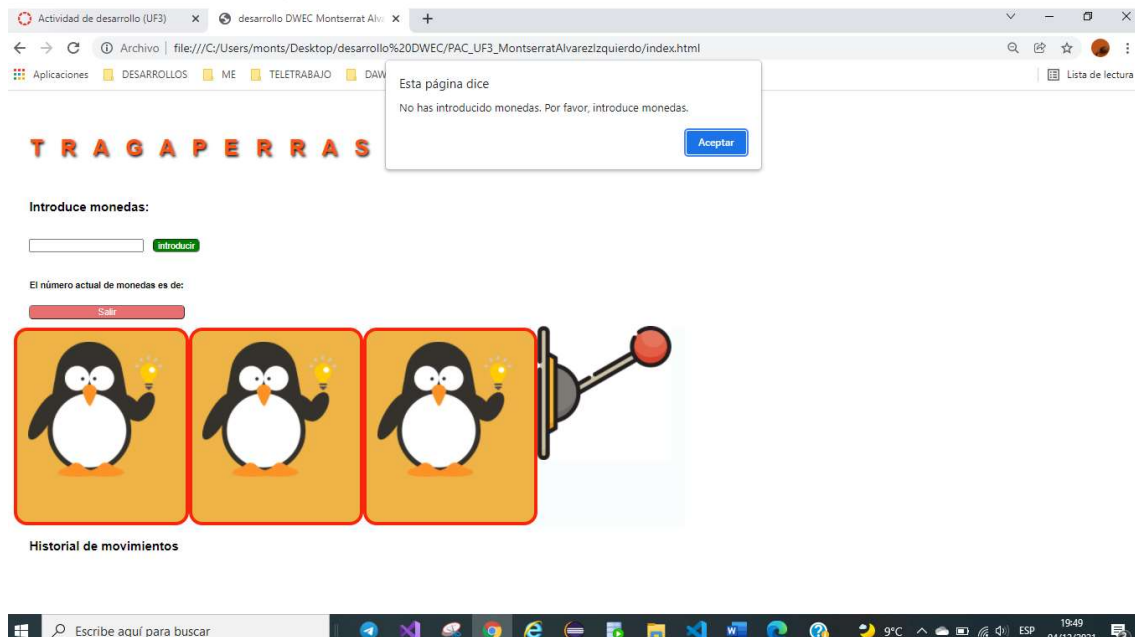
En la carpeta del proyecto he incluido una carpeta con las imágenes facilitadas por el enunciado, este pdf con carátula, y he creado tres archivos:

- **index.html**, con la estructura de la aplicación,
- **styles.css** con los estilos de la aplicación (aunque en este caso no he utilizado Bootstrap porque en la vt indicaste que no lo utilizásemos),
- y **javascript.js**, con la lógica de la aplicación.

1. Intentar una tirada sin introducir monedas en la tragaperras.

- Si el usuario sin introducir monedas en el input hace clic en el botón "introducir", lo primero que va a ocurrir es que le aparezca una alerta con el mensaje **"No has introducido monedas. Por favor, introduce monedas."**, dando aviso para que el usuario introduzca monedas.

navegador:

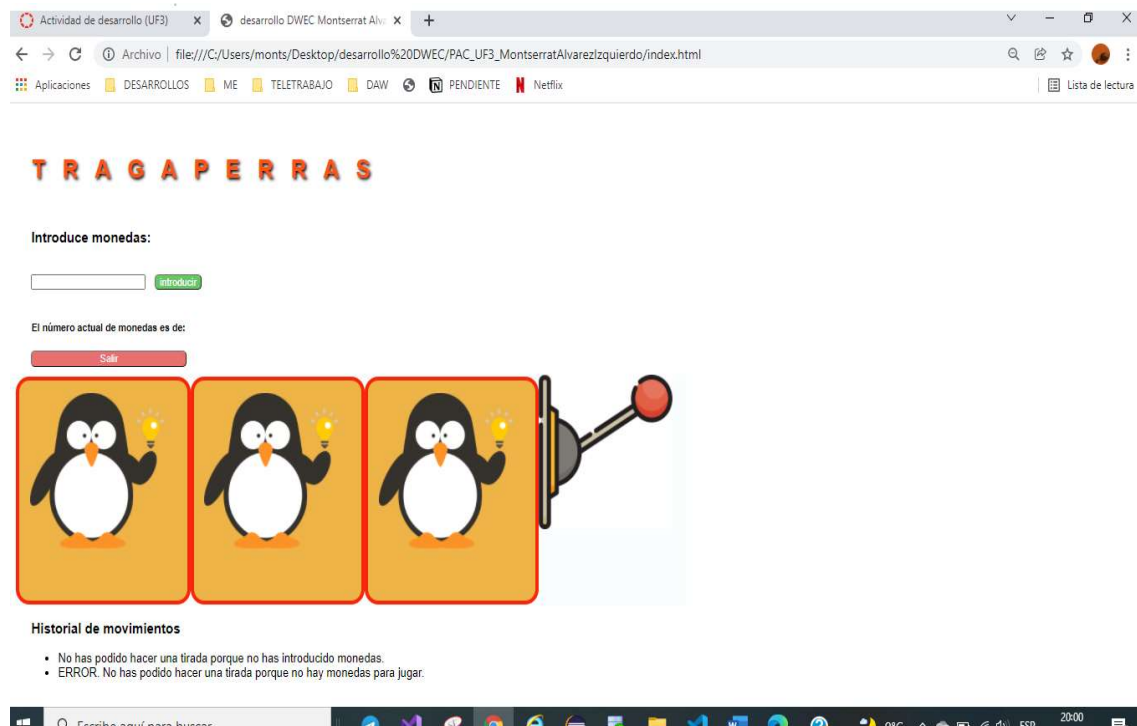


- Si a pesar de esta alerta y haciendo caso omiso, el usuario intenta una tirada sin introducir monedas en el input, al hacer clic en el ratón sobre el elemento palanca, se ejecuta la función **pullHandle()**, que cambia el src de la imagen del elemento para que su imagen sea visualmente la palanca abajo.
- Al soltar el ratón sobre el elemento palanca, se ejecuta la función **pushHandle()**, que aparte de volver a cambiar el src de la imagen del elemento para que su imagen sea visualmente la palanca arriba, hace una serie de comprobaciones que detallaré en los siguientes puntos del pdf, pero antes revisa si la variable está vacía y si lo está, llamará a la función **addHistory()** pasándole por parámetro el mensaje **"ERROR. No has podido hacer una tirada porque no hay monedas para jugar."**, de forma que en el historial aparezca éste mensaje. Inicialmente había creado también una alerta pero la he eliminado porque me resulta menos invasivo y más amigable que aparezca en el historial, ya que las alertas son bastante pesadas, y ya hay otras alertas en el desarrollo.

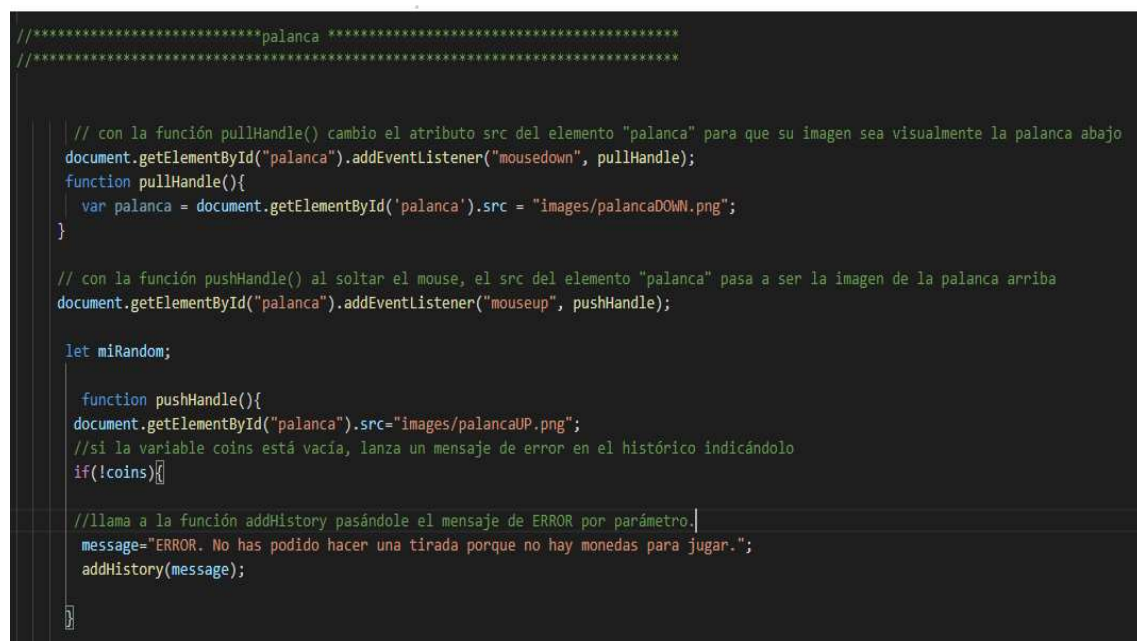
JavaScript.js:

```
//*****botón submitCoins*****  
//*****  
  
document.getElementById("submitCoins").addEventListener("click", submitCoins);  
// función que ejecuta tras hacer clic en el boton "introducir"  
function submitCoins() {  
  
    //sustituye el valor que había al cargar la pág. en el h2, por el número de monedas que el usuario ha introducido en el input  
    coins = document.getElementById("coins").innerHTML = document.getElementById("inputCoins").value;  
  
    //si coins es 0, hago saltar una alerta con un mensaje, y también llamo a la función addHistory pasándole otro mensaje por parámetro  
    if(coins==0){  
        alert("No has introducido monedas. Por favor, introduce monedas.");  
        message="No has podido hacer una tirada porque no has introducido monedas.";  
        addHistory(message);  
    }  
}
```

navegador:



JavaScript.js:



- La función addHistory() crea una etiqueta en su nodo padre a la que le añadirá un texto, que le paso por parámetro

```

/***** HISTORIAL *****/
// La función addHistory creará una etiqueta <li></li> en su nodo padre <ul></ul> a la que le añadirá un texto, que le pasaremos por parámetro

function addHistory(message){
    var li = document.createElement("li");
    var msg = document.createTextNode(message);
    li.appendChild(msg);
    historyUl.appendChild(li);
}

/*****

```

2. Introducir 10 monedas.

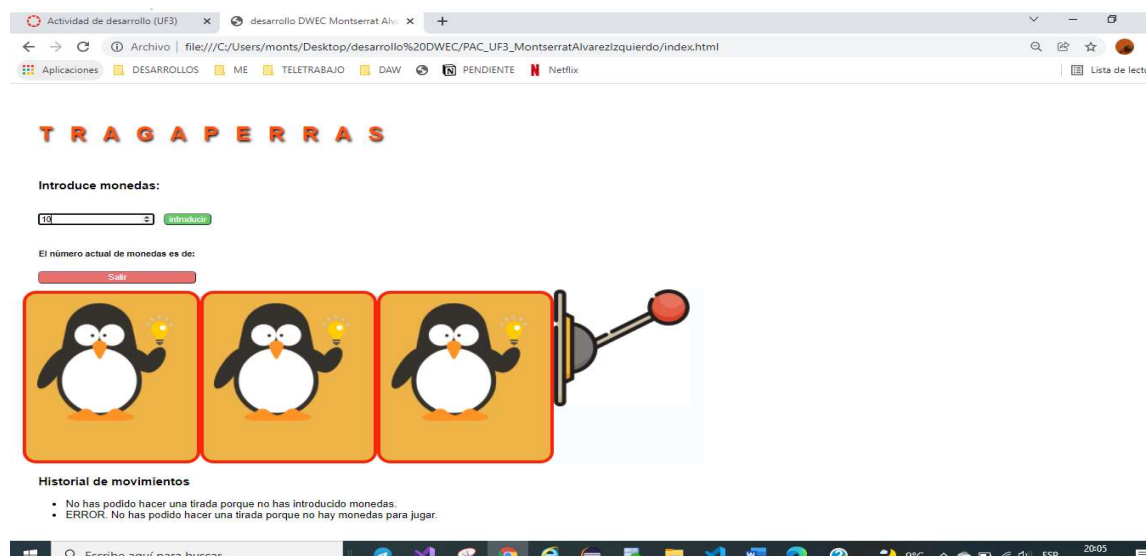
- El usuario sólo podrá introducir valores positivos dado que he añadido el atributo `min="0"` en la etiqueta del input para que no permita valores negativos.

index.html :

[illegible]

- El usuario introduce el valor "10" en el input de tipo number.

navegador:



- Tras introducir 10 monedas, y pulsar el botón **submitCoins**, sustituyo el valor inicial de coins que había al cargar la pág. en la etiqueta h2, por el número de monedas que el usuario ha introducido en el input.

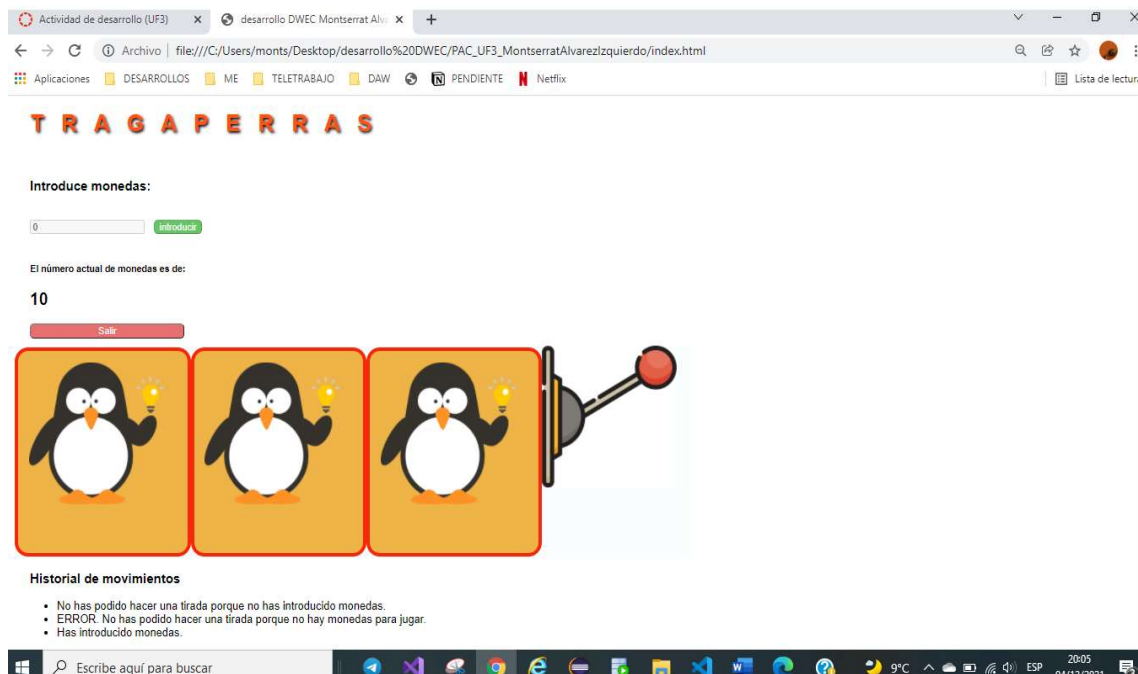
JavaScript.js:

```
//*****botón submitCoins*****
//*****

document.getElementById("submitCoins").addEventListener("click", submitCoins);
// función que ejecuta tras hacer clic en el boton "introducir"
function submitCoins() {

//sustituye el valor que había al cargar la pág. en el h2, por el número de monedas que el usuario ha introducido en el input
coins = document.getElementById("coins").innerHTML = document.getElementById('inputCoins').value;
```

navegador:



- llamo a la función **addHistory()** y le paso por parámetro el mensaje "Has introducido monedas."
-

JavaScript.js:

```
} else {
//llamo a la función addHistory y le paso por parámetro un mensaje
message="Has introducido monedas.";
addHistory(message);
```

- paso a 0 el valor del input

JavaScript.js:

```
//paso a 0 el valor del input
document.getElementById("inputCoins").value = "0";
```

- y bloqueó la posibilidad de introducir más monedas, desactivando tanto el input como el botón "introducir",

JavaScript.js:

```
// se bloqueará la posibilidad de introducir más monedas, desactivando tanto el input como el botón "introducir"
document.getElementById("inputCoins").disabled = true;
document.getElementById("submitCoins").disabled = true;
```

3. Realizar 5 tiradas.

TIRADA1

- Al hacer clic al ratón sobre la palanca y teniendo monedas en el input, se ejecuta la función **pullHandle()** en la que se cambia el atributo src del elemento "palanca" ahora al src="images/palancaDOWN.png" para que su imagen sea visualmente la palanca abajo.

JavaScript.js:

```
//*****palanca *****
//*****
// con la función pullHandle() cambio el atributo src del elemento "palanca" para que su imagen sea visualmente la palanca abajo
document.getElementById("palanca").addEventListener("mousedown", pullHandle);
function pullHandle(){
    var palanca = document.getElementById('palanca').src = "images/palancaDOWN.png";
}
```

- Al soltar el botón del ratón sobre el elemento palanca, se ejecuta la función **pushHandle()** que hace varias cosas, que detallo:
 - El src del elemento "palanca" pasa a ser la imagen de la palanca arriba src="images/palancaUP.png"

JavaScript.js:

```
// con la función pushHandle() al soltar el mouse, el src del elemento "palanca" pasa a ser la imagen de la palanca arriba
document.getElementById("palanca").addEventListener("mouseup", pushHandle);
```

- Resta una moneda a coins

JavaScript.js:

```
//resta una moneda (tirada)
coins--;
```

- utilizo `Math.floor(Math.random())` para devolver entero aleatorio entre 1 y la longitud del array facilitado para el ejercicio, y almaceno en tres variables los tres enteros aleatorios que utilizaré para que, dentro del array facilitado por el ejercicio, dé la posición del elemento que aparecerá en el contenedor tras ejecutarse la función.
- Luego creo tres variables que almacenarán el resultado de añadirle (concatenar) la ruta y la extensión al elemento del array obtenido aleatoriamente, para que se creen los 3 srcs a las imágenes a mostrar al usuario en los tres contenedores.
- Pinta el nuevo valor de coins.
- Llama a la función **addHistory()** para añadir el mensaje "Has gastado una moneda." al historial.

JavaScript.js:

```
//resta una moneda (tirada)
coins--;

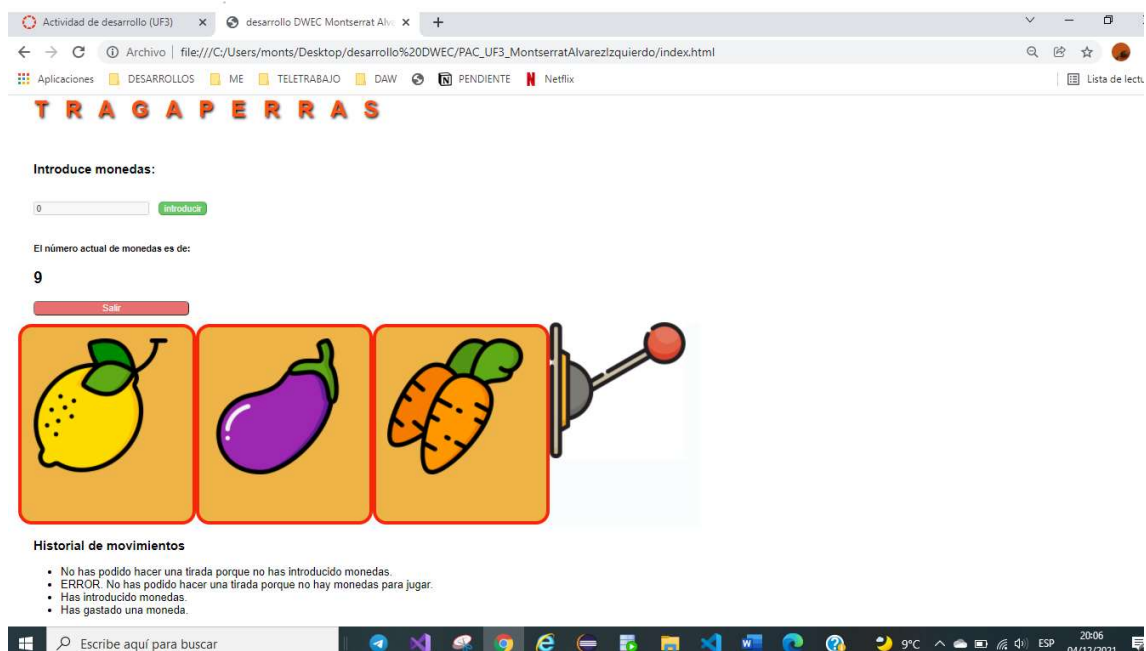
//utilizo Math.floor(Math.random() para devolver entero aleatorio entre 1 y la longitud del array srcImgBox https://www.w3schools.com/
miRandom1 = Math.floor(Math.random() * listaImagenes.length);
miRandom2 = Math.floor(Math.random() * listaImagenes.length);
miRandom3 = Math.floor(Math.random() * listaImagenes.length);

let result1 = document.getElementById('box1').src = ("images/" + listaImagenes[miRandom1] + ".png");
let result2 = document.getElementById('box2').src = ("images/" + listaImagenes[miRandom2] + ".png");
let result3 = document.getElementById('box3').src = ("images/" + listaImagenes[miRandom3] + ".png");

//pinta el valor de coins y llama a la función addHistory para añadir el mensaje conforme se ha gastado una moneda (tirada)
document.getElementById("coins").innerHTML = coins;
message="Has gastado una moneda.";
addHistory(message);
```

- Y finalmente, para indicar los premios en función de las imágenes que se obtengan aleatoriamente, se comparan las tres imágenes que, en esta primera tirada, no procede mencionar, dado que no ha salido ningún premio.

navegador:



TIRADA 2

En este caso al aparecer un dollar, hemos obtenido un premio de UNA moneda.

JavaScript.js:

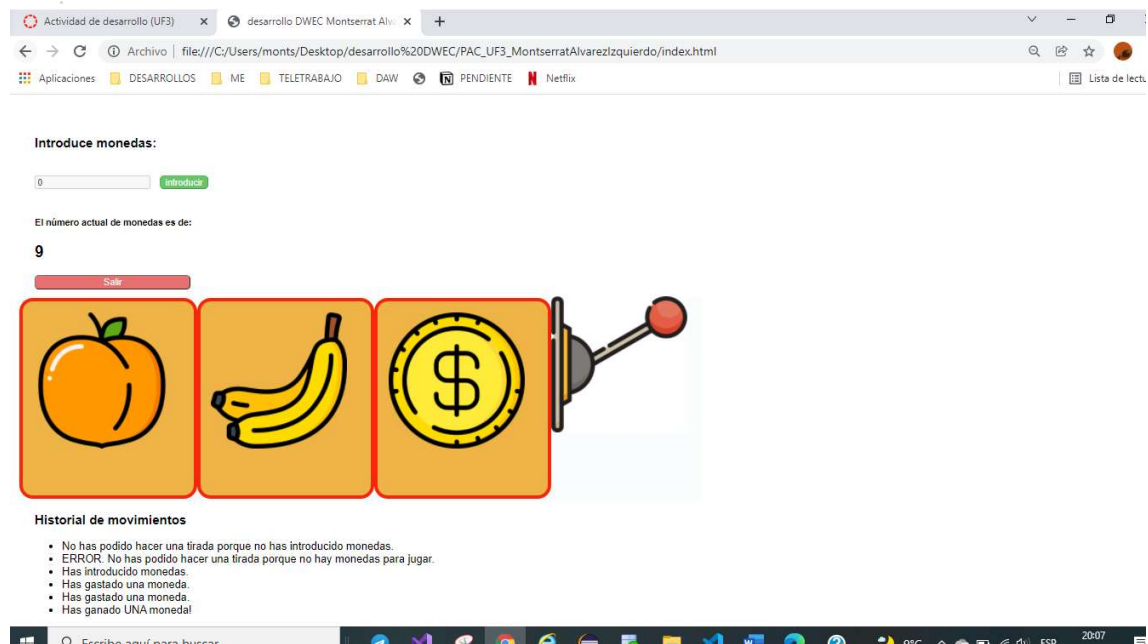
```
// *****COMPARAMOS LOS 3 RESULTADOS PARA OTORGAR LOS PREMIOS SI PROCEDE*****  
//*****
```

(...) previamente al código que muestro referente al resultado obtenido en esta tirada, he descartado que no son ni 3 dollar, ni 2 dollar, ni 3 vegetales iguales, ni 2 vegetales iguales y 1 dollar, y al descartarse estas comparaciones anteriores, por lo que sólo me planteo si resultado 1 ó 2 ó 3 son dollar. Si entra en el if else porque se cumple la condición como es el caso, pues suma a coins 1 moneda, pinta el nuevo valor de coins, y llama a la función **addHistory(message)** pasándole el mensaje "Has ganado UNA moneda!".

JavaScript.js:

```
/****** SI NO SON NI 3 DOLLAR, NI 2 DOLLAR, NI 3 VEGETALES, NI 2 VEGETALES Y 1 DOLLAR, ENTRARÁ AQUÍ *****  
si sale 1 dollar ***** "Has ganado UNA moneda!" ***** coins++;  
Planteo si resultado 1 ó 2 ó 3 son dollar ya que no puede ser ni 3 dollar, ni 2 dollar, ni 1 dollar y 2 vegetales iguales*/  
else if(result1 == "images/" + listaImagenes[4] + ".png" || result2 == "images/" + listaImagenes[4] + ".png" || result3 == "images/" +  
coins ++;  
document.getElementById("coins").innerHTML = coins;  
message="Has ganado UNA moneda!";  
addHistory(message);  
}
```

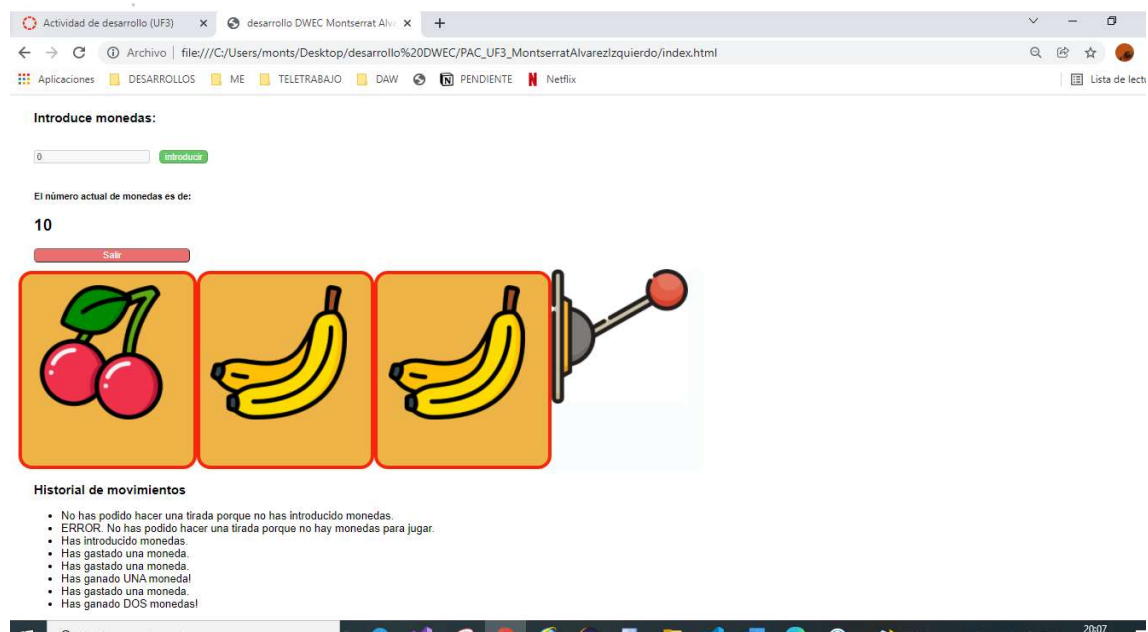

navegador:



TIRADA 3

En este caso al aparecer dos vegetales iguales, hemos obtenido un premio de DOS monedas (...) previamente al código que muestro referente al resultado obtenido en esta tirada, he descartado que no son ni 3 dollar, ni 2 dollar, ni 3 vegetales iguales, ni 2 vegetales iguales y 1 dollar, y al descartarse estas comparaciones anteriores sólo planteo en este if else, que sean 2 resultados iguales, porque seguro serán vegetales dado que ya está descartado que sean dos dollars y también descartado que sean 3 dollars o 3 vegetales. Si entra en el if else porque se cumple la condición como es el caso, pues suma a coins 2 monedas, pinta el nuevo valor de coins, y llama a la función **addHistory(message)** pasándole el mensaje "Has ganado DOS monedas!" para que se añada al historial.

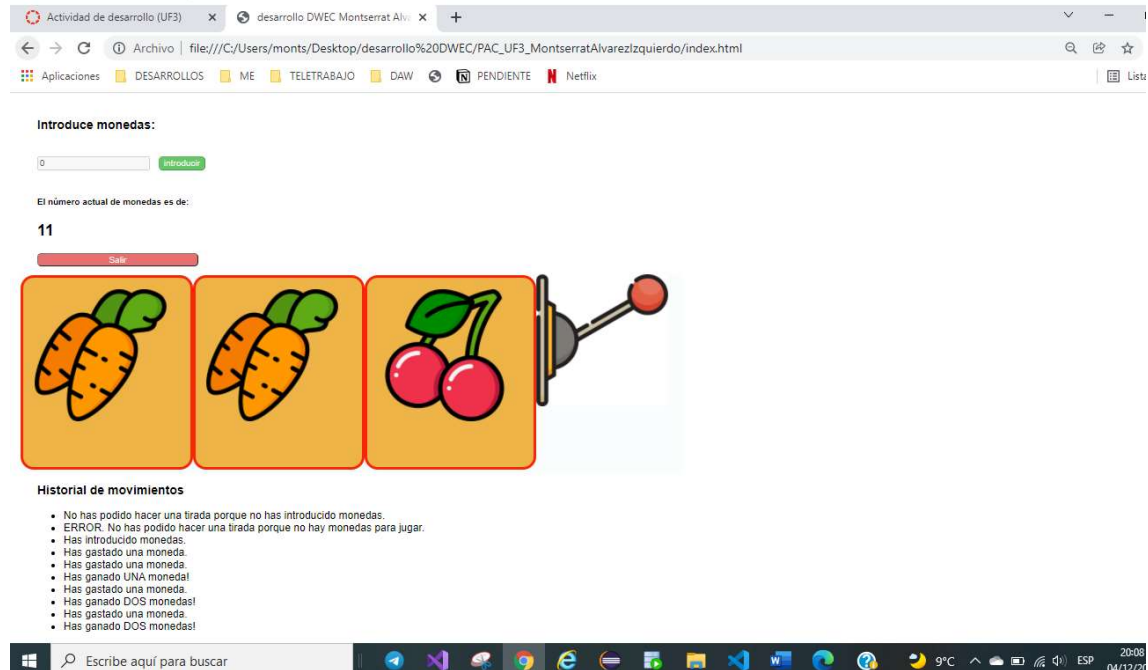
navegador:



TIRADA 4

En este caso al aparecer dos vegetales iguales, también hemos obtenido un premio de DOS monedas, como en la tirada anterior.

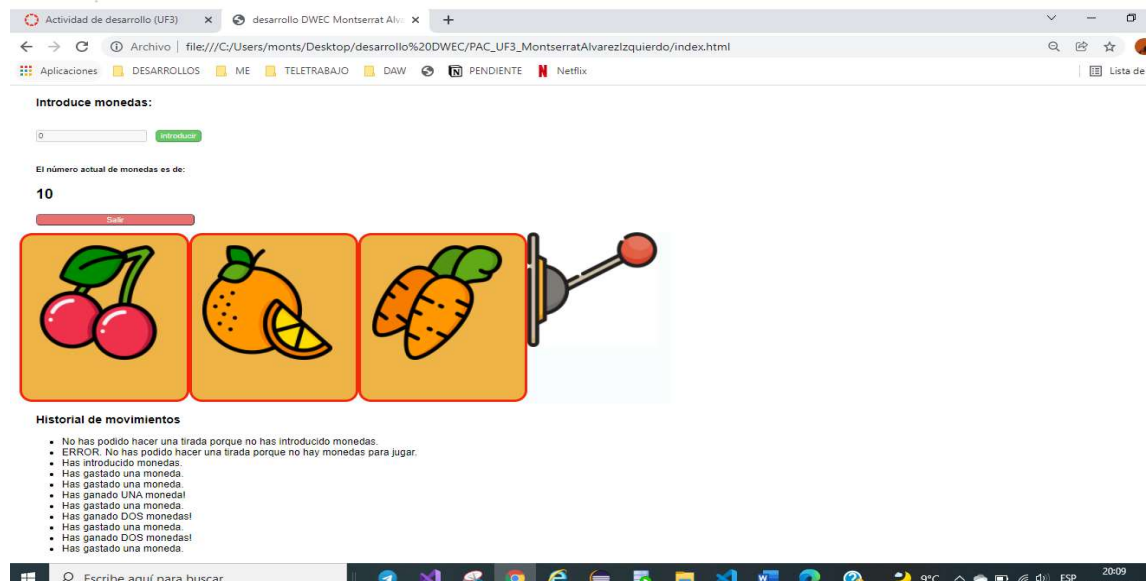
navegador:



TIRADA 5

En esta tirada no hay premio, por lo que resta la moneda por la tirada a coins sin sumarle nada porque no tiene ningún premio en este caso, pinta el nuevo valor de coins y llama a la función **addHistory()** para añadir el mensaje "Has gastado una moneda." al historial.

navegador:



4. Sacar monedas.

Al hacer clic en el botón "exit" se ejecuta la function exit() que hace varias cosas:

- lanza una alerta con un mensaje del total de monedas conseguidas "Has conseguido un total de " + coins + " monedas.", donde coins es el valor de coins en ese momento.
- vuelve a activar tanto el input como el botón "introducir".
- el valor de coins pasa al input.
- paso a 0 el valor del coins en el h2.
- y llamo a la función addHistory pasándole por parámetro el mensaje "Sacas todas las monedas."

index.html:

```
<!--botón "exit"-->  
<button type="button" id="exit">Salir</button><br>
```

JavaScript.js:

```
*****botón exit *****  
*****  
document.getElementById("exit").addEventListener("click", exit);  
// función que ejecuta tras hacer clic en el boton "salir"  
function exit() {  
  
    // lanza una alerta con un mensaje del total de monedas conseguidas  
    alert("Has conseguido un total de " + coins + " monedas.");  
  
    // vuelvo a activar tanto el input como el botón "introducir"  
    document.getElementById("submitCoins").disabled = false;  
    document.getElementById("inputCoins").disabled = false;  
  
    // el valor de coins pasa al input  
    document.getElementById("inputCoins").value = coins;  
  
    // paso a 0 el valor del coins en el h2  
    coins = document.getElementById("coins").innerHTML = "0";  
    // llamo a la función addHistory pasándole por parámetro el mensaje "Sacas todas las monedas."  
    message="Sacas todas las monedas.";  
    addHistory(message);  
}
```

navegador:

Introduce monedas:

0 Introducir

El número actual de monedas es de: 10

Esta página dice
Has conseguido un total de 10 monedas.

Aceptar

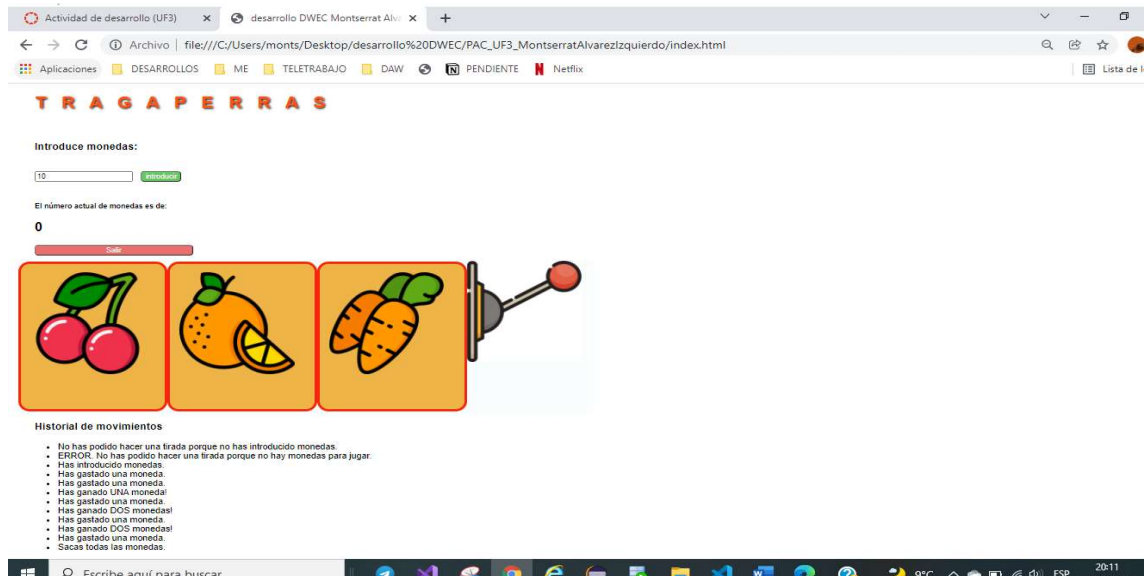
Historial de movimientos

- No has podido hacer una tirada porque no has introducido monedas.
- ERROR: No has podido hacer una tirada porque no hay monedas para jugar.
- Has introducido monedas.
- Has gastado una moneda.
- Has gastado una moneda.
- Has ganado UNA moneda!
- Has gastado una moneda.
- Has ganado DOS monedas!
- Has gastado una moneda.
- Has ganado DOS monedas!
- Has gastado una moneda.

5. Introducir 4 monedas adicionales.

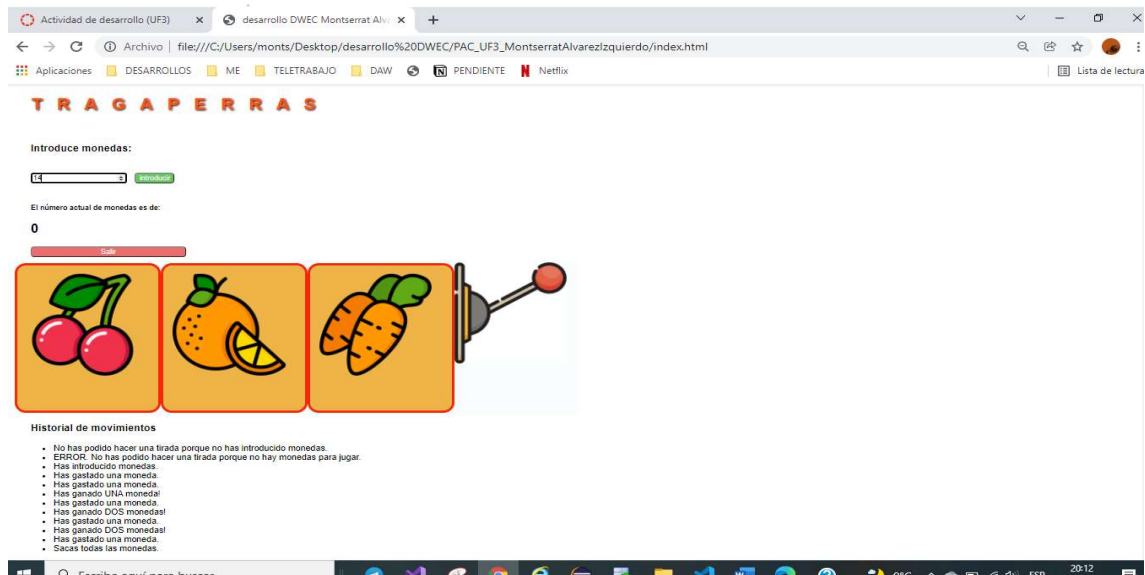
- Al pulsar aceptar en el mensaje de alerta conforme he sacado las monedas, éstas monedas vuelven al input.

navegador:



- Tras esto, añado 4 más, que en mi caso suman un total de 14 monedas.

navegador:



- Tras pulsar el botón **submitCoins**, sustituyo el valor de coins de la etiqueta h2, por el número de monedas que hemos introducido en el input,
- llamo a la función **addHistory()** y le paso por parámetro el mensaje "Has introducido monedas."

JavaScript.js:

```

    } else {
        //llamo a la función addHistory y le paso por parámetro un mensaje
        message="Has introducido monedas.";
        addHistory(message);
    }
}

```

- paso a 0 el valor del input,

JavaScript.js:

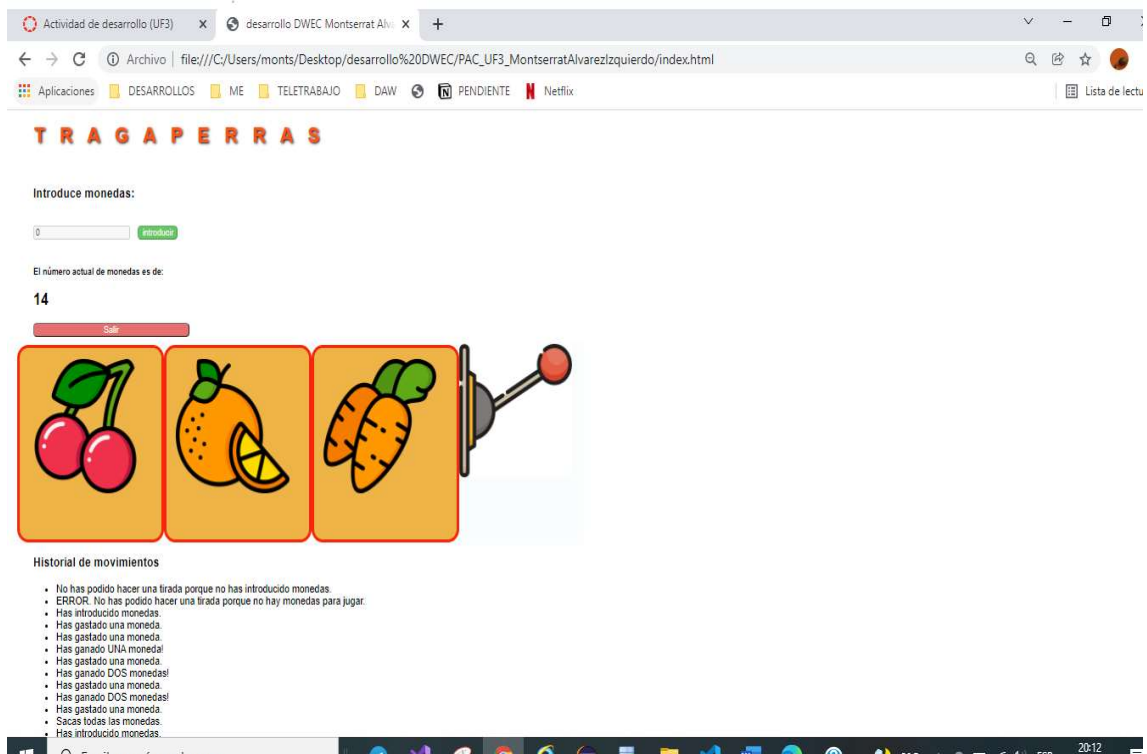
```
//paso a 0 el valor del input
document.getElementById("inputCoins").value = "0";
```

- y bloquéo la posibilidad de introducir más monedas, desactivando tanto el input como el botón "introducir".

JavaScript.js:

```
// se bloqueará la posibilidad de introducir más monedas, desactivando tanto el input como el botón "introducir"
document.getElementById("inputCoins").disabled = true;
document.getElementById("submitCoins").disabled = true;
```

navegador:

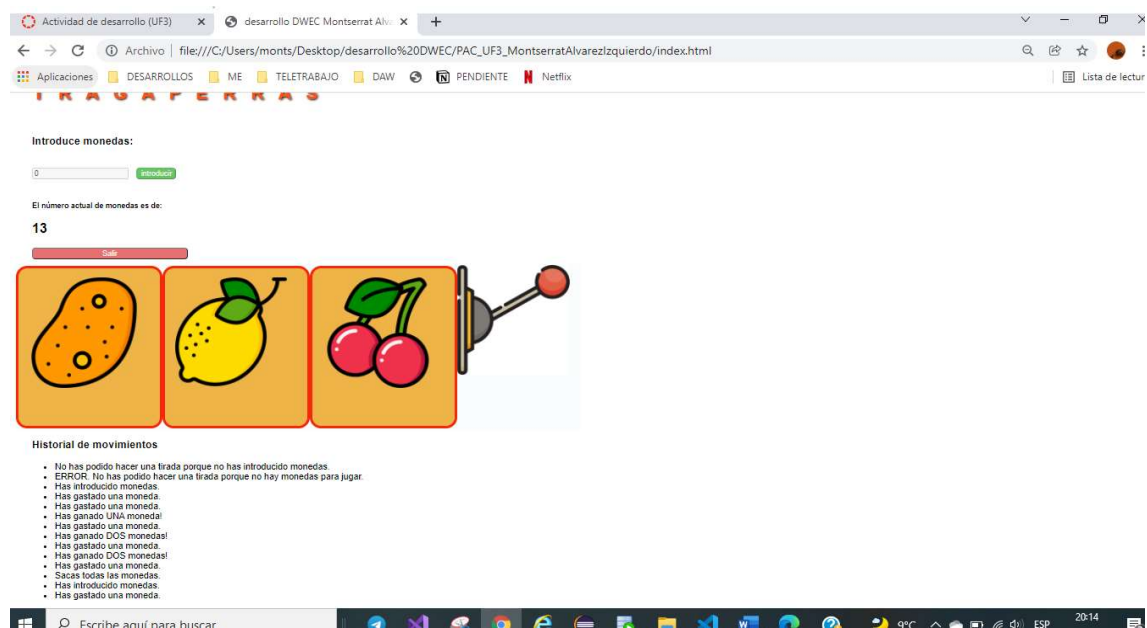


6. Realizar 5 tiradas adicionales.

TIRADA ADICIONAL 1

Sin premio, por lo que sólo se restará uno a coins y se mostrará en historial este gasto.

navegador:



TIRADA ADICIONAL 2

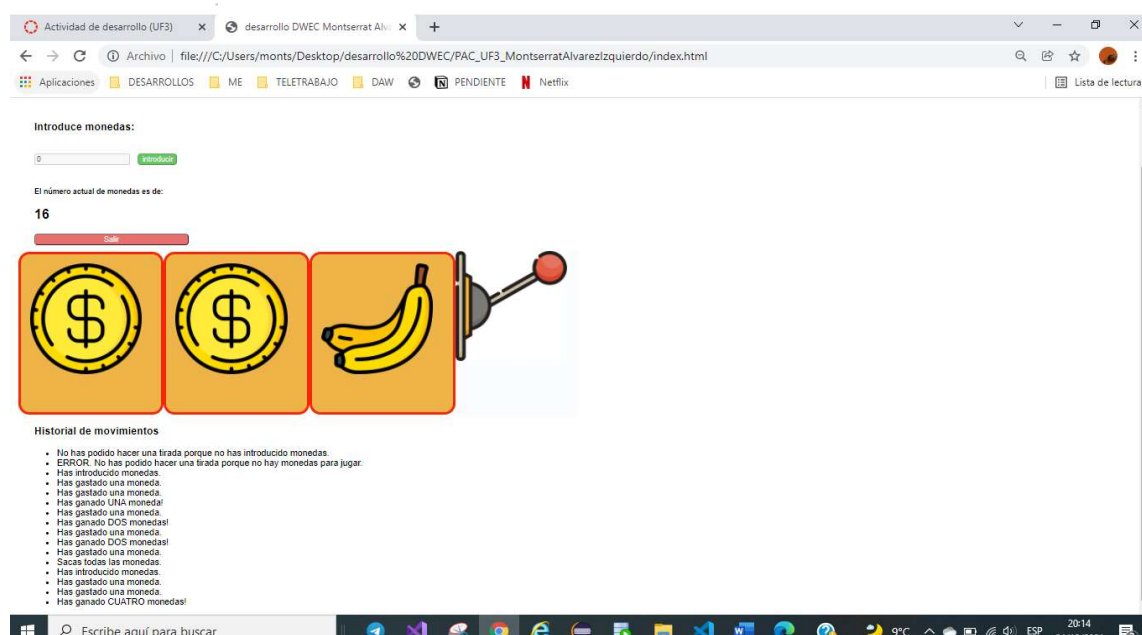
En este caso al salir dos monedas de dollar, sí que obtenemos un premio de CUATRO monedas.

(...) previamente al código que muestro referente al resultado obtenido en esta tirada, he descartado que no son 3 dollar por lo que me planteo si 2 resultados (en las tres combinaciones) son iguales y además que uno de ellos sea un dollar. Si entra en el if else porque se cumple la condición como es el caso, pues suma a coins 4 monedas, pinta el nuevo valor de coins, y llama a la función **addHistory(message)** pasándole el mensaje "Has ganado CUATRO monedas!".

JavaScript.js:

```
/******SI NO SON 3 DOLLAR, ENTRARÁ AQUÍ*****  
si sale 2 dollar ***** "Has ganado CUATRO monedas!" ***** coins = coins + 4;  
Planteo si 2 resultados (en las tres combinaciones) son iguales y además que uno de ellos sea un dollar *****  
else if ((result1 == result2 && result1 == "images/" + listaImagenes[4] + ".png")  
|| (result2 == result3 && result2 == "images/" + listaImagenes[4] + ".png") ||  
(result1 == result3 && result1 == "images/" + listaImagenes[4] + ".png")){  
  coins = coins + 4;  
  document.getElementById("coins").innerHTML = coins;  
  message="Has ganado CUATRO monedas!";  
  addHistory(message);  
}
```

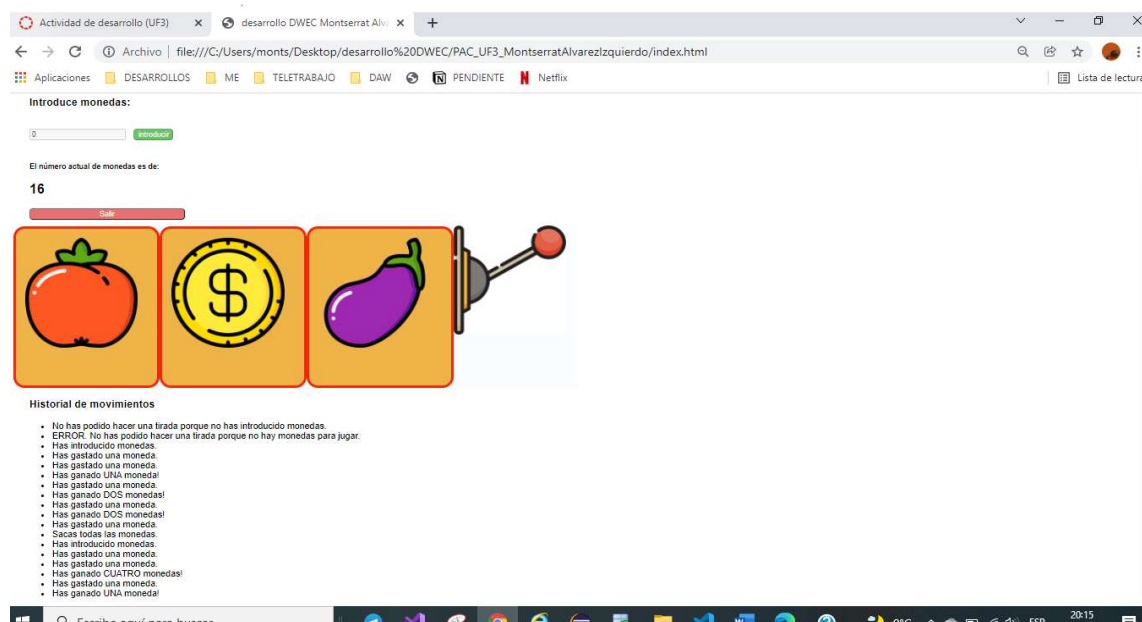

navegador:



TIRADA ADICIONAL 3

En este caso al aparecer un dollar, hemos obtenido un premio de UNA moneda. Ya explicado el código que interviene en la anterior **TIRADA 2** en la que nos salió el mismo premio.

navegador:



TIRADA ADICIONAL 4

Sin premio, por lo que sólo se restará uno a coins y se mostrará en historial este gasto.

navegador:

