

MEMORIN

Juego de memoria V2

Margalida Moyà Fiol

2022/2023

**Desarrollo Web en Entorno Cliente
CFGS Desarrollo de Aplicaciones Web**

Índice

Enlace al proyecto	3
Planificación del proyecto	4
Registro de errores	6

Enlace al proyecto

Enlace al GitHub:

<https://github.com/margaMF/DWC.git>

https://github.com/margaMF/DWC/tree/master/memorin_v2

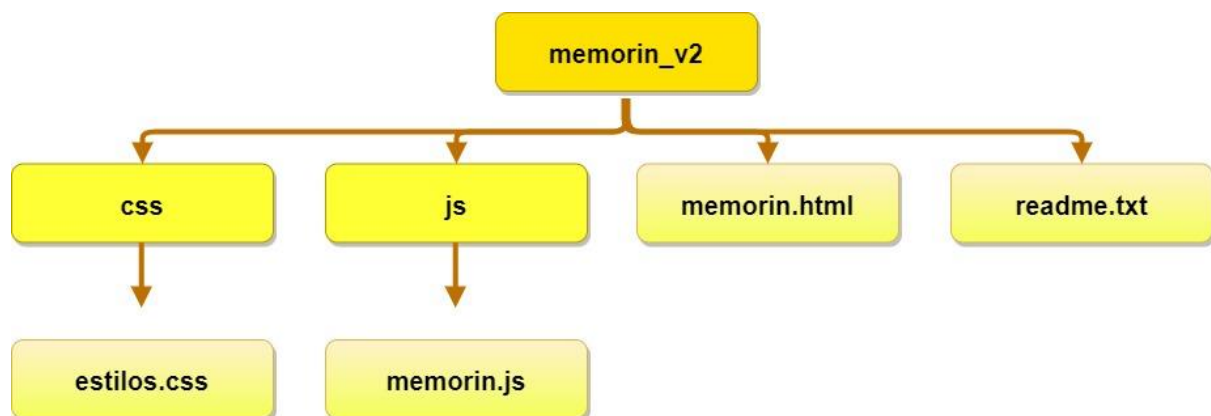
Planificación del proyecto

Inicio: 10/01/23

Fin previsto: 22/01/23

Fin real: 22/01/23

ESTRUCTURA DE LOS DIRECTORIOS:



Planificación del proyecto:

Inicio: 09/01/2023 **Final:** 10/01/2023

- Planificar todo lo que se debe hacer, el orden y los días en los que se va a realizar el proyecto.

Archivo .html:

- Estructura básica del archivo.
- Se mantiene igual a la versión 1.
- **Modificación1 Inicio: 20/01/2023 Final: 20/01/2023:** Se deberán añadir las etiquetas correspondientes para generar el botón de reiniciar el juego.
- **Modificación2 Inicio: /01/2023 Final: /01/2023** Se deberán añadir las etiquetas correspondientes para generar el panel de puntuación del juego.

Archivo .css:

- Aspecto visual que debe tener el proyecto.
- Se mantiene igual a la versión 1.
- **Modificación1 Inicio: 20/01/2023 Final: 20/01/2023:** Se deberán añadir los estilos correspondientes para el botón de reiniciar el juego.
- **Modificación2:** Se deberán añadir los estilos correspondientes para el panel de puntuación del juego.

Archivo .js:**- RQ6****Inicio: 11/01/2023 Final: 11/01/2023****Inicio Real: 11/01/2023 Final Real: 11/01/2023**

- Dibujar el tablero a través del DOM.
- Esconder el tablero al iniciar el juego.

- RQ7**Inicio: 11/01/2023 Final: 11/01/2023****Inicio Real: 12/01/2023 Final Real: 13/01/2023**

- Voltear las casillas con el click derecho del ratón.

- RQ8**Inicio: 12/01/2023 Final: 14/01/2023****Inicio Real: 13 /01/2023 Final Real: 22/01/2023**

- Ajustar las instrucciones que debe tener el juego:
 - Todas las cartas tendrán que estar boca abajo.
 - El jugador escoge dos cartas y las gira.
 - Si son iguales, las casillas se quedan destapadas y cambian a color verde.
 - Si no son iguales, las vuelve a girar y escoge otras dos recordando dónde están estas.
 - Sigue hasta encontrar a todas las parejas.

- RQ9**Inicio: 15/01/2023 Final: 17/01/2023****Inicio Real: 20/01/2023 Final Real: 22/01/2023**

- Botón para reiniciar el juego y generar un tablero nuevo.
- Avisar al usuario de que se reiniciará el tablero.
- Permitir que siga con el anterior si lo desea.

- RQ10**Inicio: 17/01/2023 Final: 19/01/2023****Inicio Real: /01/2023 Final Real: /01/2023**

- Incorporar un marcador de puntos que se actualice cada vez que se destape una pareja correctamente.
 - Debe estar inicializado en 0/0.
 - El primer núm son los puntos del usuario.
 - El segundo núm son los puntos máximos.

- RQ11

Inicio: 17/01/2023 Final: 17/01/2023

Inicio Real: /01/2023 Final Real: /01/2023

- Indicaciones de los puntos.
 - Si acierta al primer intento, 10 puntos.
 - Si acierta al segundo intento, 5 puntos.
 - Si acierta al tercer intento, 2.5 puntos.
 - Con más intentos no se suman puntos.

- RQ12

Inicio: 20/01/2023 Final: 22/01/2023

Inicio Real: 22/01/2023 Final Real: 22/01/2023

- Mensaje al finalizar el partido.
 - Enhorabuena.
 - Puntos obtenidos.
 - Tiempo.

Archivo registro de errores:

Inicio:11 /01/2023 Final: 22/01/2023

- Contendrá el registro de todos los errores que vayan surgiendo durante la realización del proyecto.

Archivo aclaraciones con el cliente:

Inicio:11 /01/2023 Final: 22/01/2023

- Contendrá todas las dudas y aclaraciones hechas al cliente necesarias para la realización del proyecto.

Archivo readme:

Inicio: 22/01/2023 Final: 22/01/2023

- Explicación del proyecto.
- Instrucciones del juego.
- Navegador que se ha utilizado.

Enlace al Drive:

https://docs.google.com/document/d/1I8VVP5Oz8dU2NZpzKkDoMsmQpDSZxBMQ_YrT2VR3_xk/edit

Registro de errores

Error nº1:

Fallo:11/01/23 Solución: 11/01/23

```

▶ Array(4)                                                                memorin.js:159
✖ Uncaught TypeError: memorin1.pintarTablero is not a function           memorin.js:160
  at memorin.js:160:10

```

No se imprimía el tablero porque no había creado el evento inicial.

Error nº2:

Fallo:11/01/23 Solución: 11/01/23

```

✖ ▶ Uncaught TypeError: Cannot read properties of undefined (reading 'document')
  at Memorin.pintarTableroDOM (memorin.js:54:25)                         memorin.js:54
  at window.onload (memorin.js:160:14)

```

El tablero seguía sin imprimirse porque había puesto:
columna.document.createElement('td'); en vez de
columna = document.createElement('td');

Error nº3:

Fallo:12/01/23 Solución: 13/01/23

```

▼ (4) [Array(6), Array(6), Array(6), Array(6)] ⓘ
  ▶ 0: (6) ['&#128125;', '&#128373;', '&#129313;', '&#128585;', '&#129302;', '&#129313;']
  ▶ 1: (6) ['&#128519;', '&#129313;', '&#128123;', '&#128126;', '&#128585;', '&#128373;']
  ▶ 2: (6) ['&#128520;', '&#128123;', '&#128126;', '&#129312;', '&#128125;', '&#128520;']
  ▶ 3: (6) ['&#129312;', '&#128520;', '&#129302;', '&#129313;', '&#128520;', '&#128519;']
    length: 4
    [[Prototype]]: Array(0)
✖ ▶ Uncaught TypeError: Cannot read properties of undefined (reading 'fila')
  at Memorin.despejar (memorin.js:155:43)                                memorin.js:155

```

No estaba detectando ni fila ni columna porque no estaba capturando el evento. Y porque estaba poniendo el mismo nombre a los dos métodos.

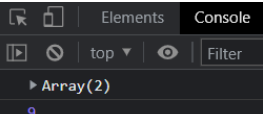
Error nº4:

Fallo:17/01/23 Solución: 17/01/23

```

arrayIconoCasilla = valorCasilla; arrayIconoCasilla = "&#129313;", valorCasilla = "&#129313;"
console.log(arrayIconoCasilla.length); arrayIconoCasilla = "&#129313;"

```



The screenshot shows the browser's developer console with the 'Console' tab selected. It displays the error message 'Uncaught TypeError: Cannot read properties of undefined (reading 'fila')' at line 155, column 43 of memorin.js. Below the error, the state of the arrayIconoCasilla variable is shown as an array of 4 elements, each being a string of 6 characters representing a 6x6 grid of memory icons.

Al guardar el valor de la casilla dentro del arrayIconoCasilla se guarda cada carácter como un elemento distinto del array.

Debía usar el `array.push` para guardar correctamente el valor de casilla y además al declarar el array dentro del evento cada vez que se ejecutaba el evento se sobrescribía el array, por lo que cada vez tenía un único elemento dentro del array.

Error nº5:

Fallo: 17/01/23 Solución: 18/01/23

```
179 esPareja(celda){ celda = td#f1_c0 {colSpan: 1, rowSpan: 1, headers: '', cellIndex: 0, align: '', ...}
180 //Comprobamos si el arrayIconoCasilla tiene dos posiciones y si estas dos son iguales.
181 //Comprobar si recibe el valor del icono y la posición de la celda.
182 if(this.arrayIconoCasilla.length == 2){
183     if(this.arrayIconoCasilla[0] == this.arrayIconoCasilla[1]){
184         celda.style.backgroundColor = 'green'; celda = td#f1_c0 {colSpan: 1, rowSpan: 1, headers: ''
185     }
186 }
```

El método no obtiene dos celdas distintas y no les pinta el fondo de color verde.

En el array guardaba el icono, pero al pasarle celda por parámetro a `esPareja()` solo le pasaba la última celda seleccionada por lo que al indicarle que cambiase el fondo de color solo cambiaba la última celda.

Al final dentro del array he guardado la celda entera con todos sus valores y cada vez que se hace click se guarda una celda nueva.

```
this.arrayIconoCasilla.push(celda);
}

esPareja(){
    //Comprobamos si el arrayIconoCasilla tiene dos posiciones y si estas dos son iguales.
    if(this.arrayIconoCasilla.length == 2){
        if(this.arrayIconoCasilla[0].innerHTML == this.arrayIconoCasilla[1].innerHTML){
            this.arrayIconoCasilla[0].style.backgroundColor = 'rgb(175, 255, 110)';
            this.arrayIconoCasilla[1].style.backgroundColor = 'rgb(175, 255, 110)';
        }
    }
}
```

Error nº6:

Fallo: 18/01/23 Solución: /01/23

```
// celda.removeEventListener('contextmenu', this.despejar); Lo miramos en clase pero no llegó ha funcionar.
You: 5 days ago • Error: undefined
```

```
//Esta opción tampoco funciona.
// this.arrayIconoCasilla[0].style.pointerEvents = 'none';
// this.arrayIconoCasilla[1].style.pointerEvents = 'none';
```

No he podido quitar ni bloquear el evento cuando las dos casillas son iguales.

Error nº7:

Fallo: 20/01/23 Solución: 20/01/23

```
esPareja(){
  //Comprobamos si el arrayIconoCasilla tiene dos posiciones y si estas dos son iguales.
  if(this.arrayIconoCasilla.length == 2){
    if(this.arrayIconoCasilla[0].innerHTML == this.arrayIconoCasilla[1].innerHTML){
      this.arrayIconoCasilla[0].style.backgroundColor = 'rgb(175, 255, 110)';
      this.arrayIconoCasilla[1].style.backgroundColor = 'rgb(175, 255, 110)';
    }
  }
```

Una vez encontrada la primera pareja correcta no se vuelve a cumplir la condición y en las demás parejas el color de fondo de la celda no cambia.

Se debía vaciar el array cada vez que se encontraba una pareja, para así poder cumplirse la condición según las posiciones.

```
esPareja(){
  //Comprobamos si el arrayIconoCasilla tiene dos posiciones y si estas dos son iguales.
  if(this.arrayIconoCasilla.length == 2){
    if(this.arrayIconoCasilla[0].innerHTML == this.arrayIconoCasilla[1].innerHTML){
      this.arrayIconoCasilla[0].style.backgroundColor = 'rgb(175, 255, 110)';
      this.arrayIconoCasilla[1].style.backgroundColor = 'rgb(175, 255, 110)';
      this.arrayIconoCasilla = [];
    }
  }
```

Enlace al Drive:

https://docs.google.com/document/d/1WBSA_fXRtOrXJXJ_1DqQYo4ctQLB3a-gUvgdQ_Duu5Q/edit