

Tarea 2

Requisitos: Para completar la tarea de forma satisfactoria se deben de cumplir una serie de requisitos previos:

- Cualquier tipo de **copia** hará que la calificación de la tarea sea de 0, si es de un compañero la calificación será 0 para ambos.
- La tarea se entregará a través del **Moodle** habilitado para la asignatura.
- Se deberá entregar una carpeta comprimida con los ficheros creados. Se tendrá en cuenta una correcta estructura.
- El que funcione no implica la máxima calificación. Para toda la tarea se tendrá en cuenta **la calidad de la solución, el uso adecuado de los diferentes recursos del lenguaje y la aplicación de las buenas prácticas**.
- Para esta tarea deberás enviar una **carpeta comprimida con tu nombre y apellidos**. Se espera que el código esté correctamente documentado y comentado.
- Si hubiera partes en las que has buscado información de internet u otras fuentes, es **obligatorio** para calificar el ejercicio que se especifiquen todos los enlaces de dónde se ha extraído la información. Podría ser en un comentario dentro del propio fichero. La omisión de los enlaces o copiar y pegar texto directamente repercutirá en un 0 en la calificación de ese apartado.

FORMATO DE ENTREGA



Nombre_Apellido1_Apellido2, con todos los ficheros dentro, comprimido en:



Nombre_Apellido1_Apellido2.rar o zip

MANIPULACIÓN DE DOCUMENTOS WEB

Vamos a crear el juego de crear parejas. Tendremos un contenedor que tendrá ítems. Estos ítems serán una especie de cartas en las que se mostrará un reverso. El jugador deberá encontrar las parejas haciendo clic en las imágenes.

Al final, el objetivo es hacer una **web con JavaScript**, por lo que durante toda la tarea se tendrá en cuenta:

- Deberás utilizar las etiquetas que proporciona HTML5 respecto a la **web semántica**.
- Tienes libertad para añadir más etiquetas si lo consideras oportuno (div, span, section, etc.). Se tendrá en cuenta una maquetación trabajada. A menudo en los ejercicios se especifica únicamente la base con la que trabajar. Cuanto más trabajada esté la página **mejor calificación** tendrá.
- Tendrá una parte de estilo común a **todas las páginas** y otro **específico**. Cuando se apliquen las reglas a estos estilos comunes, deberán de ser definidas en uno o varios ficheros externos .css e importados a todas las páginas. Se tendrá en cuenta el criterio a la hora de ubicar las reglas, el uso de selectores o el sobrescribir reglas que se adapten mejor al estilo propuesto, etc.
- Deberás usar Grid Layout o Flexbox.

REQUISITO 1 (2 puntos): Documentación. Deberás documentar el funcionamiento de la aplicación. El documento será en formato PDF. Es de obligada realización y entrega para que se realice la corrección de la tarea. En caso contrario la calificación será de 0. Deberás especificar como mínimo:

- Cómo funcionan los puntos clave de la aplicación, mostrando ese bloque de código y realizando una explicación adjunta.
- Dificultades encontradas y formas de solucionarlas.
- Una explicación de las decisiones tomadas en el diseño de la estructura HTML en base a las expectativas de modificaciones realizadas durante el transcurso del juego.

REQUISITO 2 (2 puntos): Flujo del juego. Se deberá cumplir:

- Cuando se haga clic en una imagen, ésta se revelará. Se hará clic en otra para ver si son la misma o no. Si son la misma, se dejarán visibles. Si no lo son, se volverán a ocultar. Utiliza `setTimeout()` para que no sea instantáneo.
- Implementa mensajes de estado para los diferentes eventos, tal como fallar, acertar, partida nueva, etc.
- El orden de los ítems se generará de forma automática en cada partida.
- Cada ítem o imagen estará identificado por un atributo, que será compartido por su pareja. Por ejemplo:

`<img atributo="imagen_perro"...`

Este será el atributo a comprobar cuando se comparen dos ítems.

- Cuando una de las imágenes seleccionadas esté esperando a seleccionar otra carta, se aplicará un estilo distinto a la caja para poder diferenciar el estado en espera de esa primera.

REQUISITO 3 (2 puntos): Control de la partida. Para controlar la partida:

- Antes de la partida el juego nos preguntará el tamaño del tablero (por número de parejas) y el número de intentos fallidos permitidos en cada partida.
- Los intentos permitidos se mostrarán mediante iconos. Cada error borrará un icono. La partida finalizará al agotar todos los intentos.
- Al terminar la partida se nos redirigirá a una página de victoria o derrota, desde la cual igualmente podremos jugar otra partida.
- Se debe poder reiniciar la partida mediante un botón (incluso antes de acabar la partida).

REQUISITO 4 (2 puntos): Límite de tiempo. Habrá un límite de tiempo de la partida especificado también al inicio de la misma. Si se acaba, se perderá la partida igualmente. Idealmente se mostrará el contador. Además, se irán mostrando las estadísticas durante el juego (parejas encontradas de las totales, porcentaje de aciertos, etc.).

REQUISITO 5 (2 puntos): Estilos dinámicos. Implementar un sistema en el que el jugador pueda cambiar el tema de juego. Implementación libre. Se valorará la idoneidad y trabajo de la solución aportada.