

PAC de Desarrollo UF3

CFGS DAW

# Módulo 06: Desarrollo web en entorno cliente

2S 2023/2024



# 1. Normativa

Requisitos necesarios que debe cumplir el trabajo:

- La PAC de Desarrollo se debe entregar únicamente a través del campus, en la tarea específica para ello y dentro de los plazos de entrega establecidos en la guía didáctica. En caso de no cumplir dichos plazos, no se podrán enviar de forma posterior.
- El día y hora máximo para entregar una PAC de desarrollo es el día especificado en la guía didáctica.
- Para realizar esta actividad se debe consultar los contenidos del material didáctico y hacer uso de fuentes bibliográficas y recursos web.
- Las respuestas deben estar debidamente argumentadas. No se admiten respuestas escuetas sin justificar.
- No se aceptan copias literales de Internet, que se considerarán plagio. Se puede utilizar Internet para localizar información, pero el redactado de las respuestas debe ser de elaboración propia, citando la fuente de donde se ha obtenido esta información.
- Se deben referenciar las fuentes consultadas, ya que en caso de no hacerlo podría considerarse plagio. Se recomienda seguir las [normas APA para citar páginas webs](#).
- Es responsabilidad del alumno/a comprobar que el archivo subido en la plataforma es el correcto, ya que en ningún caso el profesor revisará el documento antes del periodo de corrección.
- El documento debe realizarse a ordenador, nunca a mano, y entregar en PDF, no en otros formatos o en editable.
- El documento debe tener portada e índice y el contenido estar estructurado en diferentes secciones, con encabezados y numeración de páginas.

- El documento debe utilizar letra Arial 11, con interlineado 1 o una configuración de estilo similar que facilite su lectura.
- El documento debe redactarse en minúsculas siguiendo las normas ortográficas básicas, con cohesión, coherencia, fluidez del texto y utilizando un vocabulario adecuado al tema. No hacerlo puede penalizar hasta 1 punto la calificación.

## 2. Criterios de corrección

A la hora de realizar la corrección y evaluación del trabajo presentado se tendrán en cuenta los siguientes criterios:

- La calificación numérica oscila del 0 al 10. El profesor podrá disminuir hasta 1 punto la calificación obtenida en caso de que haya errores ortográficos y/o su presentación no se adecúe a los estándares básicos establecidos.
- En ningún caso se corregirán aquellos trabajos que sean una copia literal de Internet u otras fuentes.
- Debe ser un trabajo propio donde se traten todos los conceptos mencionados.
- Si no se entrega una PAC de desarrollo, la calificación equivaldrá a un 0.
- Si se detecta la entrega de una PAC copiada, ya sea de una fuente externa o con un contenido idéntico al de otro alumno/a, serán evaluadas como suspenso, con una calificación de 0.

## 3. Enunciado de la actividad

Esta actividad consiste en la realización de una aplicación web con las siguientes características:

### 3.1 Descripción

Se quiere desarrollar una aplicación web que permita a sus usuarios jugar a Piedra, Papel o Tijera contra la máquina, eligiendo entre las diferentes opciones y permitiendo llevar un historial de los últimos resultados.

### 3.2 Tecnologías

**Solo se podrá utilizar JavaScript.** Todo lo necesario en HTML y CSS ya se suministra en el enunciado. Se adjunta todos los ficheros necesarios para la realización de esta actividad. **No se podrán utilizar otros ficheros y no se podrán modificar los ficheros HTML y CSS que se entregan, el no cumplimiento de esta norma supondrá un 0.**

### 3.3 Objetivos

Objetivo principal: Desarrollar un sistema que permita jugar contra la máquina al Piedra, Papel o Tijera.

Objetivos específicos:

- Desarrollar el sistema de reconocimiento de usuarios y de número de partidas a jugar.
- Desarrollar un sistema que permita al usuario seleccionar una opción para jugar, generar una opción aleatoria para la máquina y conocer el resultado.

- Desarrollar el sistema control del historial de los resultados de cada jugada y restablecimiento de las partidas.

## 3.4 Requisitos

La aplicación se divide en cuatro partes con los siguientes requisitos y comportamientos:

### Configuración de la aplicación

- Asignará al botón ¡JUGAR! el evento que permita realizar las comprobaciones del comienzo de la partida.
- Asignará a todas las imágenes, salvo a la última, el evento que permita seleccionar la opción del jugador y poner en ellas las imágenes que les corresponden. Estas imágenes se generarán a partir del array “posibilidades” que se suministra en la primera línea del fichero JS y se le añadirá la ruta hasta ellas, el indicador que es de “Jugador” y la extensión del fichero.
- Asignará al botón ¡YA! el evento que permita realizar la tirada y calcular el resultado de esta.
- Asignará al botón RESET el evento que permita reestablecer la aplicación.

### Comienzo de partida

- Se proporcionan dos campos de texto donde el usuario introducirá su nombre y el número de partidas que va a jugar.
- Un nombre será válido si tiene más de tres caracteres y el primero de ellos no es un número.
- Una cantidad de partidas será válida si es mayor que 0.

- Al pulsar el botón ¡JUGAR!, realizará las comprobaciones de si son válidos el nombre y el número de partidas, e indicará qué campos son incorrectos utilizando la clase “fondoRojo”.
- Si ambos campos son válidos, eliminará la clase “fondoRojo” (en caso de que se hubieran introducido previamente datos incorrectos), desactivará ambos campos de texto para que no se puedan introducir nuevos datos y volcará el valor introducido en el <span> “total” que indica el número de tiradas totales que se podrán realizar.

### Elección y tirada

- Permitirá elegir entre una de las posibles opciones que se declaren como <img> dentro del <div> del “jugador”.
- Al seleccionar una opción le aplicará el estilo de la clase “seleccionado” y al resto “noSeleccionado”.
- Al pulsar sobre el botón ¡YA! generará una opción aleatoria para el <img> dentro del <div> de la “máquina” y sumará una partida en el <span> “actual” que indica el número de tiradas realizadas hasta el momento.
- La opción aleatoria la generará a partir del array “posibilidades” que se suministra en la primera línea del fichero y se le añadirá la ruta hasta ella, el indicador que es de “Ordenador” y la extensión del fichero.
- Comprobará el resultado de la jugada teniendo en cuenta que:
  - Lo que esté en la primera posición del array “posibilidades” gana a lo que esté en la última.
  - Lo que esté en la posición “n” del array gana a lo que está en la posición “n-1” del array. Es decir, lo que esté en la posición 2 gana a lo que está en la posición 1, y así sucesivamente.

- De esta forma, el código será compatible con cualquier aplicación, siempre que el número de opciones que tenga el jugador corresponda con la longitud que tenga el array.

### Historial de partidas

- Cuando se realice una tirada, aparecerá en el historial el resultado de la jugada:
  - Si gana el jugador, indicará el nombre del jugador introducido al comienzo de la partida. Por ejemplo “Gana Ramón”.
  - Si gana la máquina, el mensaje “Gana la máquina”.
  - En caso de empate, el mensaje “Empate”.
- Cuando se pulse el botón RESET, mostrará el mensaje “Nueva partida” y realizará los siguientes cambios en la aplicación:
  - Volverá a activar los campos de texto del comienzo de partida, dejando a “0” las partidas introducidas y manteniendo el nombre del jugador.
  - Volverá a poner a 0 los contadores de partidas “actual” y “total”.
  - Pondrá la imagen por defecto en la opción de la máquina.
  - Mantendrá el historial de resultados hasta el momento.

## 3.5 Pruebas

Deberá realizarse un documento de pruebas en el que se incluya capturas de la aplicación de cada paso y los comentarios sobre el código que se consideren necesarios para cumplir los requisitos de este enunciado, en el uso de la aplicación en los siguientes casos de uso:

1. Introducción de usuario con datos no válidos.
2. Introducción de cantidad de partidas con datos no válidos.
3. Acceso a la aplicación con datos válidos.
4. Seleccionar una de las opciones y jugar al menos 5 partidas.

5. Pulsar el botón RESET y jugar al menos 3 partidas.

## 4. Evaluación

La evaluación de esta actividad se realizará siguiendo la siguiente rúbrica, y siempre que se hayan cumplido los **requisitos mínimos establecidos en el enunciado de la tarea**, teniendo en cuenta que será **necesario superar los requisitos de un nivel inferior para optar a la calificación del superior**:

| Criterios  | Alta  | Media  | Baja  | Nada  | Peso |
|--|---|--|---|---|------|
| Configuración de la aplicación: botones              | Se ha añadido correctamente el evento correspondiente al botón RESET  | Se ha añadido correctamente el evento correspondiente al botón ¡YA!      | Se ha añadido correctamente el evento correspondiente al botón ¡JUGAR!  | No cumplen los requisitos.  | 0.5  |
| Configuración de la aplicación: opciones del jugador | Se han añadido correctamente los eventos correspondientes a las opciones, y funciona independientemente del número de elementos que haya. | -  | Se han añadido correctamente los eventos correspondientes a las opciones.   | No cumplen los requisitos.  | 0.5  |
| Comienzo de partida                                  | Se cumplen todos los requisitos cuando se pulsa el botón ¡JUGAR!  | Se comprueba correctamente el nombre de usuario y su indicador de error. | Se comprueba correctamente el número de tiradas y su indicador de error.  | No cumplen los requisitos.  | 1    |
| Elección y tirada                                    | Se cumplen todos los requisitos cuando se pulsa el botón ¡YA!   | Se genera y muestra un valor aleatorio para la máquina.                  | Se puede seleccionar una de las opciones teniendo en cuenta los requisitos de apariencia de la selección, a partir del array sin modificar. | No cumplen los requisitos.  | 1.5  |
| Historial de partidas                                | Se cumplen todos los requisitos cuando se pulsa el botón RESET.   | Al pulsar el botón RESET, se muestra el mensaje.                         | Cada vez que se pulsa el botón ¡YA! se muestra el resultado de la tirada.   | No cumplen los requisitos.  | 1    |
| Pruebas  | Se incluyen todas las pruebas que se piden.   | Se incluyen bastantes de las pruebas que se piden.                       | Se incluyen algunas de las pruebas que se piden.  | No se incluyen pruebas o están desorganizadas o no se ven correctamente o no incluyen todos los pasos que comprueben su superación. | 0.5  |



## 5. Entrega

Sigue las siguientes instrucciones para realizar correctamente la entrega del proyecto:

- Un fichero comprimido con la nomenclatura PAC\_UF3\_ApellidosNombre.zip (siendo Apellidos y Nombre tus datos personales) y que tenga en su interior lo siguiente:
  - Una carpeta que contenga el código que se pide en esta PAC de desarrollo (HTML, CSS y JS), **con todo lo necesario para poderlo visualizar y ejecutar correctamente.**
  - El fichero en formato .pdf con la siguiente nomenclatura:  
PAC\_UF3\_ApellidosNombre.pdf
- **Revisa bien que el formato del documento siga lo que se especifica en la normativa.**
- **Para que el ejercicio sea corregido, este debe entregarse obligatoriamente en la tarea y en el formato estipulado.** No se corregirán proyectos entregados por otros medios, como los adjuntos de los comentarios de la tarea o los mensajes privados.