

**Convocatoria:** Extraordinaria

**Ciclo:** Desarrollo Aplicaciones Web

**Módulo:** Desarrollo Web en Entorno Servidor

## JUEGO DEL AHORCADO

La aplicación base consiste en dos proyectos, por un lado, el proyecto `dwes_23_2` que corresponde a la aplicación del juego del ahorcado que debe ser actualizada con las modificaciones funcionales solicitadas en el enunciado.

El juego del ahorcado consiste en descubrir una palabra secreta enviando una a una las letras que pensamos que componen dicha palabra. Cada fallo supone añadir una parte del dibujo del ahorcado. Si la figura del ahorcado se completa entonces el jugador pierde la partida. En esta implementación del juego no se usan tildes ni la letra ñ.

Antes de comenzar la prueba asegúrate que has creado la base de datos `hangman` formada por una tabla `usuarios` que contiene una fila (1, 'pepe', '123456', 'pepe@gmail.com')

El examen consiste en realizar dos modificaciones a la aplicación del juego del ahorcado aportado como punto de partida. **Al final de la prueba se entregará un único proyecto con todos los cambios realizados.**

### 1 Permitir baja de usuario. (3 ptos)

Se trata de que la aplicación permita a los usuarios darse de baja del juego.

Esta funcionalidad consiste en lo siguiente:

- El acceso a esta funcionalidad se realiza por medio del enlace *Baja* que aparece en el menú desplegable asociado al nombre del usuario creado en la vista *app* y que aparece en la parte superior derecha de la página del juego.
- La petición se envía al script de control *index.php* con el parámetro *botonbaja* activado para que se procese el borrado del usuario de la base de datos y el cierre de la sesión en curso.
- Una vez procesada la baja se muestra la vista con el formulario de login llamada *formlogin* con un mensaje de confirmación *Baja* realizada con éxito.
- Se puede recrear el usuario *pepe* borrado desde el interfaz de *phpMyAdmin* para seguir trabajando con la aplicación.

### 2 Solicitud de partida personalizada. (7 ptos)

Se trata de implementar el mecanismo por el cual el jugador puede solicitar una partida personalizada aportando **de manera opcional** la longitud mínima, máxima y un máximo de tres letras que queramos que aparezcan en la palabra secreta.

**Se simplifica la funcionalidad asumiendo que siempre vamos a encontrar palabras en el almacén que cumplan los requisitos personalizados.** Se recomienda solicitar palabras con parámetros fáciles de cumplir.

La funcionalidad consiste en lo siguiente:

- Se debe activar la funcionalidad pinchando en el enlace *Nueva Partida Personalizada* que aparece en la vista de *juego*.
- Crear la vista llamada *formpartidapersonalizada* con un formulario para solicitar los siguientes parámetros no requeridos. Se adjunta una imagen que muestra la vista. No hace falta que respetes los formatos en su totalidad.

- Longitud mínima de la palabra. Campo de entrada de tipo *number*. (<input type="number" ....)
- Longitud máxima de la palabra. Campo de entrada de tipo *number*. (<input type="number" ....)
- Letras contenidas en la palabra. Campo de entrada de tipo "text". (<input type="text" ....)
- Fíjate en el enlace *Volver* que nos devuelva a la vista del juego si queremos abortar la operación.
- El formulario es validado (en cliente o servidor) realizando las siguientes comprobaciones:
  - Longitud mínima entre 3 a 14.
  - Longitud máxima entre 5 a 20.
  - Longitud mínima menor que longitud máxima.
  - Letras contenidas en la palabra contiene una cadena de tres letras como máximo.
- Si el servidor detecta un error en los datos del formulario se muestra de nuevo el formulario con los valores introducidos y los siguientes mensajes de error personalizados:
  - "El valor de la longitud mínima debe estar entre 3 a 14"
  - "El valor de la longitud máxima debe estar entre 5 a 20"
  - "El valor de longitud mínima es mayor que el valor de longitud máxima"
  - "El valor de letras contenidas en la palabra debe tener 3 letras como máximo"
- La especificación del constructor de la clase Hangman deberá de modificarse para que reciba los requisitos como parámetros opcionales. Por otro lado, la implementación también cambia ya que deberá solicitar palabras al almacén hasta que reciba una palabra que cumpla los requisitos.

 Ahorcado

Volver

pepe ▾

### Partida Personalizada

Longitud  
Mínima

Longitud  
Máxima

Contiene las  
letras

Enviar

## Guía de evaluación

### Permitir baja de usuario. (3 ptos)

#### **Baja de usuario. (1 pto)**

La petición de usuario se procesa y el usuario es borrado de la base de datos

#### **Bonus – El usuario es eliminado por un objeto de la clase *UsuarioDAO* . (1.5 pto)**

La eliminación del registro se realiza por un objeto de la clase *UsuarioDAO* usando *PDO*.

#### **Muestra de la vista *formlogin* con mensaje de confirmación de baja. (0.5 pto)**

Cuando se da de baja al usuario se le muestra un mensaje de confirmación tal y como pide el enunciado.

### Solicitud de partida personalizada. (7 ptos)

#### **Solicitar una nueva partida personalizada. (1 pto)**

El jugador pulsa en el enlace de *Nueva Partida Personalizada* y el script *juego.php* recibe la petición y la trata en una rama de ejecución para enviar la vista del formulario de requisitos.

#### **Creación de la vista *formpartidapersonalizada* para personalizar la palabra. (1 pto)**

Se crea la vista *formpartidapersonalizada* para recoger los requisitos personalizados de la palabra con la que jugar.

#### **Bonus – Creación de la vista *formpartidapersonalizada* con BladeOne. (1 pto)**

La vista *formpartidapersonalizada* se ha creado con el motor de vistas BladeOne. La vista *formpartidapersonalizada* hereda de la vista *app* que contiene los aspectos comunes del resto de vistas.

#### **Validación de los valores del usuario incluidos en el formulario. (2 ptos)**

Realizar las validaciones requeridas en el enunciado sobre los valores escritos en el formulario. Incluir los mensajes personalizados de error en el formulario para que sean corregidos.

#### **Obtención de una palabra secreta que cumple con los requisitos. (2 ptos)**

Adaptar el código del constructor de la clase *Hangman* para que se soliciten palabras al almacén hasta que se obtenga una palabra que cumpla con los requisitos personalizados para jugar la partida.