

Convocatoria: Ordinaria

Ciclo: Desarrollo Aplicaciones Web

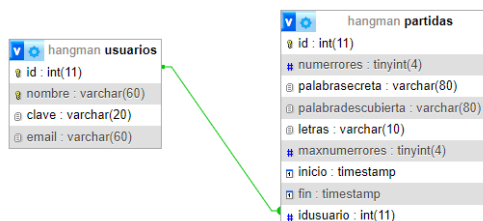
Módulo: Desarrollo Web en Entorno Servidor

JUEGO DEL AHORCADO

La aplicación base consiste en un proyecto creado con *Apache Netbeans* que corresponde a la aplicación del juego del ahorcado que debe ser actualizada con las modificaciones descritas en este 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 de que has creado la base de datos *hangman* formada por una tabla *usuarios* que contiene dos filas; (1, 'pepe', '123456', 'pepe@pepe.es') y (2, 'luis', '654321', 'luis@luis.es') y una tabla *partidas* que está vacía. Puedes encontrar el esquema de la base de datos en la carpeta *bd* del proyecto suministrado.



La prueba consiste en extender la funcionalidad del juego del ahorcado aportado como punto de partida. **Al final de la prueba se entregará un único proyecto con todos los cambios realizados.** Las modificaciones/extensiones a realizar son las siguientes:

1 Persistencia de las partidas del juego en la BD. (5 ptos)

Se trata de que la aplicación persista las partidas jugadas por cada jugador en la base de datos. Las partidas se persisten cuando se crean para el juego (**inserción** de un nuevo registro que corresponde a la nueva partida) y cuando se acaban, ya sea porque la partida se gana, se pierde o se abandona solicitando una nueva partida (**modificación** del registro que corresponde a la partida con los datos finales de la misma). No hace falta persistir los cambios de la partida mientras está en juego.

La funcionalidad se implementará siguiendo estas directrices:

- Extender la clase *Partida* con los campos *idUsuario*, *inicio* y *fin* para recoger el id del usuario que está jugando la partida, la fecha y hora (DateTime) del inicio de la partida y la fecha y hora (DateTime) del fin de la partida.
- Crear la clase *PartidaDAO* para dar persistencia a los objetos de la clase *Partida* e implementar los métodos *crear* y *modificar* para realizar la inserción y actualización de registros de la tabla *partidas* a partir de los objetos de la clase *Partida* utilizados en la aplicación. La implementación de los métodos debería realizarse con consultas preparadas de la librería PDO. La base de datos ya dispone de una tabla *partidas* para almacenar las partidas jugadas.
- Completar el script de control *juego.php* para que realice las operaciones de persistencia al


comienzo, fin y abandono de cada partida.

2 Solicitud de resolución de partida. (4,5 ptos)

Se trata de implementar el mecanismo por el cual el jugador puede resolver una partida enviando una palabra completa en cualquier momento del juego. La palabra propuesta por el jugador se escribe en la misma caja de entrada que las letras y se pincha en el botón de resolver que se muestra a la derecha de ésta. Una vez comprobada la palabra se informará al jugador con un mensaje (*Enhorabuena!* o *Has perdido!*) de la misma manera que se hace con el juego normal. También se mostrará la palabra secreta y se deshabilitarán los botones de *Enviar jugada* y *Resolver* tal y como se puede ver en la imagen al final de la sección.

La funcionalidad se implementará mediante *AJAX* siguiendo estas directrices:

- El archivo *public/assets/js/resolver.js* contiene la plantilla del código *JavaScript* necesario para implementar la comunicación *AJAX* entre el cliente y el servidor en *JQuery* y *JavaScript* puro. Se debe de implementar el cuerpo de la función *resolverPartida* en la modalidad preferida (*JQuery* o *JavaScript* puro). Se proporciona el código que crea el manejador de eventos sobre el botón y las funciones que muestran la información en la página del juego, las funciones que muestran la información en la página y las funciones que deshabilitan los botones una vez solicitada la resolución de la partida. Es necesario comentar el código en la modalidad que no se vaya a utilizar.
- El cliente rellena la caja de texto y envía la solicitud de resolución de la palabra hacia el script *juego.php*. El script recoge la petición responde con un objeto *JSON* con dos campos que representan el resultado de la comprobación y la palabra secreta con la que se juega.
- La comprobación de la palabra se realiza mediante un nuevo método de la clase *Partida* llamado *compruebaPalabra* que recibe como parámetro de entrada la palabra del jugador y modifica el estado de la partida de la siguiente forma: Si la palabra es correcta se cambia la propiedad de *palabraDescubierta* sustituyendo todos los guiones por las letras correspondientes, si la palabra no es correcta, se modifica la propiedad *numErrores* al máximo número de errores. El método devuelve un valor booleano que indica si la palabra es correcta o no.
- Cuando el cliente recibe la respuesta del servidor usa las funciones *JavaScript* proporcionadas para mostrar la información y deshabilitar los botones.
- El jugador puede seguir jugando si pincha en *Nueva Partida*.

 Ahorcado

Nueva Partida

pepe

Enhorabuena!

TAGARNINA


✓

Enviar Jugada

Resolver

Las letras introducidas hasta el momento son:

S G C R A L E I N



Guía de evaluación

Es muy importante tener en cuenta que, para puntuar en las secciones descritas en la guía de evaluación, el código aportado deberá funcionar de acuerdo con lo especificado en el enunciado. No se tendrá en cuenta código que no implemente la funcionalidad requerida.

Persistencia de las partidas del juego en la BD (5 pts)

Extensión de la clase Partida

Extensión de la clase Partida con los campos idUsuario, inicio y fin con los tipos de datos correctos.

Puntuación máxima: 0,5 pts

Creación de la clase PartidaDAO

Creación de la clase PartidaDAO con los métodos crear y modificar para realizar la inserción y modificación de un registro de la tabla partidas. Cada método implementado correctamente aporta 1,5 pts. Se penalizará la nota si no se usan consultas preparadas con PDO.

Puntuación máxima: 3 pts

Extensión del script de control *juego.php* para realizar la persistencia

Se completa el código del script de control *juego.php* para que las operaciones de persistencia se realicen en los momentos adecuados del juego.

Puntuación máxima: 1,5 pts

Solicitud de resolución de partida (4,5 pts)

Comunicación por AJAX para resolver partidas

Se programa la función en JavaScript (jQuery o JavaScript puro) para realizar la comunicación entre el cliente y el servidor.

Puntuación máxima: 1,5 pts

Creación del método *compruebaPalabra*

Se añade un nuevo método a la clase Partida llamado *compruebaPalabra* para comprobar si la palabra del jugador es correcta o no.

Puntuación máxima: 1,5 pts

Extensión del script *juego.php* para responder a la petición AJAX

Se responde a la petición AJAX con información sobre la comprobación de la palabra.

Puntuación máxima: 1 pto

Gestión de la respuesta del servidor en la petición AJAX

Se recibe la respuesta del servidor y se procesan los datos actualizando la página según corresponda.

Puntuación máxima: 0,5 pts