

**Convocatoria:** Ordinaria

**Ciclo:** Desarrollo Aplicaciones Web

**Módulo:** Desarrollo Web en Entorno Servidor

## Instrucciones para la realización de la prueba.

Las instrucciones que se detallan a continuación son solamente relevantes para la configuración de los equipos del centro, también para cualquier equipo que replique esa misma configuración. Si el alumno/a utiliza otro tipo de entorno debe asegurarse que tiene conocimientos para realizar estas mismas operaciones en su entorno.

### 1 Al inicio de la prueba

- Conectarse al aula virtual y descargar el proyecto en formato zip alojado en la tarea con el nombre “Prueba Final Ordinaria”. En el caso en que no haya acceso al aula virtual se habilitará un mecanismo alternativo para obtener el proyecto de código inicial. Para ello es necesario disponer de las claves del aula virtual. Se recomienda que se traigan apuntadas, aunque ya puedan estar guardadas en el PC por si hay que conectarse desde otra máquina.
- Descomprimir el archivo en la carpeta en la ubicación elegida, por ejemplo (D: o E:). La unidad C: está protegida y se restaura después de cada reinicio del PC por lo que no se recomienda copiar el proyecto en ninguna carpeta en C:. Si se decide desplegar en C:/xampp/htdocs es recomendable hacer copias de seguridad del proyecto en (D: o E:) cada cierto tiempo para no perder el trabajo si el PC se bloquea.
- No es necesario descargar ninguna librería con composer porque ya están todas las librerías necesarias en la carpeta “vendor” del proyecto.
- Para asegurarse de que la configuración de la carga automática de composer es correcta, situarse en la carpeta del proyecto con la herramienta de símbolo del sistema (cmd) y ejecuta el comando: composer dump-autoload
- Abrir el proyecto en el IDE Apache Netbeans.
- Arrancar el panel de control de XAMPP a través del menú de programas de Windows.
- Arrancar Apache y MySQL desde el panel de control de XAMPP, pinchando sobre los botones “Start”.
- Arrancar phpMyAdmin desde el panel de control de XAMPP, pinchando sobre el botón “Admin” asociado a MySQL.
- Importar el esquema de la base de datos de la aplicación pinchando en la pestaña “Importar” en el menú superior. Una vez en la página de importación pinchar en el botón “Seleccionar archivo” y navegar hasta la carpeta “bd” del proyecto descomprimido. Seleccionar el fichero con extensión “.sql” donde reside el esquema y pulsar el botón “Continuar” situado abajo a la derecha. Ignora los errores #1045 Base de datos no seleccionada que puedan ocurrir.

### 2 Durante la prueba

- Se aconseja encarecidamente que después de terminar el desarrollo de cualquier punto solicitado en el enunciado se realice una copia de seguridad del proyecto en la unidad D:. Se trata de evitar que el código añadido en una de las partes de la prueba introduzca errores que

no permitan corregir otras áreas de la prueba. Es muy importante poder volver a un estado en el que proyecto se ejecuta correctamente y muestra nuestro progreso en la prueba.

- **Se debe añadir el nombre y apellidos del alumno/a como comentario en todos los ficheros que sean editados, ya sea porque modificamos su contenido o porque decidimos añadirlos nuevos. Aquellos cambios introducidos en ficheros en los que no se ha añadido el nombre no se tendrán en cuenta durante la corrección.**

### **3 Al final de la prueba**

- Comprimir la carpeta del proyecto y su contenido con zip o rar. Nombrar el fichero de la siguiente forma: nombre\_apellido1\_ordinaria.zip (.rar)
- Subir el archivo .zip al aula virtual en la tarea “Prueba Final Ordinaria”.

## Enunciado Convocatoria Ordinaria

### Juego del Ahorcado

El juego de 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 el juego no se usan letras con tildes.

Antes de comenzar la prueba asegúrate que has creado la base de datos “hangman” que consiste en 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 autoregistro de usuarios. (6 ptos)

Se trata de que la aplicación permita que los usuarios se registren de manera automática. Esta funcionalidad consiste en lo siguiente:


- El acceso al formulario de registro se realiza por medio del enlace “Regístrate” que aparece en la página de inicio, en la vista “formlogin”.
- Implementación de una vista que corresponda con un formulario de registro que requiere el nombre de usuario, clave y correo electrónico. Los datos del formulario deben ser validados según las siguientes reglas:
  - El nombre de usuario es obligatorio y debe estar compuesto por una secuencia de 3 a 15 caracteres de palabra. La expresión regular es “/^\w{3,15}\$/”.
  - El mensaje de error asociado será: “El nombre de usuario debe tener entre 3 y 15 letras sin blancos”.
  - La clave es obligatoria y debe estar compuesto por 6 dígitos. Realiza la comprobación sin expresiones regulares y usando funciones y/o el filtro de validación adecuado en PHP.
  - El mensaje de error asociado será: “El password debe estar compuesto por 6 dígitos”.
  - El correo es opcional y debe tener un formato válido. Realiza la comprobación sin expresiones regulares y usando funciones y/o el filtro de validación adecuado en PHP.
  - El mensaje de error asociado será: “El email debe tener el formato correcto”.
- Si los datos no son válidos se volverá a mostrar el formulario relleno con los datos ya introducidos y con información sobre del campo incorrecto.
- Si los datos son correctos se almacena el nuevo usuario en la base de datos.
- Una vez realizado el registro se invoca la vista de login para que el nuevo usuario use sus credenciales para entrar en la aplicación.
- Incluir el control de esta funcionalidad en el script index.php.

## 2 Permitir al jugador solicitar una pista. (4 ptos)


Se trata de implementar el mecanismo por el cual el jugador puede solicitar pistas para resolver la palabra oculta. Se pueden solicitar tantas pistas como se quiera. Esta funcionalidad consiste en lo siguiente:

- El acceso a la funcionalidad se realiza pinchando en el botón de pista que aparece en la vista “juego”. El botón de deshabilita una vez que termina la partida.
- El cliente envía la solicitud de la pista al servidor y éste devuelve una letra que pueda ayudar al jugador a descubrir la palabra. La letra se muestra debajo de la lista de letras introducidas y desaparece una vez que se usa.
- Crea algún método de la clase del modelo Hangman para que calcule la pista que corresponde a la letra no descubierta que más ocurrencias tiene en la palabra, en caso de empate se elige la primera letra en orden alfabético.
- Incluir esta funcionalidad en el script juego.php

Se muestra una imagen de cómo debe de aparecer la información sobre la pista.

 Ahorcado Nueva Partida david ▾

A \_ \_ T



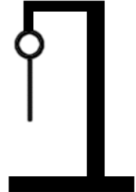
Enviar Jugada

Pista

Las letras introducidas hasta el momento son:

A E T

La pista solicitada es: U



## **Guía de evaluación Convocatoria Ordinaria**

### **Permitir autoregistro de usuarios**

#### **Formulario de autoregistro (1 pto)**

El formulario de autoregistro contiene los campos adecuados y puede ser rellenado y enviado a la aplicación.

#### **Bonus - Formulario de autoregistro creado con BladeOne (1 pto)**

El formulario se ha creado mediante una vista BladeOne.

#### **Validación de datos introducidos en el formulario de autoregistro (1,5 ptos)**

La aplicación valida los datos introducidos en los campos del formulario ya sea en el cliente o en el servidor mostrando los mensajes de error requeridos. La correcta validación de cada campo aporta 0,5 ptos.

#### **Registro de usuario en la BBDD (1 pto)**

El script index.php recibe los datos del formulario y los usa para almacenar un nuevo usuario en la tabla usuarios de la base de datos

#### **Bonus - El controlador usa objetos del modelo o repositorios para la operación de persistencia. (1,5 pto)**

La inserción en la base de datos se realiza a través del objeto del modelo Usuario o un objeto repositorio de usuarios usando la librería PDO

### **Permitir al jugador solicitar una pista**

#### **El jugador puede solicitar una pista a la aplicación. (0,5)**

El jugador pulsa en el botón de la pista y el script juego.php recibe la petición para solicitar la pista.

#### **Bonus La petición llega a la aplicación por medio de AJAX (1)**

La comunicación se realiza mediante AJAX

#### **El controlador solicita al objeto del modelo que le proporcione los datos de la pista (1,5)**

Se implementa un método en la clase Hangman para programar el cálculo correcto de la letra escogida como pista.

#### **El cliente muestra la información del juego con los datos correspondientes a la pista. (0,5)**

La información correspondiente a la pista se muestra en la vista, es decir, la letra sugerida se muestra debajo de la lista de letras introducidas.

#### **El botón de pista se deshabilita al final de la partida. (0,5)**

El botón de la pista se deshabilita cuando la partida finaliza y se activa cuando se inicia una nueva partida.