

Prueba Técnica

Full Stack Developer Prat Brands

MARC PROVINCIALE ISACH

Datos de contacto:

provinciale_marc2@hotmail.com

1. Instrucciones de instalación

1. Instalar un servidor web donde poder ejecutar proyectos de manera local. En mi caso he utilizado el apache que ya viene instalado con el wamp_server.
2. Tener una versión de PHP $\geq 7.4.24$, la que he utilizado es la 7.4.24.
3. Descargar el proyecto en formato rar o del repositorio git https://github.com/M-Provi/test_backend.git
4. Poner el proyecto en la carpeta raíz www del servidor web para poder ser ejecutado.
5. Inicializar los servicios necesarios para ejecutar, en este caso el del servidor web.
6. En mi caso no he hecho ninguna modificación del nombre de la url, así que para acceder desde cualquier navegador simplemente hay que escribir localhost/test_backend

2. Tecnologías utilizadas

Como había una serie de restricciones para realizar la prueba no se ha utilizado ninguna tecnología relevante.

Para la base de datos como se especificaba que se tenían que recoger los datos de un fichero en lugar de una base de datos he creado un fichero superheroes.json para facilitar la obtención de datos.

Para desarrollar el backend he seguido el patrón MVC con POO en el modelo de superhéroes.

Para el frontend he utilizado lenguajes web base, en este caso HTML, CSS y JS ya que no estaba seguro de si se podía utilizar laravel para todo el desarrollo del proyecto y he preferido mantenerme sin ningún framework.

3. Información relevante

Aquí comentaré los aspectos relevantes sobre decisiones de diseño o funcionalidades de la solución propuesta.

Primero de todo en la página principal se puede seleccionar el número de jugadores que va a tener la partida, el nombre de los jugadores y el número de vidas que se va a tener para averiguar el superhéroe.

En el juego pueden jugar todos los usuarios que se deseen, especificando o no el nombre de estos, ya que si no se especifica se añadirán por defecto ciertos usuarios con el nombre default 1, default 2...

Si se añade un nombre de jugador y hay más nombres que número de jugadores especificados este se va a incrementar. Si se quiere disminuir la lista de jugadores simplemente con reducir el número de jugadores seleccionará los N primeros.

Tanto el número de vidas como el número de jugadores siempre será mínimo 1.

Al inicializar el juego se selecciona un superhéroe de forma aleatoria, empieza el primer jugador que se ha especificado o el default 0 en caso de no haber y se tiene que ir introduciendo las letras en la casilla correspondiente.

Cuando se introduce una letra se comprobará si esta es correcta, si lo es se descubrirá esa letra en el colgado, y si no lo es se restará una vida del número de vidas total y se pasará al siguiente jugador. Además, se añadirá la letra a las letras utilizadas.

Al finalizar el juego se actualiza la pantalla mediante JavaScript mostrando si se ha ganado o se ha perdido, el superhéroe que era y si se quiere crear un nuevo juego o volver a jugar con los mismos parámetros.

Al decidir crear un nuevo juego se redirecciona a la pagina principal borrando cualquier dato de sesión.

Al decidir volver a jugar se utilizan los mismos parámetros que se han especificado anteriormente.

Todas las llamadas de pase de parámetros para crear un nuevo juego se han hecho a través del método POST y aprovechando también la funcionalidad de PHP `$_SESSION`.

También se ha creado un sistema de layouts para facilitar el mantenimiento de estos, y se ha intentando unificar todo el diseño visual y hacerlo responsive gracias a Bootstrap.

En el modelo de Superhéroes se han añadido las funciones básicas de getters y setters, además de funciones específicas como pueden ser obtener todos los superhéroes y uno aleatorio. También se podría haber añadido la opción de guardar o eliminar superhéroes de la base de datos que simplemente se podría hacer a través de especificar un url como por ejemplo `testbackend/superhero/save` para llamar al controlador de superhéroes y al método `save` donde se guardaría en la base de datos o el fichero json los datos del superhéroe especificados ya se por POST o GET. Y un procedimiento parecido para eliminar y modificar.

El controlador he decidido utilizar un controlador principal ya que no se tenía que gestionar prácticamente ningún modelo y vista suficientemente específicos, y para los dos métodos que hay no hacía falta más.

4.Pruebas realizadas

Las pruebas que he realizado han sido las siguientes:

- Intentar introducir parámetros no posibles como números, números negativos, letras u otros en campos input.
- Inicializar un juego especificando número de jugadores, nombre de los jugadores y número de vidas.
- Inicializar un juego especificando el número de jugadores y el número de vidas.
- Intentar inicializar un juego sin especificar campos.
- Comprobar que se obtienen superhéroes aleatorios en cada partida.
- Comprobar que si no se especifican jugadores o no los suficientes el resto sea default.
- Comprobar que el juego inicializa correctamente con el superhéroe especificado.
- Comprobar que solo se pueden introducir letras en el juego.
- Comprobar que se gana al introducir el superhéroe correcto.
- Comprobar que se pierde al perder todas las vidas.
- Comprobar que el juego se reinicia con los mismos parámetros pero superhéroe diferente al darle a jugar de nuevo.
- Comprobar que el juego se redirecciona a la página principal al pulsar nuevo juego.
- Comprobar que todos los aspectos visuales se actualizan correctamente según las acciones que haya hecho el usuario como añadir un jugador, ganar/perder o introducir una letra.