

PROGRAMACIÓN DEL CLIENTE WEB. PRÁCTICA 3.

Objetivos de la práctica

- Aprender a utilizar los APIs de HTML.
- Aprender a integrar todos los conocimientos adquiridos durante el curso en una única aplicación web y conseguir que interactúen entre sí de la manera más eficiente y productiva posible.

Enunciado de la práctica

Esta práctica es independiente de las dos anteriores. Debéis implementar el conocido juego del sudoku (<https://es.wikipedia.org/wiki/Sudoku>). En este enunciado se explica cómo debe ser el funcionamiento del juego que se pide para la práctica.

IMPORTANTE: No se permite utilizar ningún framework, ni ninguna otra herramienta que no se haya visto en la asignatura. Tampoco se permitirán implementaciones de este juego que no se hayan realizado como se indica en este enunciado.

Descripción del juego

El sitio web constará de las siguientes páginas:

- **index.html.** Esta será la página inicial de la práctica y en ella aparecerá todo lo necesario para el juego. En el apartado de tareas a realizar se detalla todo lo que debe aparecer en esta página.
- **acerca.html.** Al igual que en el resto de las prácticas, esta página contendrá la información sobre el autor o autores de la práctica, así como también cualquier otra información relacionada con el desarrollo de la misma: partes realizadas y sin realizar, si fuera el caso; problemas encontrados y solución encontrada; etc..

El juego a desarrollar es el típico juego del sudoku. En esta implementación se permitirá al usuario elegir entre dos tamaños: 4 y 9. El juego comenzará en la página `index.html`, en la que se mostrará el tablero del juego, del tamaño seleccionado por el usuario, pero vacío, junto a un botón para empezar el juego.

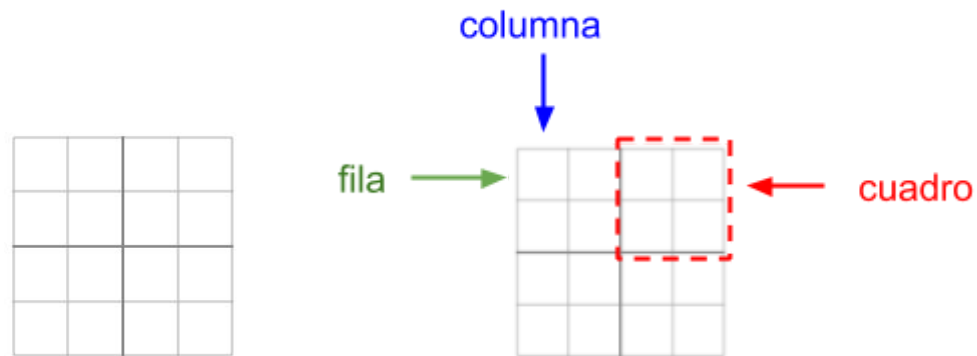
Junto al enunciado de la práctica se os proporciona un servidor RESTful (igual que se hizo en la práctica 2) muy simple. El servidor realiza dos funciones básicas: generar sudokus del tamaño que se le indique y comprobar los errores que tengan los números que ha puesto el usuario. Como siempre, al final de este enunciado se os proporciona la lista de peticiones que podéis hacer al servidor.

También se os proporciona un script de creación de la base de datos del juego, en la que el servidor guardará los sudokus creados, de manera que pueda haber más de una partida jugándose al mismo tiempo. El servidor también accederá a la base de datos para comprobar los errores cometidos por el jugador de un sudoku determinado y eliminar los

sudokus finalizados.

Tareas a realizar

- 1) (0,25 puntos) Todas las páginas debe pasar satisfactoriamente y en todo momento la validación HTML en <http://validator.w3.org>.
- 2) (0,5 puntos) Todas las páginas tendrán en común la siguientes partes, además de las propias de cada una:
 - Cabecera con: logotipo, nombre/título del sitio web y un pequeño texto explicativo del objetivo del juego.
 - i. En el caso de la página `acerca.html`, al hacer clic en el logotipo iremos a la página `index.html`.
 - Pie con, al menos, información de contacto, copyright, año de creación.
 - i. En el caso de la página `index.html`, la información de contacto será un enlace a la página `acerca.html`, en la que se mostrará la información completa del autor o autores de la práctica y la documentación de la misma.
 - ii. En el caso de la página `acerca.html`, el pie de página incluirá un botón/enlace que permitirá volver a la página `index.html`.
- 3) (0,25 puntos) El diseño de las páginas se hará utilizando la técnica *Mobile First*, permitiendo la correcta visualización de todo el contenido de la página en todo momento.
- 4) (0,75 punto) Página `index.html`. Al cargar la página debe mostrar la zona de juego en la que habrá lo siguiente:
 - Un título.
 - Un elemento que permita al usuario seleccionar el tamaño del sudoku. Por simplificación solo podrá elegir entre 4 y 9.
 - Un botón para empezar a jugar.
 - Tablero de juego (implementado con canvas), con las divisiones de filas y columnas correspondientes al tamaño seleccionado. Las divisiones de los cuadros principales deben estar en un color más oscuro, tal y como se muestra en la siguiente imagen (izquierda) para un tamaño 4 de sudoku. En la parte derecha de la imagen se identifica lo que es la fila, la columna y el cuadro.



5) (0,25 puntos) Página **acerca.html**. Al igual que en las prácticas anteriores, tendrá la siguiente información:

- Texto indicando que se trata de la práctica 3 de la asignatura Programación del Cliente Web, que pertenece al 2o curso de la titulación del Grado en Ingeniería Multimedia.
- Asimismo, aparecerá la información relativa al autor o autores de la práctica (dni, nombre, apellidos, grupo de prácticas y correo electrónico).
- Cualquier comentario al respecto del desarrollo de la práctica que se considere relevante, como pueden ser problemas de compatibilidad entre navegadores, partes de la práctica no realizadas, etc.

6) **Funcionamiento del juego.**

- **Inicio.** Página **index.html**.

a) (0,5 puntos) Elemento para seleccionar el tamaño del sudoku. Cada vez que el usuario cambie el tamaño se deberá pintar nuevamente el sudoku con el tamaño seleccionado, siempre y cuando no se esté jugando. Si se está jugando una partida, el elemento para cambiar el tamaño estará desactivado y no se podrá cambiar el tamaño.

b) Botón **empezar**. Cuando el usuario pinche el en botón **empezar**, deberá realizarse lo siguiente:

- (0,25 puntos) Desactivar el elemento para el cambio de tamaño.
- (0,5 puntos) Coger el tamaño seleccionado en el elemento para el cambio de tamaño y hacer una petición ajax/fetch de tipo POST al recurso `api/sudoku/generar/{TAMAÑO}` del servidor. El servidor generará un sudoku del tamaño indicado y lo devolverá en formato JSON (ver apartado de peticiones al final del enunciado). Junto al sudoku, el servidor devolverá un token que servirá para identificar la partida a la que pertenece el sudoku y que habrá que enviar como cabecera `Authorization` en el resto de operaciones que se pueden realizar sobre él.
- Una vez se tenga la información devuelta del servidor, habrá que realizar lo siguiente:
 - (0,25 puntos) Guardar en `sessionStorage` la información devuelta para

poder recuperarla cuando sea necesario.

- 2) (0,5 puntos) Pintar el sudoku de manera que los números fijos proporcionados aparezcan con el fondo de un color destacado y los huecos aparezcan en otro color, tal y como se puede ver en la siguiente imagen:

4			
	3		4
			3
3	4	2	1

- 3) (0,25 puntos) El botón de empezar se sustituirá por dos botones: uno para comprobar errores y otro para finalizar la partida antes de tiempo.
- 4) (0,5 punto) Se mostrará un contador de tiempo de juego al usuario que mostrará en todo momento el tiempo que se está invirtiendo en resolver el sudoku. Este contador de tiempo mostrará en el formato hh:mm:ss las horas, minutos y segundos transcurridos desde el inicio de la partida. Solo se detendrá cuando finalice el juego.

c) Botón para comprobar errores.


- i) (0,25 puntos) Cuando el usuario pinche este botón, se enviará una petición de tipo POST a la url `api/sudoku/{ID_SUDOKU}/comprobar`, pasando la cabecera `Authorization` con el valor del token recibido al crear el sudoku (ver apartado de peticiones al final del enunciado). Además, deberá pasarse como parámetro `juego` el array del sudoku con las jugadas (números puestos) del usuario.
- ii) (0,5 puntos) En el objeto respuesta del servidor vendrá una propiedad `FALLOS` que contiene una lista de errores en forma de array de objetos javascript en los que se indica la fila y columna de la celda errónea.
- 1) Habrá que destacar el fondo de las celdas erróneas con un color distinto al resto de celdas, por ejemplo rojo.
- 2) Estas celdas dejarán de estar destacadas en cuanto el cursor del ratón entre en el canvas del sudoku.

d) Botón para finalizar. Permitirá al usuario finalizar la partida antes de tiempo. Cuando se pinche este botón se deberá hacer lo siguiente:

- i) (0,25 puntos) Hacer una petición de tipo DELETE a la url `api/sudoku/{ID_SUDOKU}`, pasando la cabecera `Authorization` con el valor del token recibido al crear el sudoku (ver apartado de peticiones al final del enunciado), para que el servidor elimine de la BD la partida correspondiente.
- ii) (0,5 puntos) Al recibir la respuesta del servidor, si la petición se ha realizado correctamente y el servidor ha eliminado la partida de la BD, se deberá parar el contador de tiempo y “reiniciar” el juego, dejando la página

y el sudoku mostrado al usuario listo para empezar a jugar otra partida, tal y como se indica en el apartado 4.

7) Interacción con el sudoku (canvas) cuando se está jugando una partida.

- a) Al mover el ratón por el sudoku, solo cuando el cursor del ratón esté encima de una de las celdas en las que el usuario puede poner un número (son los huecos iniciales del sudoku), habrá que hacer lo siguiente:
 - i) (0,25 puntos) El cursor del ratón cambiará a 
 - ii) (0,5 puntos) El color de fondo de la celda se destacará con un color diferente al resto de celdas.
- b) Al hacer clic sobre una celda con la que el usuario puede interactuar. Estas celdas son las que en el sudoku inicial están vacías. Habrá que hacer lo siguiente:
 - i) (0,75 puntos) Se destacarán (usando un color de fondo distinto al resto de celdas) todas las celdas en las que el usuario puede introducir números, correspondientes a la fila, columna y cuadro al que pertenece la celda en la que se ha hecho clic.
 - ii) (0,5 puntos) La celda pasará a estar *seleccionada*. Para ello su color de fondo será distinto al de las celdas de su fila, columna y cuadro (se recomienda utilizar el mismo color que se utilizó en el apartado 7.a.ii). Su borde también se destacará en color rojo.
 - iii) (0,25 puntos) Aparecerá una sección de números disponibles en la que se mostrará el título “Números disponibles” y los números del 1 hasta el tamaño del sudoku, para que el usuario pueda elegir el que quiera colocar en la celda seleccionada.
 - 1) (0,25 puntos) Al pasar el ratón por encima de cada uno de los números disponibles, éste se destacará de los demás mediante CSS y el cursor del ratón pasará a mostrarse como se indica en el apartado 7.a.i.
 - 2) (0,5 puntos) Al pinchar sobre uno de los números disponibles, este se colocará en la celda destacada y el sudoku se repintará sin que aparezcan ya la celda, fila, columna y cuadro destacados. También desaparecerá la sección de números disponibles.
 - 3) (0,75 punto) Tras colocar el número seleccionado por el usuario en el sudoku, se deberá comprobar si se han rellenado todos los huecos del sudoku. Si todos los huecos del sudoku están rellenos, se debe comprobar si se ha completado correctamente. Para ello se hará la misma petición al servidor que la utilizada para comprobar errores. Una vez nos responda el servidor:
 - Si el número de errores es 0, el sudoku se ha completado correctamente. Entonces, se detendrá el contador de tiempo y se sacará un mensaje emergente al usuario similar a: “¡¡¡ENHORABUENA!!! Has completado el sudoku correctamente en un tiempo de hh:mm:ss”, siendo hh:mm:ss el tiempo mostrado en el contador de tiempo del juego. Este

mensaje tendrá un botón de aceptar que, al pincharlo, cerrará el mensaje emergente y reiniciará el juego, haciendo lo mismo que en el apartado 6.d.

- Si hay errores, no se detendrá el contador de tiempo y se sacará un mensaje emergente al usuario con un texto similar a: "HAY XX errores. ¿Quieres intentar corregirlos?". El mensaje emergente tendrá dos botones: "SÍ" y "NO". Si se pulsa el botón de "SÍ", se cerrará el mensaje emergente y el juego seguirá. Si se pulsa el botón "NO", se detendrá el contador de tiempo, se cerrará el mensaje emergente y se reiniciará el juego, haciendo lo mismo que en el apartado 6.d.

Entrega

- El plazo de entrega de la práctica finalizará el **domingo 24 de mayo de 2020, a las 23:59h.**
- La entrega se realizará a través de la plataforma Moodle mediante la opción habilitada para ello y consistirá en un único fichero comprimido que contendrá todos los ficheros de la práctica para su correcto funcionamiento. Se recomienda comprimir la carpeta completa de la práctica.

Peticiones al servidor *RESTful* de la práctica 3

ERROR

Para todas las peticiones, si se produce un error se devuelve el siguiente texto en formato JSON:

```
{"RESULTADO":"ERROR","CODIGO":código del error, "DESCRIPCION":Descripción del error}
```

Nota: Algunas de las peticiones necesitan que se les pase la cabecera "Authorization" cuyo valor es el token que se recibe al hacer la petición para generar un sudoku. Por ejemplo:

```
"Authorization":"cbacffb7067d943cd925fa5bb0..."
```

Peticiones GET

No hay

Peticiones POST

- **RECURSO:** [api/sudoku](#)

- **Generar un nuevo sudoku:** [api/sudoku/generar/{TAMAÑO}](#)

Parámetros de la petición: Ninguno

Respuesta:

- Si se ha podido realizar correctamente la petición:

```
{ "RESULTADO": "OK", "CODIGO": 200, "DESCRIPCION": "Juego creado correctamente", "ID": 176, "TOKEN": "68d1ef3fdeea79770eb2ce27ee701a25f01e7e740d5c22fe601dfb2dec8b9a469ac08c0a1bce2dfcbbe4966adf206c2acecb74284ebadab4870d2031a7c2302", "SUDOKU": [[0,3,0,1],[0,0,0,3],[0,2,3,4],[3,4,0,0]] }
```

Importante:

- Los números devueltos van de 0 a $\{TAMAÑO\} - 1$
 - El token que devuelve se debe utilizar en el resto de peticiones al servidor que se hagan para el sudoku creado.
- **Comprobar errores en sudoku:** [api/sudoku/{ID_SUDOKU}/comprobar](#)

$\{ID_SUDOKU\}$ es el ID del sudoku que se quiere comprobar.

Es necesario enviar la cabecera "Authorization": " $\{TOKEN\}$ "

Parámetros de la petición:

- **juego:** array (con el formato de devuelto en SUDOKU en la petición de creación) del sudoku de la partida, incluidas las jugadas del usuario.

Respuesta:

- Si se ha podido realizar correctamente la petición:

```
{ "RESULTADO": "OK", "CODIGO": 200, "DESCRIPCION": "Lista de errores", "ID": "175", "FALLOS": [{"fila": 0, "columna": 1}, {"fila": 2, "columna": 0}] }
```

Importante:

- Los valores de fila y columna van de 0 a $\{TAMAÑO\} - 1$
- Si no hay fallos, el vector que viene en FALLOS tendrá 0 elementos.

Peticiones DELETE

- **RECURSO:** [api/sudoku](#)

- **Elimina un sudoku de la base de datos:** [api/sudoku/{ID_SUDOKU}](#)

`{ID_SUDOKU}` es el ID del sudoku que se quiere eliminar.

Es necesario enviar la cabecera "Authorization": "{TOKEN}"

Parámetros de la petición: Ninguno

Respuesta:

- Si se ha podido realizar correctamente la petición:
`{"RESULTADO": "OK", "CODIGO": 200, "DESCRIPCION": "Operaci\u00f3n realizada correctamente."}`