

# Informe del proyecto

---

El proyecto esta separado en 3 archivos:

- index.html contiene el esqueleto principal y la estructura de la pagina. Aqui se incluyen los elementos HTML en el orden que se pretenden mostrar.
- style.css contiene los elementos de estilos de algunas partes HTML.
- script.js contiene el codigo del script que ejecuta la pagina. Aqui esta la logica del juego, el control de errores, etc. Es el archivo mas importante de los tres.

## Estructura de la pagina

---

Inicialmente se muestra una seccion en la que se introducen datos de usuario. Esta seccion aparece inicialmente y esta definida en un nodo div para poder ocultarla despues. Se muestra un formulario con el input de texto y el boton de validacion, que valida el nombre introducido y activa la siguiente seccion. Al activar esa seccion aparece un mensaje de saludo y se activa el boton que comienza la partida.

Cuando se comienza la partida se oculta el div con los elementos anteriores y se muestra otro div que contiene la parte relacionada con el juego. Para ocultar o mostrar se ha utilizado la propiedad display en css. Tambien en este punto se genera la tabla agregando filas y columnas a la tabla desde una funcion generarTablero en Javascript.

## Logica del juego

---

El juego se divide en turnos, que se activan al pulsar el boton del dado. Este boton tiene un listener asociado que llama a la funcion tirarDado. Esta funcion genera un numero aleatorio para el dado y marca las casillas a las que se puede llegar y muestra el numero obtenido sobre la imagen del boton.

Para marcar estas casillas primero se comprueba si al avanzar en alguna de las direcciones (norte, sur, este, oeste) se exceden los limites del tablero. Si no es asi se resalta la casilla correspondiente cambiando su imagen a una version coloreada. Tambien se agrega un event listener a esas casillas que se asocia con la funcion moverHeroe al hacer click. Al hacer click sobre alguna de estas casillas se ejecuta dicha funcion, que mueve el heroe a esa casilla y a continuacion quita el resalte rojo y los event listeners.

Tambien es posible volver a activar el dado mediante el boton sin haber movido el heroe. Esto tambien implica eliminar el resalte al igual que antes.

Este proceso continua hasta que el heroe llega a la casilla donde esta el tesoro. En la funcion moverHeroe se comprueba cada vez si las coordenadas de la casilla destino son las mismas que las de la casilla del tesoro. En ese caso se muestra el mensaje correspondiente de victoria, se comprueba el record actual en local storage y se actualiza si se ha batido.

## Desafios

---

- Cubrir todos los casos especiales que surgen en el juego y hacer que la logica sea correcta en todos los casos ha llevado bastante tiempo. He tenido que hacer bastante depuracion desde el modo desarrollador en el navegador para entender por que el juego no estaba funcionando bien.
- Eliminar el resalte y los listeners de las casillas resaltadas individualmente complicaria el proceso ya que habria que guardarlas y actualizarlas. Por eso he decidido reestablecer las imagenes y quitar los listeners de todas las casillas del tablero.
- El manejo de algunos elementos HTML y CSS ha requerido bastante ensayo y error para lograr que se comportaran adecuadamente. Por ejemplo, para mostrar el dado como imagen en un boton y mostrar el texto con el valor de la tirada por encima. O para resaltar en rojo las casillas a las que puede moverse el heroe (al final he optado por reemplazar la imagen con una version coloreada).
- Al organizar las variables y el codigo he utilizado objetos JSON para representar los elementos del juego: jugador, mapa y meta. En lugar de tener varias variables globales me ha parecido mejor idea agrupar las que estan relacionadas en objetos JSON.