

Título: Documentación del Proyecto – Aventura del Héroe y el Cofre

En este proyecto he desarrollado un **pequeño juego** en JavaScript donde un héroe debe llegar a un cofre dentro de un tablero de 10x10 casillas. El **objetivo principal** era practicar la creación **de elementos del DOM** con JavaScript, el uso de **eventos**, la manipulación de **arrays con bucles** y el **almacenamiento** de información en **localStorage**.

Primero, el usuario introduce su nombre en un formulario. Ese **formulario** se crea completamente desde JavaScript: el **label**, el **input** de texto y los **botones**. Al pulsar el botón “**Introducir nombre**” se validan los datos. Compruebo que el nombre tenga al menos 4 caracteres y que no contenga números. Si algo no es correcto, se **muestra un alert** con el motivo. Cuando el nombre es válido, se actualiza un párrafo con un mensaje tipo “**A luchar héroe: NOMBRE**” y se habilita el botón “Jugar”.

Al pulsar “**Jugar**” se genera dinámicamente una **tabla de 10 filas por 10 columnas**. En cada celda se coloca una imagen: la mayoría son imágenes de suelo, en la posición [0,0] aparece la imagen del héroe y en la posición [9,9] la imagen del cofre. En este momento se oculta el botón “Jugar” y aparece el botón del dado “**Tirar Dado**”, que incluye dentro una imagen de la cara del dado. De esta forma el tablero y la interfaz se montan de forma dinámica solo cuando el jugador ya ha introducido un nombre válido.

Cada vez que el jugador pulsa el botón del dado se genera un **número aleatorio** entre 1 y 6. Según el número que salga, se actualiza la imagen del dado. A continuación, el programa calcula qué **celdas** son **alcanzables** desde la posición actual del héroe en las **cuatro direcciones** (arriba, abajo, izquierda y derecha) sin salirse del tablero. Esas casillas posibles se **marcan** cambiando el **borde de la celda a rojo**, para que el usuario vea claramente **dónde puede moverse**. El resto de las celdas quedan con el borde normal.

El tablero tiene un **evento de clic** que permite mover al héroe. Solo se acepta el clic si la celda tiene el borde rojo. En ese caso, se **cambia la imagen** de la celda anterior del **héroe por el suelo** y se pone la imagen del **héroe en la nueva**

posición. También se limpia el borde de todas las celdas para que, en la siguiente tirada, solo aparezcan marcadas las nuevas posibilidades de movimiento. De esta manera **controlo los movimientos válidos y evito movimientos fuera de las casillas** permitidas por el dado.

Durante la partida, se lleva un **contador del número de tiradas**. Cada vez que se pulsa el botón del dado se incrementa la variable **numeroTiradas** y se muestra el valor actualizado en un párrafo de la página. Cuando el héroe llega a la **casilla del cofre** (fila 9, columna 9), el juego muestra un **mensaje con el número de tiradas usadas** y se comprueba si ese resultado mejora el récord anterior.

Para almacenar el récord he utilizado **localStorage**. Guardo el mejor número de tiradas con la clave "**recordTiradas**". Al ganar una partida, si todavía no hay récord, se guarda la puntuación actual como primer récord. Si ya existe, solo lo actualizo cuando el número de tiradas de la nueva partida es menor que el récord guardado. Además, hay un botón "**Logro de Récord**" que permite **consultar** en cualquier momento cuál es el **mejor resultado almacenado**. De esta forma, aunque se recargue la página, el récord se mantiene mientras no se borren los datos del navegador.

En cuanto al **diseño**, he utilizado una estética inspirada en aventuras tipo exploración (**similar a Lara Croft**), con un fondo oscuro de selva y tonos marrones y dorados en el formulario y el tablero. El objetivo es que visualmente **recuerde a una expedición en busca de un tesoro**, pero sin complicar demasiado el CSS. El formulario del nombre está en un recuadro que destaca sobre el fondo y el tablero de juego tiene un marco que simula una especie de reliquia antigua.

Entre las **principales dificultades** del proyecto destacaría la **lógica de los movimientos del héroe** y los **límites del tablero** para que el héroe no se salga de la tabla. Otra parte que me costó al principio fue el uso de **localStorage**, pero finalmente he conseguido guardar y recuperar el récord sin problemas.

En general, este proyecto me ha servido para entender mejor cómo **manejar el DOM**, trabajar con **eventos**, usar **bucles y condiciones**, y a empezar a usar **localStorage** para guardar datos. También me ha permitido unir toda la parte de lógica del juego con un diseño más cuidado usando CSS.