**Segundo Proyecto Práctico.**

**1ª Evaluación:**

**Construcción de un Juego de Tablero. Caza del Tesoro.**

**ÍNDICE**

[**Descripción General** 3](#_Toc182313299)

[**Especificación de Requisitos** 3](#_Toc182313300)

[**1.** **Requisitos Funcionales:** 3](#_Toc182313301)

[**2.** **Requisitos de Interfaz de Usuario:** 4](#_Toc182313302)

[**3.** **Requisitos de Sistema:** 4](#_Toc182313303)

[**4.** **Requisitos de Seguridad:** 4](#_Toc182313304)

[**Análisis.** 4](#_Toc182313305)

[**Diagrama de Casos de Uso:** 4](#_Toc182313306)

[**Diseño.** 5](#_Toc182313307)

[**Diseño de la Funcionalidad:** 5](#_Toc182313308)

[**Entrega Final** 6](#_Toc182313309)

# **Descripción General.**

El objetivo de este proyecto es desarrollar una aventura interactiva en la cual el usuario controla a un personaje que se mueve por una cuadrícula de 10x10 en busca de un cofre, empleando técnicas de programación en JavaScript y HTML para gestionar el formulario, los movimientos en la tabla y el almacenamiento de datos en localStorage. El proyecto evalúa los criterios de evaluación de los R.A.5 y R.A.6, con una puntuación máxima de 10,00.

**Nota Importante:** Los requisitos, análisis y diseño especificados en este documento deben cumplirse al 100%. Si el programador considera necesario añadir mejoras para optimizar o ampliar la usabilidad de la aplicación, estas podrán implementarse, siempre que se justifiquen por escrito y se presenten al cliente (profesor) para su aprobación.

# **Especificación de Requisitos.**

## **Requisitos Funcionales:**

* + **Formulario de Nombre**: La aplicación debe contener un campo de formulario para que el usuario introduzca su nombre, seguido de un botón "Introducir nombre".
    - Si el nombre tiene menos de 4 letras, mostrará el mensaje “El nombre debe tener 4 o más letras”.
    - Si contiene números, el mensaje será “Números no permitidos”.
    - Si el nombre es válido, se modificará el contenido de un elemento HTML para mostrar “A luchar héroe: [nombre]” y se habilitará un botón “Jugar”.
  + **Generación de la Aventura**:
    - El botón "Jugar", inicialmente deshabilitado, activará una cuadrícula de 10x10 con imágenes que representen suelo.
    - La celda en la esquina superior izquierda contendrá al héroe y la esquina inferior derecha el cofre.
    - Al iniciar, el botón "Jugar" desaparece y aparece un botón "Tirar dado" con una imagen de un dado.
  + **Movimiento del Héroe**:
    - Al pulsar "Tirar dado", se genera un número aleatorio entre 1 y 6 que aparece en la imagen del dado, permitiendo que el héroe se mueva un número de celdas igual al número mostrado en el dado.
    - Se resaltarán en rojo las celdas a las que el héroe puede moverse (horizontal o verticalmente) y, al hacer clic en una de estas, el héroe se desplaza, dejando su posición anterior en blanco.
  + **Victoria y Récord de Tiradas**:
    - Cuando el héroe alcanza el cofre, se muestra un mensaje de victoria que indica el número de tiradas usadas.
    - Si existe un récord anterior en localStorage, se compara con las tiradas actuales.
      * Si el récord no es superado, muestra el mensaje “Récord no superado, el actual récord es X”.
      * Si se establece un nuevo récord, se actualiza localStorage con el nuevo valor y se notifica al usuario.

## **Requisitos de Interfaz de Usuario:**

* + **Interfaz Interactiva**:
    - El formulario inicial solicita el nombre del jugador, controlando la longitud y el uso de caracteres no permitidos.
    - La cuadrícula visual debe ser una tabla HTML 10x10 que incluya imágenes que representen el suelo, el héroe y el cofre.
    - Un dado muestra el número aleatorio en una imagen, y las posibles celdas de movimiento se resaltan en rojo.
  + **Feedback Visual**:
    - Mensajes de victoria o error se muestran con alert() u otro elemento que consideres más atractivo visualmente.
    - La tabla de juego y el dado se actualizan visualmente tras cada movimiento o lanzamiento del dado.

## **Requisitos de Sistema:**

* + **Entorno de Ejecución**: La aplicación debe funcionar en navegadores modernos (Chrome, Firefox, Edge, Safari) sin necesidad de conexión a un servidor.
  + **Almacenamiento Local**: La aplicación usa localStorage para almacenar y recuperar el récord de tiradas.

## **Requisitos de Seguridad:**

* + **Validación de Inputs**: El campo de nombre debe validar la longitud y los caracteres permitidos antes de iniciar el juego.
  + **Manejo de Errores**: La aplicación debe manejar adecuadamente errores de entrada y casos de borde (por ejemplo, intentos de movimiento no válidos).

# **Análisis.**

## **Diagrama de Casos de Uso:**

* **Actores**:
  + **Usuario**: Interactúa con el formulario y realiza movimientos en el juego.
* **Casos de Uso**:
  + **Introducir nombre**: El usuario ingresa su nombre y activa el juego.
  + **Jugar**: El usuario mueve al héroe en la tabla de juego usando un dado y alcanzando el cofre.
  + **Tirar Dado**: Cada tirada permite un movimiento y suma una tirada al contador.
  + **Logro de Récord**: El juego compara las tiradas del jugador con el récord actual en localStorage.

# **Diseño.**

## **Diseño de la Funcionalidad:**

* 1. **Flujo de Trabajo de la Aventura**:
     + El usuario ingresa su nombre.
     + Selecciona "Jugar" para iniciar la aventura.
     + Al pulsar "Tirar dado", se habilitan los movimientos posibles, y el usuario elige una celda.
     + Si el héroe alcanza el cofre, se muestra el mensaje de victoria y se actualiza el récord si es superado.
  2. **Estructura del Código**:
     + **Funciones Principales**:
       - iniciarJuego(): Función principal que valida el nombre y habilita el botón "Jugar".
       - generarTablero(): Genera la tabla 10x10 e inicia la posición del héroe y el cofre.
       - tirarDado(): Lanza un dado y determina las posibles celdas de movimiento.
       - moverHeroe(): Permite al héroe moverse en la cuadrícula y actualiza su posición.
       - validarNombre(): Valida la longitud y el contenido del nombre ingresado.
  3. **Estructura de Datos**:
     + **Variables Principales**:
       - nombre: Almacena el nombre del usuario.
       - posicionHeroe: Almacena la posición actual del héroe en la cuadrícula.
       - numeroTiradas: Lleva el conteo de las tiradas.
     + **Almacenamiento en localStorage**:
       - recordTiradas: Guarda el récord de tiradas en localStorage para comparar y actualizar al final del juego.

# **Entrega Final.**

El proyecto completo deberá entregarse como un archivo comprimido (*.zip/*.rar) que incluya todos los archivos necesarios para ejecutar la aplicación. Para la entrega, será necesario proporcionar el enlace al repositorio de GitHub del estudiante en Moodle.

El repositorio deberá incluir al menos los siguientes elementos:

1. **Código Funcional**:
   * Archivos .html, .css, y .js que componen la aplicación.
2. **Informe de Casos de Uso**:
   * Un diagrama que ilustre los principales casos de uso de la aplicación.
3. **Documentación del Proyecto**:
   * Documento breve explicando las decisiones de diseño y los desafíos encontrados.
4. **Archivo README.md**:
   * Un archivo README que detalle la funcionalidad de la aplicación y los pasos para ejecutar el proyecto.