**Juego de Super Mario Bros en el navegador**

Santiago Sánchez Grisales

*Instituto Tecnológico Metropolitano de Medellín*

*Medellín, Colombia*

santiagosanchez309942@correo.itm.edu.co

**Abstract - Super Mario Bros, fue el juego de Nintendo que marcos récords por su éxito, en este proyecto que recreara una parte del juego utilizando HTML, JavaScript y CSS como principales lenguajes para el código fuente, también se usaran otras herramientas, entre ellas MDL, una biblioteca desarrollada por la empresa Google la cual permite manejas graficas en una página en el navegador; de esta biblioteca se usa principalmente la etiqueta <canvas> ya que es la base. Además, se usa el editor de código visual studio code, un software creado por Microsoft para editar código de diferentes lenguajes, a este software se instala extensiones para mejorar el rendimiento de desarrollo.**

# **I. DESCRIPCIÓN DEL JUEGO SUPER MARIO BROS**

## ***A. Breve historia Super Mario Bros***

Es un video juego diseñado por Shigeru Miyamoto, lanzado el 13 de septiembre de 1985 y producido por la compañía Nintendo.

Este juego termino siendo uno de los más exitosos en la historia (más de 10 millones de cartuchos vendidos) en 1999 fue reconocido como uno de los videojuegos más vendidos de todos los tiempos, fue tal su impacto, que termino siendo uno de los iconos principales de Nintendo, y uno de los personajes más reconocidos de los videojuegos.

## ***B. Descripción del juego tomada del manual del videojuego:***

La siguiente descripción es la que tenía los manuales que entregaba Nintendo con el videojuego:

*Un día, el reino de la pacífica gente hongo fue invadido por Koopa, una tribu de tortugas famosas por su magia oscura. La silenciosa y pacífica gente hongo se convirtieron en simples piedras, ladrillos e incluso en plantas de pelo de caballo, y el Reino Champiñón cayó en la ruina.*

*El único que puede deshacer el hechizo mágico en Mushroom People y devolverlos a su estado normal es la Princesa Toadstool, la hija del Rey Champiñón. Desafortunadamente, ella está actualmente en manos del gran rey de las tortugas Koopa.*

*Mario, el héroe de esta historia (tal vez) escucha acerca de la difícil situación de la gente hongo y se embarca en una misión para liberar a la Princesa Toadstool del malvado Koopa y restaurar el reino caído de la gente hongo. ¡Eres Mario! ¡Depende de ti salvar a la gente hongo de la magia oscura de Koopa!*

## ***C. Jugabilidad***

En sus tiempos su principal característica era su desplazamiento lateral, el personaje principal (Mario) debe enfrentar a Bowser y rescatar a la Princesa Peach. Sin embargo, para llegar a su objetivo tiene que avanzar por varios mundos (cada mundo con 4 niveles), en total son 8 mundos.

En cada uno de estos niveles hay un tiempo límite marcado en la pantalla.

En cada mundo, en su último nivel se encuentra un castillo Bowser, pero la princesa se encuentra en el último de estos.

## ***D. Niveles***

En la carpeta actual del informe esta la carpeta “Niveles” en la cual hay imágenes con el diseño de los niveles y su respectivo contenido.

***E. Música***

Los audios reproducidos en el juego fueron compuestos por Kōji Kondō, reproducidos en un sintetizador, los audios están en la carpeta del presente informe llamada “Musica”, los audios ya están en formato wav obtenidos del juego original de la versión estadounidense.

# **II. DESARROLLO**

## ***A. Lenguaje y biblioteca de desarrollo***

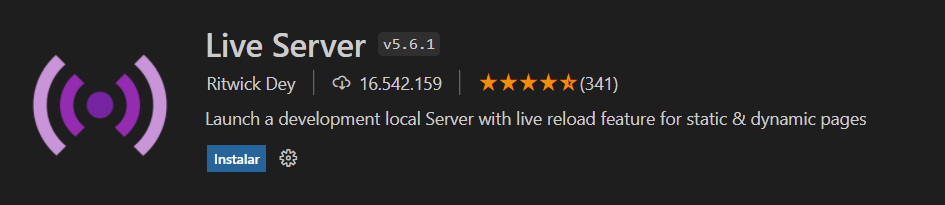
El juego se ejecutará en el navegador, para ello se usará HTML, CSS y JavaScript para la gran parte del código fuente, para el motor del juego se va a usar la biblioteca CSS MDL (Material Design Lite).

**Versión utilizada en este proyecto:** V1.3.0

## ***B. Software de desarrollo***

Para ser prácticos, lo ideal es un editor de código que permita trabajar en cualquiera de los lenguajes a trabajar, uno de los más conocidos es visual studio code de la empresa Microsoft.

Para el trabajo actual no influye la versión que se use, no obstante, es útil instalar la extensión live server, ya que este permite crear un servidor virtual para la página y carga de forma automática los cambio en el código fuente, además de esta, el editor recomienda otras extensiones de acuerdo al lenguaje que se esté trabajando, esas extensiones se recomiendan instalar para que el editor pueda procesar y/o ofrecer ayuda al programador.



Captura de pantalla de la extensión Live Server (Autoría propia)

La primera vez ejecutando el código con la extensión instalada, saldrá una alerta de seguridad de Windows, esto es para proteger del equipo; se escoge la opción permitir acceso ya que el código no es malicioso.

Para probar el código, en este caso requerimos de un intérprete, en este caso contamos con la ventaja que los navegadores web ya tienen uno incluido. Debido a que cada navegador tiene algunas variaciones, establezco Microsoft Edge de Microsoft como predeterminado para ejecutar el código.

# **III. DESARROLLO DEL CÓDIGO FUENTE**

Una vez configurado visual studio code, entramos a la página oficial (https://getmdl.io/) de Material Design lite y en la pestaña getting started están las instrucciones para descargar y/o utilizar la biblioteca.

Si fue descargada se copia la carpeta en el directorio raíz de nuestro código y si la intención es usar los recursos desde internet, lo único es copiar las líneas de código que da la página para importar los recursos en el archivo HTML (Para este proyecto, se traen todos los recueros desde internet)

## ***A. Documentación empleada***

Para el desarrollo del proyecto las principales fuentes de documentación fue la página oficial de MDL y (<https://developer.mozilla.org/es/docs/Web/API/Canvas_API>), esta última, son reconocidas por ser de las mejores páginas de documentación para el desarrollo de software, además en muchos casos trae ejemplos con la posibilidad de editar el código directamente para hacer pruebas.

Entre las dos páginas están las instrucciones necesarias para crear los módulos, los componentes y las configuraciones de la página, sin embargo, conociendo los lenguajes en que está basada (HTML, CSS y JavaScript) es posible realizar personalizaciones más complejas.

**Versión del navegador Microsoft Edge en la que se desarrolló el proyecto:** 96.0.1054.29

## ***B. Colisiones***

A través de cálculos matemáticos, se podrá determinar si un objeto colisiona con otro cuando se desplaza sobre el canvas.

Se determinará la posición en donde se encuentra el objeto y hacia donde se deberá desplazar (hacia los lados o hacia arriba/abajo) para determinar si hay una colisión a la derecha/izquierda o arriba/abajo del objeto que se está desplazando y del objeto con el cual puede colisionar.

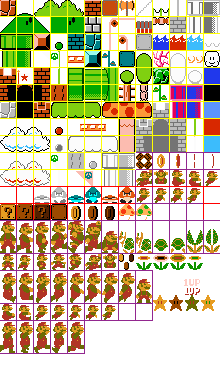
Ocurre una intersección cuando un objeto está encima de otro. Se determinará mediante el área del objeto, para saber si algún punto prioritario de un objeto está encima de otro.

Canvas permite crear animaciones cuadro por cuadro: se dibuja un cuadro, a continuación, se dibuja otro y así hasta sucesivamente. Al mostrar sucesivamente esos cuadros, se crea la ilusión de animación.

## 

## ***C. Diseños***

Los diseños fueron tomados del usuario de GitHub redxdev, su URL es: <https://github.com/redxdev>



Diseños del juego. Tomada del usuario redxdev

# **Bibliografía**

Docs, M. W. (22 de 11 de 2021). *API Canvas*. Obtenido de https://developer.mozilla.org/es/docs/Web/API/Canvas\_API

Google. (2014). *Material design lite*. Obtenido de https://getmdl.io/

*Pagina Fandom Super Mario Bros*. (s.f.). Obtenido de https://mario.fandom.com/es/wiki/Super\_Mario\_Bros.

*Poki*. (s.f.). Obtenido de https://poki.com/

*Super Mario Bros 1 NES*. (12 de 05 de 2011). Obtenido de https://oldschoolgamer.foroactivo.com/t116-super-mario-bros-1-nes