Inicio del proyecto

En este curso vamos a estar desarrollando una aplicación llamada: Platzi Video. En toda plataforma de video hay un componente especial en el desarrollo, tenemos que saber implementar el MediPlayer, en este curso vamos a estar desarrollando este feature de forma modular, esto quiere decir que vamos a desarrollar plugins que vamos a implementar a nuestro reproductor, extendiéndole sus funcionalidades. Vamos a comenzar con un poco de CSS y HTML ya escrito.

Primer paso

Crearemos nuestros primeros archivos usando npm init -y, donde -y es una bandera que le dicta a npm que le diga sí a todas las preguntas que haga.

npm init -y

Esto nos creará un archivo package.json que lo sustituiremos por el siguiente:

{

"name": "platzi-media-player",

"version": "1.0.0",

"description": "Proyecto del Curso Profesional de JavaScript de la Escuela de JavaScript de Platzi.",

"license": "MIT",

"author": "César Augusto Barco <augustopayza@gmail.com>",

"keywords": [

"platzi"

],

"scripts": {

"start": "live-server"

},

"devDependencies": {

"live-server": "^1.2.1"

}

}

Una vez tengamos todo esto listo vamos a proceder a instalar nuestro live-server para empezar a trabajar. Para instalar esto vamos a usar el siguiente comando npm i -D live-server donde i significa install y la bandera -D develoment, esto quiere decir que no lo vamos a usar en producción.

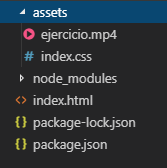
‌Una vez instalado ya lo podremos usar con el package.json que dejé arriba. Lo usaremos con el comando start que llamará a su vez a live-server.

Antes de ejecutar este vamos a implementar varios archivos. Estos serán los siguientes:

HTML

CSS

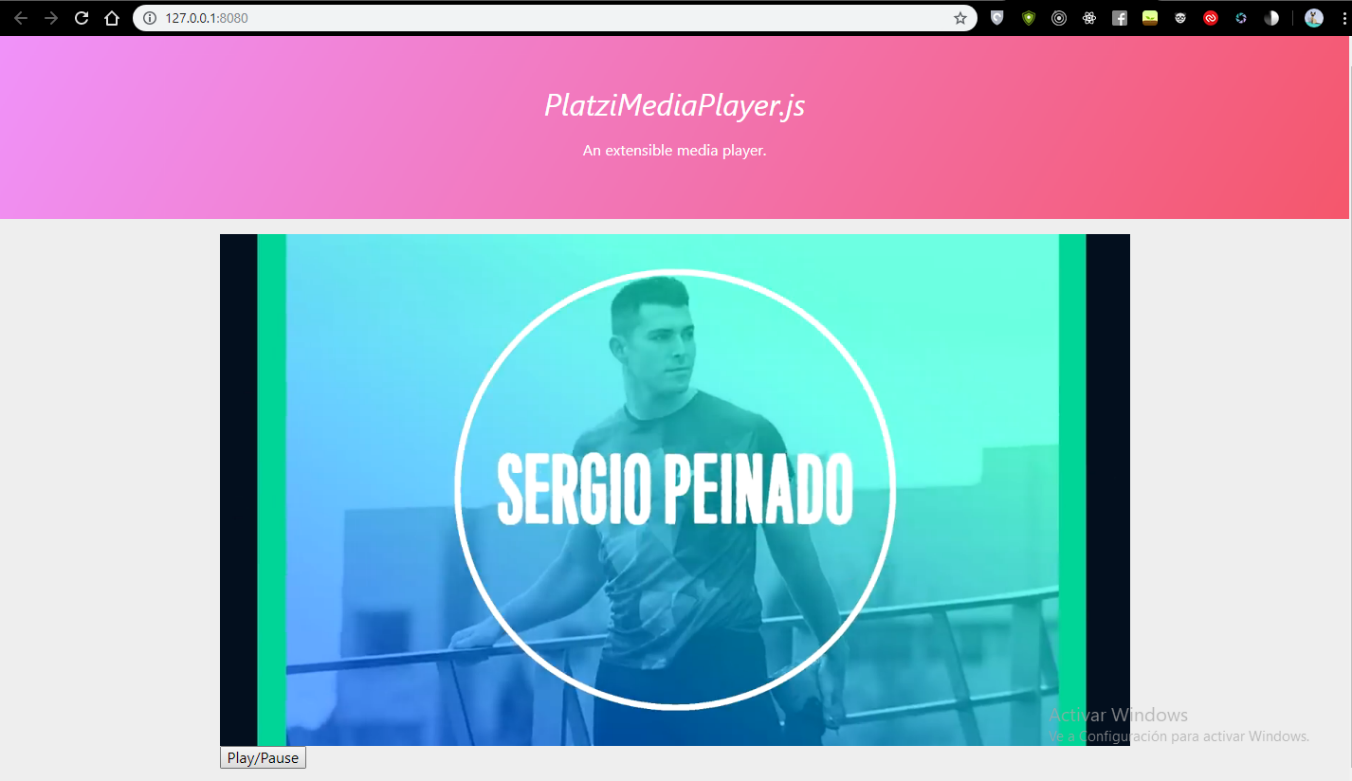
También cualquier video que tengamos en nuestra PC. Nuestras carpetas tienen que quedar de la siguiente forma:



Ahora sí vamos a ejecutar nuestro pequeña aplicación.

npm start

Nuestra pequeña aplicación andará en la IP que nos muestre la terminal.



¿Qué sigue? Tenemos un botón que no funciona, lo vamos a poner a funcionar con un media query. Abrimos nuestras etiquetas de script.

Tenemos un vídeo que debemos manipular, lo vamos a hacer con querySelector(""), a este tenemos que pasarlo un elemento, en este caso será video, es el único elemento video en nuestro HTML. Tambien debemos traer nuestro botón con `querySelector``.

**const** video = document.querySelector("video") **const** button = document.querySelector("button")

Cuando le demos click a nuestro botón queremos que el vídeo se reproduzca. Lo hacemos de la siguiente manera:

button.onclick = ()=> video.play()

El video.play() se saca de la API que trae el navegador, todos los elementos del DOM traen un API. Para saber cuales son las opciones de esta API podemos ir a MDN a ver toda la documentación. No podemos darle play de una vez a penas se entre en la página, esto pasa por que los navegadores tienen una seguridad que no permite que esto pase, solo se puede dar play si el usuario tiene la libertad de hacerlo.

Ahora nuestro código no es muy extensible, vamos a lograr esto usando prototipado. Para hacerlo extensible se pueden usar clases, pero en este caso usaremos protitype, usaremos el siguiente código para lograrlo.

const video = document.querySelector("video");

const button = document.querySelector("button");

function MediaPlayer(){}

MediaPlayer.prototype.play = function() {

video.play()

}

const player = new MediaPlayer()

button.onclick = () => player.play();

Explicación:

Creamos una función llamada mediaPlayer que nos servirá como prototipo.

A mediaPlayer le asignamos una función llamada play usando prototype. Esta función le dará inicio al video.

Luego con el botón se acciona una función llamada player que es una instancia del prototipo mediaPlayer que creamos. La instancia se crea usando la palabra new.

Hagámoslo más reutilizable

Para que nuestro código sea más reutilizable debemos hacerlo de esta manera:

const video = document.querySelector("video");

const button = document.querySelector("button");

function MediaPlayer(config) {

this.media = config.el;

}

MediaPlayer.prototype.play = function() {

this.media.play();

};

const player = new MediaPlayer({ el: video });

button.onclick = () => player.play();

Explicación:

A nuestra función madre o prototipo le pasamos una configuración. Esta configuración lo que va a tener es el elemento video original. Le asignamos a this.media el elemento video.

A la función extendida le asignamos play() a this.media para que se ejecute cuando presionemos el botón.

En nuestra función especial player es una instancia del prototipo le asignaremos el valor de video para que lo reciba en configuración. Esto lo haremos con destructuración de objetos.

Acá no podemos usar arrow function por que el valor de this es global. Más adelante se verá con más detalle.

Para agregarle la funcionalidad de pausa y play con el mismo botón, debemos condicionar la función play de MediaPlayer de la siguiente manera:

MediaPlayer.prototype.play = function() {

if(this.media.paused){

this.media.play();

} else {

this.media.pause()

}

// o podemos usar lo siguiente:

// this.media.paused ? this.media.play() : this.media.pause()

};