# Progetto informatica multimediale

## Francesco Cucchiaro 144019

Ho deciso di fare un progetto javascript con Three.js, una libreria (framework è il termine tecnico) che permette a javascript di interagire - tramite le API di webGL – con il browser e sfruttarlo come motore grafico 3D.

Le prestazioni di questo framework open source (qui la [repo](https://github.com/mrdoob/three.js/) github) sono eccezzionali, dato che la stessa immagine 3D viene vista allo stesso modo da un computer da gaming ma anche dal cellulare con android da 50 euro. E ricordo che tutto viene fatto all’interno di un browser web.

Mi sono focalizzato proprio su questa libreria perché ci ho già lavorato l’anno scorso, quindi non devo impararmela da zero, ma anche perché è uno strumento che non limita la creatività, anzi.

Il mio progetto è quindi un [launchpad](https://www.amazon.it/Launchpad/s?k=Launchpad) (una “tastiera” midi che viene usata principalmente da dj e producer che permette, alla pressione del tasto, di lanciare dei loop sonori). Visto che siamo in un mondo tridimensionale, ho associato il drag del mouse allo spostamento nello spazio dei bottoni.

E’ consigliabile testare il sito con un web server locale, o visitare il sito <launchpad.ccns.it>

## Funzionamento

Per lavorare con three.js sono necessarie queste cose:

* includere il file three.js nel file index (in alternativa può essere usato anche un cdn)
* creare un oggetto di tipo WebGLRenderer, che facendo un analogia è l’azione di dire al falegname di crearci un telaio per il nostro quadro (righe 39 – 42)
* creare una scena, tornando all’analogia di prima questa potrebbe essere la tela bianca (righe 45-46)
* inizializzare la camera, ovvero il punto di vista che verrà visualizzato dal browser sulla scena creata (righe 52-54)

opzionalmente

* inizializzazione dei controlli del mouse (righe 56–57)
* inizializzazione del suono globale (righe 60-61)

## Figure

Una volta inizializzato tutto ciò è possibile aggiungere, direttamente da javasctipt, delle figure basiche (cerchi, rettangoli, sfere, eccetera) oppure importare dei modelli esportati da altri programmi di lavoro 3D (es cinema 4d).

Per il posizionamento è simile a come si usa con i canvas, ovvero vengono usate delle coordinate (x,y,z). E’ possibile modificare di un oggetto anche la sua scala e rotazione (su qualsiasi asse).

Inoltre, per un utilizzo più avanzato, si può modificare il materiale di un oggetto, la sua refrattività, la texture e i poligoni che lo compongono

Per questo progetto ho deciso di creare le figure direttamente da javascript con un materiale di tipo Sprite, ovvero dei piani che guardano sempre la camera. Gli oggetti poi sono stati divisi in gruppi da due per una gestione migliore. (righe 65 – 116)

## Integrazione con la musica

Per eseguire il loop al click degli oggetti 3D è stato usato un modulo di Three.js chiamato raycaster, che viene invocato da un ascoltatore normalissimo di javascript. Raycaster non fa altro che restituire l’oggetto 3D che è cliccato; questo oggetto poi è composto da un ID univoco che permette quindi di essere associato a un loop specifico.

I loop in esecuzione sono quindi aggiunti e rimossi da un array a seconda del loro stato, e grazie a questo array è possibile effettuare dei controlli in modo da colorare gli oggetti 3d in un modo o un altro.

La musica poi viene riprodotta direttamente da three.js grazie all’inizializzatore di cui abbiamo parlato prima.

Per animare lo sfondo “a ritmo” sono stati usati due timer, uno (interval) viene eseguito ogni 931ms e chiama la funzione per colorare lo sfondo di viola, e all’interno di tale funzione è presente anche un altro timer (timeout) che dopo 300ms reimposta il colore dello sfondo di default