

Práctica de jQuery

Fecha de entrega: 20-dic-2018

El objetivo de esta práctica es desarrollar con JavaScript y jQuery un sencillo juego de memoria. El juego utiliza una baraja formada por parejas de cartas iguales empezando con todas ellas dispuestas boca abajo. En cada ciclo del juego, el jugador puede descubrir dos cartas. Si ambas cartas son iguales (forman una pareja) se eliminan de la partida, y si son distintas se vuelven a poner boca abajo. El juego termina cuando el jugador ha descubierto todas las parejas. En la figura 1 se muestra la situación inicial del juego para un nivel de dificultad **Fácil** y en la figura 2 se muestra el momento en el que el jugador a descubierto dos cartas que no forman pareja y que serán dispuestas boca abajo pasado un breve lapso de tiempo.

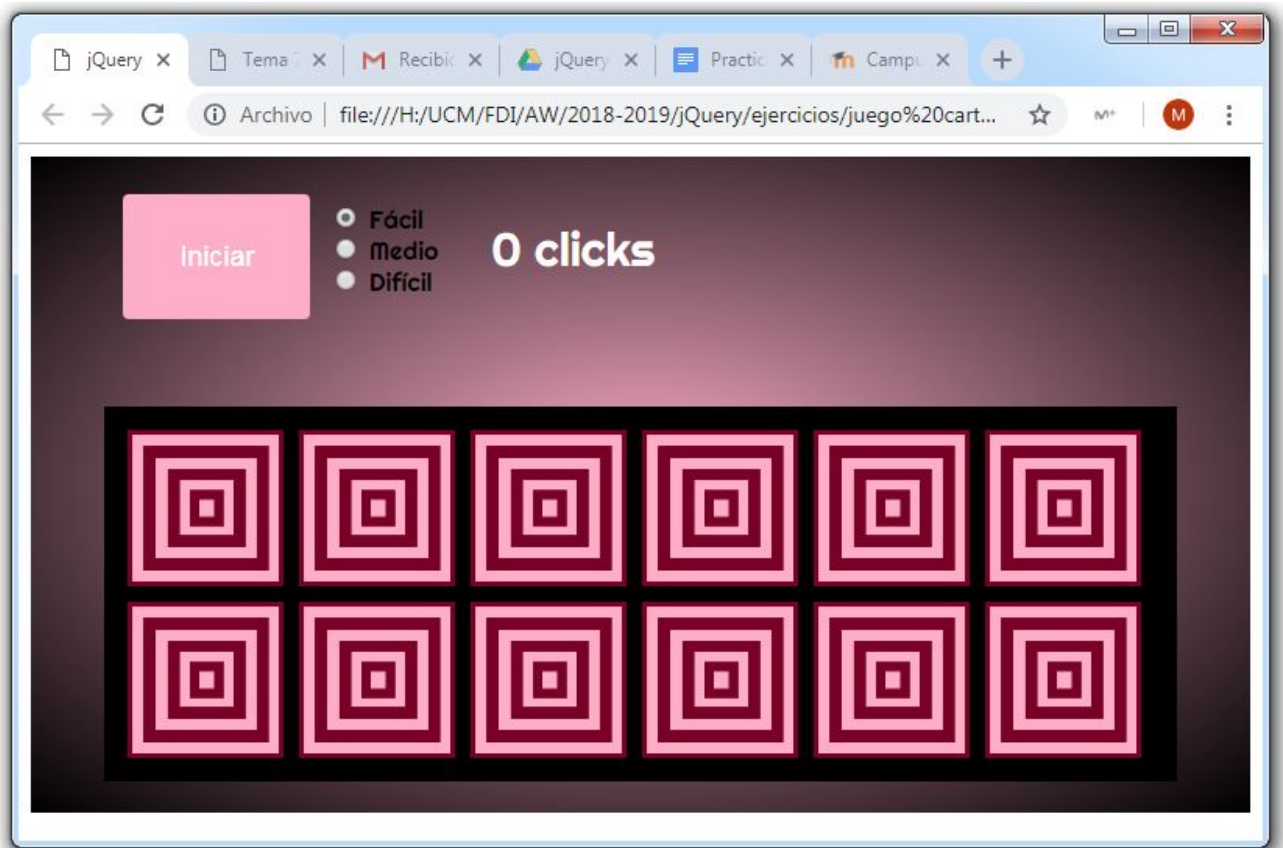


Figura 1

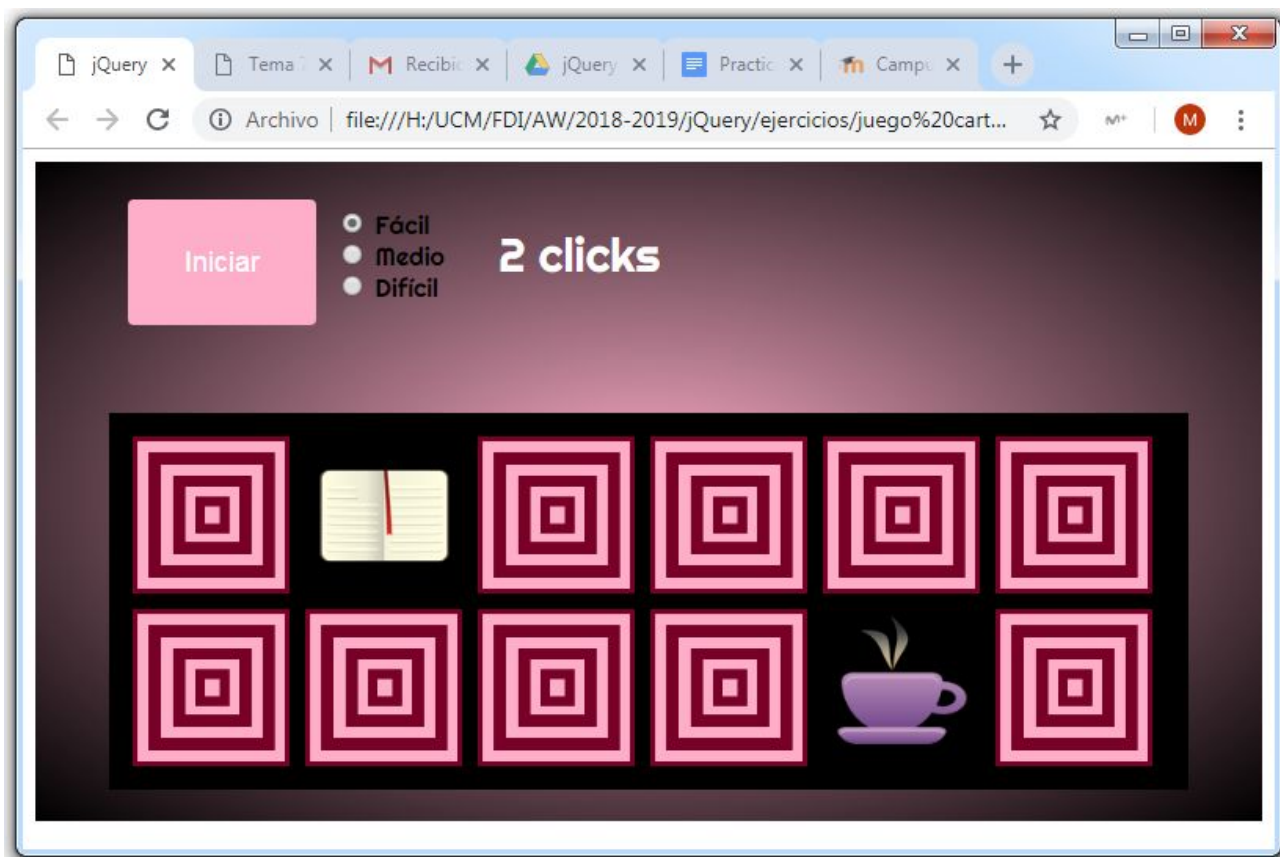


Figura 2

Cada vez que el jugador pulse el botón **Iniciar** se genera una nueva disposición de cartas. A medida que avanza el juego, se indica en la pantalla el número de clicks que el jugador ha pulsado hasta el momento (a mayor clicks menor memoria) y un resumen de las cartas que el jugador ha conseguido emparejar hasta el momento (ver figura 3).

El jugador puede seleccionar uno de los 3 niveles de dificultad del juego: **Fácil**, **Medio** y **Difícil**. La diferencia entre ellos es exclusivamente el número de cartas. En el nivel **Fácil** la baraja es de 12 cartas, en el **Medio** de 24 y en el **Difícil** de 36. En la figura 4 se muestra la situación inicial del juego cuando se ha seleccionado el nivel **Difícil**.

El diseño de la baraja y la disposición en pantalla de todos los elementos del juego que se muestra en este enunciado es solamente un ejemplo. Cada pareja de laboratorio deberá crear su propio diseño.

Los contenidos estudiados en clase de teoría acerca de JavaScript y jQuery son suficientes para desarrollar esta práctica. No obstante, los siguientes detalles pueden ser útiles para el desarrollo:

- La propiedad CSS **visibility: hidden** hace que un elemento se oculte pero siga ocupando espacio en la página.
- Para utilizar el método **setTimeout()** con una función que tenga parámetros, se debe envolver dicha función dentro de otra anónima sin parámetros que será la que aparezca en el método **setTimeout()**. El código quedaría de la siguiente manera (asumiendo un tiempo de 1000 ms):

```
setTimeout ( function() { .... llamada a la función con parámetros ... } , 1000 );
```

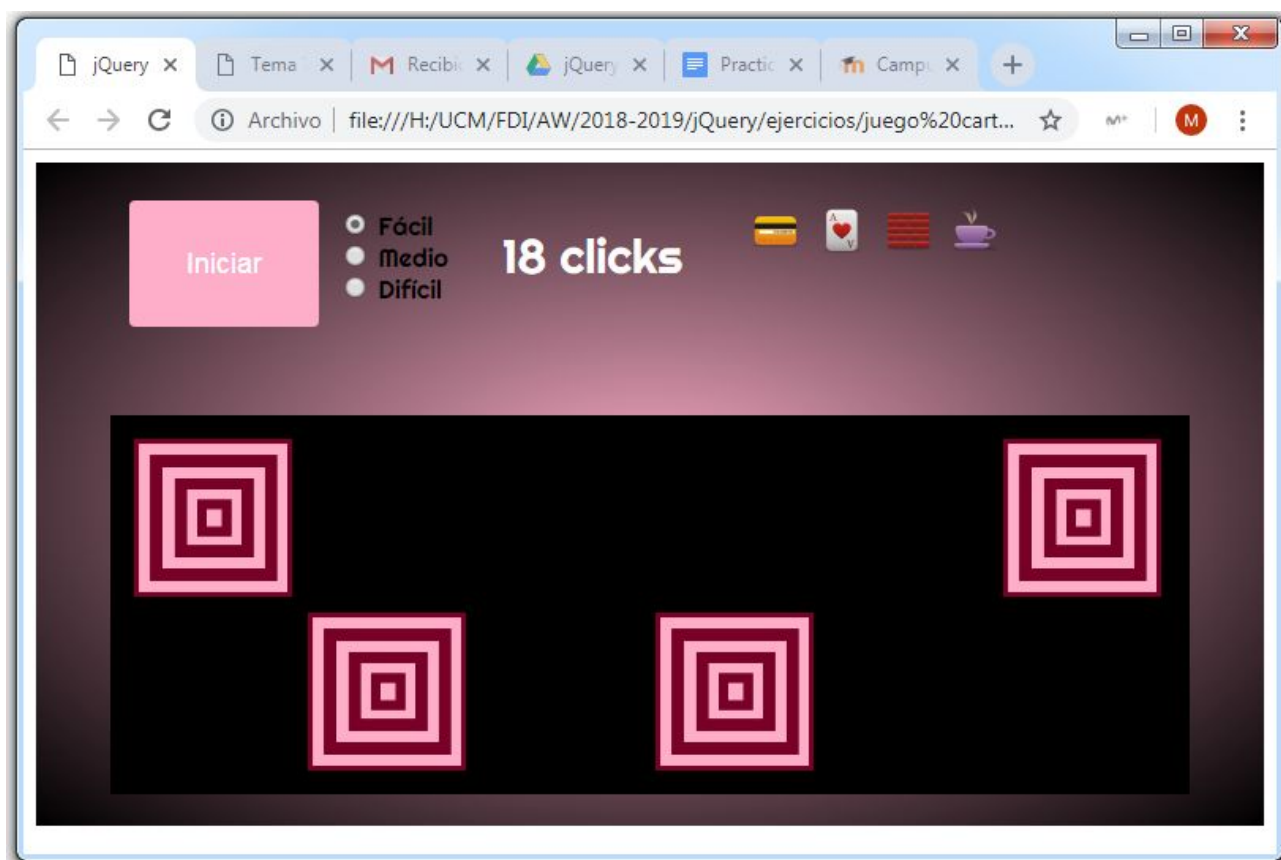


Figura 3

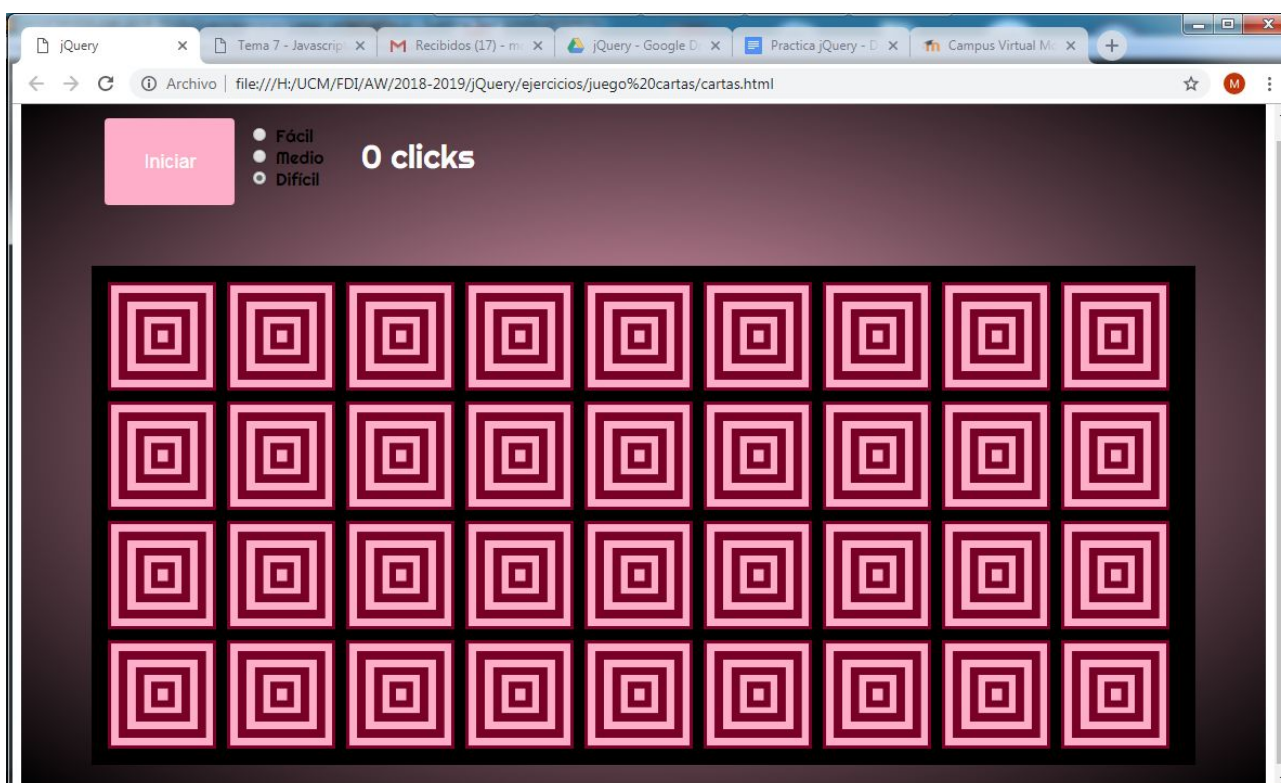


Figura 4

Entrega

La fecha límite para la entrega de esta práctica es el jueves 20 de diciembre de 2018 (16:00). Se entregará a través del CV en un fichero pxx.zip (siendo xx el número de pareja de laboratorio) que contenga todos los ficheros necesarios para reconstruir la aplicación.