

SISTEMAS E SERVIZOS DE INTERNET

Practica 2 – Video en HTML5

Jacobo Martínez Gómez







1. COMPRESIÓN DE FORMATOS

- 1.1. A partir del archivo de vídeo facilitado en la web de la asignatura, realizar la exportación de dicho archivo a los formatos necesarios para poder visualizar dicho vídeo en HTML sin problemas de apertura con los navegadores. Utilizar para ello herramientas como ffmpeg o Miro Video Converter para dicho propósito.**

Se han empleado las herramientas miro video converter y handbrake para las siguientes conversiones de video, en formatos mp4, ogg y webm.

- 1.2. Examinar las posibles diferencias en los tres formatos (MP4, Ogg y WebM) respecto a la calidad y tamaño de los archivos. Describir también los parámetros que más influyen para conseguir una buena relación calidad/tamaño de archivo. Adjuntar alguna captura de fotogramas para comparar los resultados de aplicar diferentes valores a esos parámetros.**

Como podemos observar en la siguiente captura, el formato con menor tamaño es el ogg con 7MB, lo sigue el formato MP4 con casi 9 MB y por último el de mayor peso con 14MB, el formato webm

 dexter.mp4		10/03/2022 23:56	Archivo MP4	9.003 KB
 dexter.ogg		12/03/2022 23:49	Archivo OGG	7.097 KB
 dexter.webm		11/03/2022 0:05	Archivo WEBM	14.179 KB

El parámetro que más influye en la calidad y tamaño de un archivo es el bitrate. Generamos 2 archivos con diferentes bitrates: 100 y 1000.

 Dexter_100kbps.mp4		12/03/2022 23:40	Archivo MP4	3.302 KB
 Dexter_1000kbps.mp4		13/03/2022 12:42	Archivo MP4	14.298 KB



Como se puede observar, a menor bitrate, menor peso de archivo y calidad de imagen: 100kbps 3,3MB y 1000kbps 14,2MB. En la imagen de la izquierda que se corresponde al video codificado con 100kbps de bitrate podemos observar en la parte central como está mas pixelada que la imagen de la derecha, formada por una codificación de 1000kbps de bitrate.

Para concluir, el bitrate es el parámetro mas importante a modificar para conseguir una buena relación de calidad de imagen/tamaño en un archivo de video.

2. CONTROLES DE VIDEO EN HTML5

Realizar una tabla comparativa donde refleje cómo actúa el atributo “controls” en la etiqueta de video HTML5 en los navegadores web más empleados (Internet Explorer, Chrome, Firefox, Safari). Si quisiéramos unificar este aspecto, de modo que la reproducción de un video mostrase siempre el mismo aspecto en sus controles, ¿qué es lo que deberíamos hacer?, ¿Qué opciones tendríamos?

En el navegador Chrome podemos observar todas las opciones en la barra de reproducción. Para pausar/iniciar el video basta con hacer clic encima del video o pulsando la barra espaciadora. Para avanzar o retroceder la barra de reproducción no aporta ninguna opción como puede ser en caso del navegador safari. Para esto tenemos que desplazar la barra según se quiera. Se observa la duración y el minuto actual en la parte izquierda. En la parte derecha tenemos los controles de volumen los cuales son visibles al pasar por encima y por último, en el menú de los 3 puntos podríamos modificar la velocidad de reproducción del video.

Chrome:



Por otro lado, en Firefox lo que observamos a simple vista en comparación con Chrome, es que podemos ver que volumen tenemos seleccionado sin tener que pasar específicamente encima del botón de volumen. La duración y el tiempo transcurrido se encuentran en la parte derecha de la barra. Aquí al igual que en Chrome, para desplazarse en el video tenemos una barra de progreso, donde también podemos utilizar las flechas del teclado para movernos. En Firefox por el contrario no podemos modificar la velocidad de reproducción del video.

Firefox:



Por último, en el navegador Safari podemos observar que tiene 2 opciones más en su barra de reproducción: la de moverse 15s adelante o atrás. Muy similar a la barra de Chrome salvo por que para poner la pantalla completa el botón está fuera de la barra de reproducción, en este caso se encuentra en la parte superior izquierda del reproductor. En las flechas en la parte inferior derecha se encontraría el modificador de velocidad de reproducción.

Safari:

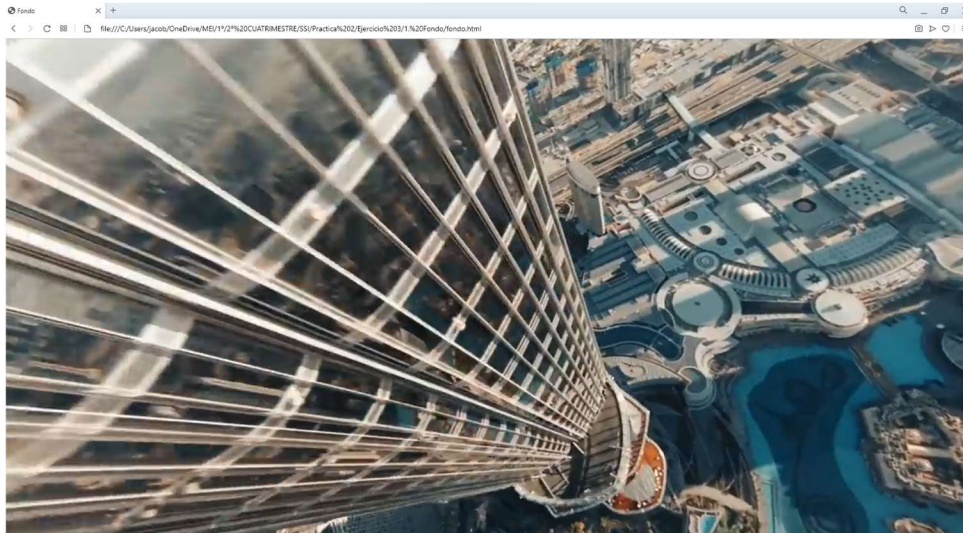


Para unificar el aspecto de la barra de reproducción en todos los navegadores, podemos diseñar la propia nuestra con HTML para la implementación, CSS para el estilado y JavaScript para las funcionalidades de cada elemento creado, por último, sobrescribir el atributo "controls" para que los controles sean ejecutados desde la nueva barra de control. Por otro

lado, podríamos también utilizar alguna API o framework que añada a los controles un estilo específico.

3. VIDEO Y CSS

Usar CSS para posicionar un video en un documento HTML5 de modo que actúe como fondo de todo el interfaz y su reproducción sea instantánea y en modo loop. La solución dada debe ser compatible con los navegadores mas empleados. Si no es posible abarcar todo con CSS, buscar otras posibilidades y describirlas.



Usar CSS para mostrar un cubo donde reproduzca en cada una de sus caras un video diferente de forma que la cubra totalmente, al 100% y sin controles de reproducción. Los videos se ejecutarán al arrancar y en modo loop. Debe ser compatible en cualquier navegador o mostrar indicaciones de ejecución en el que sea compatible. Se valorará la idea final ejecutada.



4. VIDEO Y SVG (SCALABLE VECTOR GRAPHICS)

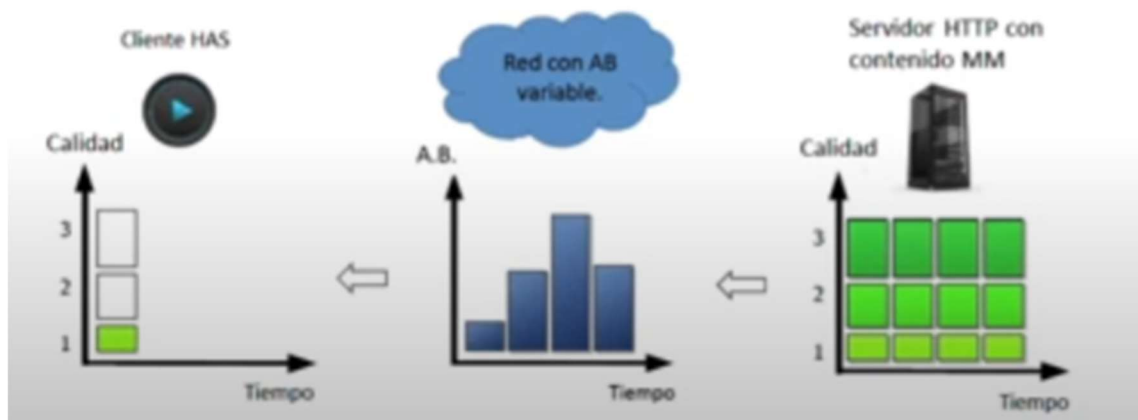
Realizar un pequeño ejemplo donde se exponga como poder aplicar una imagen o un grafico a modo de mascara sobre un elemento de video. Se puede emplear cualquier elemento grafico para ello. Adjuntar el código y describir como se realiza.

5. DIFUSION HTTP ADAPTATIVA

Explica en qué consiste esta técnica, cuáles son sus aplicaciones y describe algún ejemplo si es necesario de como se puede implementar.

Se utiliza para transmitir contenido multimedia de forma adaptada, basándose en la detección de las capacidades de los clientes para ajustar el flujo de transmisión en función del ancho de banda o recursos del equipo. Cuando disminuye el ancho de banda de acceso o cuando hay problemas de red utilizando protocolos como RTP o RTSP el video podría detenerse.

La difusión HTTP adaptativa tiene como característica principal, proporcionar en cada momento la mejor calidad posible según el contexto de cada uno de los clientes. Es preferible perder calidad en la reproducción que el medio reproducido vaya a saltos o tenga repetidas pausas por culpa de la descarga. Esta técnica se consigue generando diferentes copias del contenido con diferente calidad. En la figura de la derecha se puede observar como el cliente consigue descargar el medio en diferentes calidades.



En la grafica del medio se observa la el ancho de banda disponible en cada punto del video, teniendo así calidades diferentes a medida que avanza la reproducción.

En empresas de transmisión de medios como Netflix, HBO, Google, utilizan esta técnica para los diferentes supuestos en los servicios de streaming. Existen implementaciones como HTTP Live Streaming (HLS) de Apple; Microsoft Smooth Streaming (MSS), HTTP Dynamic Streaming (HDS) de Adobe. Por último, destacar que se creó el estándar MPEG-DASH para unir todas las soluciones ya implementadas y hacer que sean compatibles.