UACEE International Journal of Advances in Computer Science and its Applications – IJCSIA
Volume 3 : Issue 2 [ISSN 2250 – 3765]
Publication Date : 05 June 2013

Synote Mobile

HTML5 Responsive Design Video Annotation Application

Mike Wald, E.A. Draffan, Yunjia Li, Wei Jing
ECS
University of Southampton
Southampton, UK

***Abstract*— Synote Mobile has been developed as an accessible cross device and cross browser HTML5 web-based collaborative replay and annotation tool to make web-based recordings easier to access, search, manage, and exploit for learners, teachers and others. It has been developed as a new mobile HTML5 version of the award winning open source and freely available Synote which has been used since 2008 by students throughout the world to learn interactively from recordings. While most UK students now carry mobile devices capable of replaying Internet video, the majority of these devices cannot replay Synote's accessible, searchable, annotated recordings as Synote was created in 2008 when few students had phones or tablets capable of replaying these videos.**

***Keywords*— HTML5, responsive design, video, annotation, captions, mobile**

I. Introduction

Synote Mobile was developed to make web-based recordings easier to access, search, manage, and exploit for learners, teachers and others[1]. It has been developed as a new mobile HTML5 version of the award winning [2] [3] open source and freely available Synote [4] [5] [1] which has been used since 2008 by students throughout the world to learn interactively from recordings. While most UK students now carry mobile devices capable of replaying Internet video, the majority of these devices cannot replay Synote's accessible, searchable, annotated recordings as Synote was created in 2008 when few students had phones or tablets capable of replaying these videos. Synote Mobile is an accessible cross device and cross browser HTML5 collaborative replay and annotation tool. It has the facility to create transcripts manually or automatically using speech recognition transcription and can search transcriptions and annotations with multimedia fragment linked metadata supporting discovery, reuse and replay from a URL. The use of HTML5 overcomes the need to develop multiple device-specific applications. People can individually or collaboratively annotate recordings with synchronised notes and tags to make it easy to bookmark, search, find, or associate their notes or resources with, any specific section of that recording. Synote Mobile provides the interactivity and functionality for recorded material that contents pages, indexes, page numbers, bookmarks and hand written annotation provide for text books. Synote displays the recording, transcript, notes and slide images in four different panels, which uses too much screen area for a small mobile device and so Synote Mobile displays captions and notes and images separately from the video. Synote Mobile enables all students to work together on their coursework, projects and revision in more modern flexible environments than desktop computer rooms not designed for collaborative working. Students can for example collaboratively review and amend recordings and synchronised notes using their phones as well as creating and recording group audio and video presentations annotated with transcripts, indexes and notes.

II. Example Scenario Demonstrating Synote Mobile Use

During the lecture Susan takes short notes on her phone using Twitter and these are automatically uploaded into Synote with the lecture recording after the lecture allowing Susan to easily find relevant sections of the recording using Twitter's timestamps. Susan and her four friends then revise together in a small room in the library by writing on the whiteboard as they collaboratively go over previous Synote recordings and notes using their phones and add to and amend their own synchronised notes as appropriate. They also create and record a group video presentation for their coursework, adding an index and notes. Using Synote Mobile in this way enhances their collaboration compared to the alternative of finding five unused desktop computers next to each other in the main computing laboratory where there is very little desk space and other users complain about their noisy discussions.

III. Design Issues

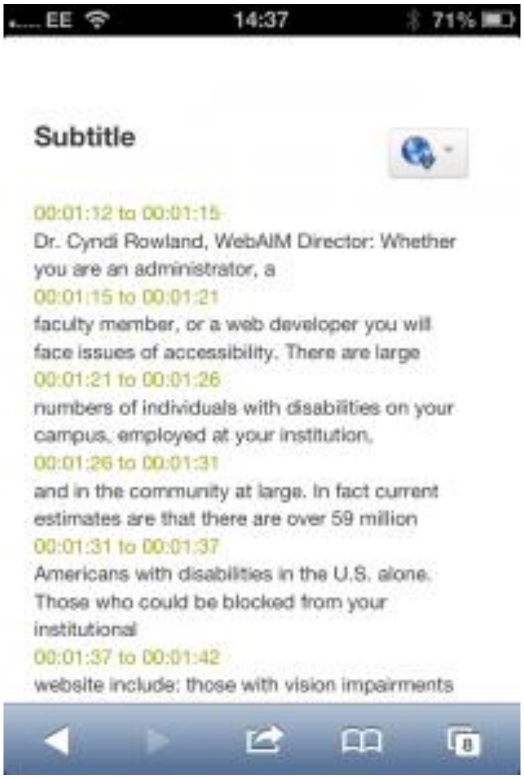
It was decided to develop a single 'responsive' HTML5 website designed to work on all platforms and screen sizes rather than to develop two versions, one for desktop/tablet and one for mobile phones which could have provided an improved user experience on mobile phones but would have required more maintenance and development. Using HTML5 enables: Web browser access with no app store download needed; Automatic enhancements with no need to connect to an app store for updates; Speed, with adaptive performance on

[1] http://linkeddata.synote.org/synote/multimediaResource/list
[2] http://www.ecs.soton.ac.uk/about/news/2598
[3] http://www.ecs.soton.ac.uk/about/news/3874
[4] www.synote.org
[5] http://www.synote.ecs.soton.ac.uk

most connections; Automatic adjustment of presentation depending on device screen size; Use of existing accounts.

A. *Selecting Metadata*

Metadata is very important to show the video and audio in the right way on each device. For responsive design, the correct metadata (e.g. encoding, frame rate etc.) is required and multimedia services like YouTube[6], Vimio[7] and Kaltura[8] have transcoded services with the metadata already available and delivery is speedy and accurate when it comes to player and device choices. With a web server just delivering multimedia files the entire file has to be downloaded before being able to jump to sections and this takes up bandwidth and slows response times at the delivery point with no guarantee of correct player or device choice. There needs to be another service to carry out the transcoding on the fly and this might be possibly implemented by a cloud to cloud service. High level metadata is about how the user interacts with the annotations and how the user sees the media with standards such as HTML5 tracks, accessibility information, schema.org and open graph plus information around the user annotations e.g. notes, comments, captions etc. Combining all the elements regarding the metadata categories and their content would give a truly responsive design for Synote mobile but incomplete HTML 5 and metadata standards make this difficult to accomplish. Pre-transcoding is possible but at the cost of running a service whereas using YouTube has no cost although copyright and confidentiality issues may arise. A lack of mutual understanding between APIs for video services means it is hard to collect media from different services as each has their own semantic descriptions for their API and therefore each link from Synote mobile would have to have its own API to get access to that specific service. Synote Mobile would need to tell users which services it can work with rather than allowing users to take any media from anywhere, as occurs with Synote.



Subtitle

00:01:12 to 00:01:15
Dr. Cyndi Rowland, WebAIM Director: Whether you are an administrator, a
00:01:15 to 00:01:21
faculty member, or a web developer you will face issues of accessibility. There are large
00:01:21 to 00:01:26
numbers of individuals with disabilities on your campus, employed at your institution,
00:01:26 to 00:01:31
and in the community at large. In fact current estimates are that there are over 59 million
00:01:31 to 00:01:37
Americans with disabilities in the U.S. alone. Those who could be blocked from your institutional
00:01:37 to 00:01:42
website include: those with vision impairments

Figure 1. Automated transcript shown on Synote Mobile

B. *Responsive Design*

Responsive designs are required to cope with tablet screens as well as smaller smart phone screens. The ever narrowing of the viewing area meant Synote's video, annotation and transcription windows could shrink and eventually be offered access via tabs. But when it came to mobile phone users there was a need to rearrange the view to be totally linear. In other words two designs of Synote Mobile were required but with changes that would happen automatically depending on the metadata received.. Responsive design also needs to provide fall forward and fall back options for embedding media players within web pages that will automatically adapt to the user's chosen device. A challenge for HTML5 video in Synote is to embed different players not only based on the media type, but also the platform. As Flash is not well supported on mobile platforms the HTML5 native player needs to be controlled through JavaScript. MediaElement.js is a "fallforward" html player, which means it is based on HTML5 native player and if the browser doesn't support HTML5, MediaElement.js will embed the self-developed Flash and Silverlight player. Comparisons of codecs and applicable browsers, including mobile devices[9] and HTML5 video players[10] with available features were studied in detail when designing Synote Mobile. It is very important to gather the metadata from the media so that the correct view and player can be selected. Three approaches are FFmpeg [11], YouTube api with metadata such as title, resolution etc. The link to the file itself offers metadata which can be extracted by a commercial service and this would need to be re-created if a free service is required. The minimum amount of data needed is the duration of the media, the format and coding and whether it is video or audio.

C. *Captions*

Captions can be displayed on a desktop browser with the video but there appear to be no standards for displaying captions within web pages across all tablets and mobiles alongside transcriptions and annotations. At present if someone is deaf they have to read the caption and watch the video and then scroll down to the note-taking mode. It is not possible on the iPhone to display both the transcript window and the video due to the size of the screen. However, it will be possible to capture an image from the video and annotate this

[6] www.youtube.com
[7] www.vimeo.com
[8] www.kaltura.com

[9] http://mediaelementjs.com/
[10] http://praegnanz.de/html5video/
[11] http://ffmpeg.org/

as part of the note taking process. One issue with captioning overlayed on the video (as suggested for Synote Mobile) rather than in a separate panel (as in Synote) is how can a hearing impaired person read the captions when the video is a thumbnail or not displayed at all (figure. 1)? There needs to be a way the captions can be displayed separately and Matterhorn designers are also now thinking about ways to achieve this.[12]

D. *Requirements*

Synote Mobile requirements include;

- SCREEN SIZE ISSUES: work on mobile phone screens as well as tablets so it should automatically detect the device and load the corresponding page or style sheet.
- DELIVERY OF THE VIDEOS: adaptive i.e. devices with low resolution and bandwidth need to download smaller file sizes. The player will take over the full screen when playing, so it will not be possible to have a split screen version of the transcriptions or the annotations and therefore it is proposed that there will be a thumbnail picture displayed alongside the video with the annotations.
- FORMAT COMPATIBILITY: Different devices cause format compatibility complications and Safari, Google Chrome and mobile IE also have different support mechanisms for HTML video codecs [13]. This means the delivery of multimedia resources has to be adaptable.
- TOUCH INTERFACE: There needs to be careful consideration given to the type of gesture driven / tap type controls. Users who wish to have VoiceOver controls will affect the gestures that can be used within not only the player but also the web pages. There is a need to keep clutter to a minimum with not too many buttons onscreen and the design aimed to follow the accessible and ease of use design concepts mentioned in the blog "Research – Planning for an easy to use and accessible mobile app." [14]
- HTML5 VIDEO: Streaming video is easier on tablets than on mobile phones[15] and is the choice for delivering a cross platform service although when fullscreen viewing with iPhones[16] the video is no longer browser based and not presented within the webpage. This fullscreen issue makes it impossible to add captions unless they are embedded when the video is made and external files will be unable to be read with the method presently used by iPhones for rendering videos. Exploring the issues around streaming video on mobile phones and tablets, it appears the latter is easier than the former. The state of HTML5 and video is well explained by LongTail Video[17] and it looks as if this is the way to go when one wants to deliver a cross platform service. As an illustration of the constantly changing scene recent updates on video streaming would now allow us to use an Apple streaming video server[18] that would appear to allow for captioning on the iPhone whereas this is not possible with YouTube videos. HTML video.org offer a helpful comparison of players [19].

IV. Testing & Evaluation

Synote Mobile has been developed as a HTML5 mobile phone web application that allows playing of videos, and reading transcripts and annotations but how much access can be achieved depends on the operating system, browser and player especially in terms of reaching the video player controls and captioning within the web page. It is easy to navigate around a specifically designed mobile web site and most features can be used in the phones available as of September 2012. However, testing also showed it is possible to work with the videos on a smaller screen of 3.5 inches although it may take three scrolls to reach the bottom and require some horizontal scrolling depending on whether one is viewing in portrait or landscape mode. When it comes to outputs it is possible to add a transcript and annotations, although small fingers and delicate typing may be required. Synote Mobile was evaluated by a group of users on a range of devices. On all the devices there was a need to buffer the video for a few seconds before any of the menus would work because mobile phones are set to avoid the download large amounts of data and no optimisation is possible. The overall opinion of the website was good and the main purpose was to discover what further issues were arising and whether users had any other requests from the point of view of the look and feel as well as usability and accessibility. The iOS onscreen keyboard always stayed below edit boxes whereas the Android one covered them on the Samsung Galaxy.

A. *Tests of HTML 5 Video and Support in Mobile Browsers*

The HTML5 video tag was proposed by Opera Software in 2007 but there are no systematic tests for the video compatibility on different mobile browsers. Tests were carried out on Android devices and iPhones with varied screen size from 3.0 to 4.8 inches to find out which video format could be played in which latest versions of mobile browsers. Test Phones were: HTC Design S, Android 2.3.5, 3.7-inch touch screen, Sony SK17i, Android 2.3.4, 3.0-inch touch screen, Samsung Galaxy S II, Android version 2.3.3, 4.3-inch touch screen, HTC G10, Android 2.3.5, 4.3-inch touch screen, Samsung Galaxy S III, 4.8-inch touch screen, iPhone 4S, 3.5 inches screen, iPhone 4, 3.5 inches screen.

[12] http://developer.apple.com/library/safari/#documentation/AudioVideo/Conceptual/Using_HTML5_Audio_Video/Device-SpecificConsiderations/Device-SpecificConsiderations.html

[13] http://www.htmlgoodies.com/html5/client/how-to-embed-video-using-html5.html#fbid=MOq4FmNWQzi

[14] http://access.ecs.soton.ac.uk/blog/synotemobile/2012/03/23/research-planning-for-an-easy-to-use-and-accessible-mobile-app/

[15] http://www.longtailvideo.com/html5

[16] http://developer.apple.com/library/safari/#documentation/AudioVideo/Conceptual/Using_HTML5_Audio_Video/Device-SpecificConsiderations/Device-SpecificConsiderations.html

[17] http://www.longtailvideo.com/support/addons/captions-plugin/14974/captions-plugin-reference-guide

[18] http://tools.ietf.org/html/draft-pantos-http-live-streaming-09

[19] http://html5video.org/wiki/HTML5_Player_Comparison

B. *Playable Test*

This test shows if a video file with a certain codec could be played in a certain browser on a mobile phone. mp4, ogv and WebM files were tested on Android 2.3, 4.0 and iOS 5 with different mobile browsers on different mobile OSs. Summarising the findings: There is no video codec format that works in all mobile browsers currently but MP4 was compatible with the "default" browsers, i.e. Android Native browser in Android phones and Safari in iPhones; No video could be played in Opera Mini Different from desktop browsers; WebM does not work with native players on mobile browsers even the Android Native Browser, which is developed by Google; Apart from Firefox, most mobile phones play the video in full screen mode by default and users cannot quit the full screen unless stopping the video; If the screen size of the mobile phone is larger than 4 inches sometimes, video won't be played in full screen mode automatically.

C. *WebVTT Captioning Test*

HTML5 video tag adds the track tag for subtitles which are very important for the accessibility of videos on the Web. Many standards have been around for a long time but Web Video Text Track (WebVTT)[20] is a new developing standard for videos on the Web that can also be used for audio description, chapter navigation, etc. Tests were conducted to determine whether mobile browsers support WebVTT natively and the answer was that none of them did. There are HTML5 video players that support WebVTT on desktop browsers but they are all "polyfills" they use JavaScript and CSS to present WebVTT together with the video. MediaElement.js failed to display the WebVTT subtitles on iPhones but was successful on the iPad. The test for HTML5 video tag and WebVTT on mobile browsers is important. Developers should be clear what video could be played in the target device. On the server side, the web applications need to host the best format compatible with the target device. Video sharing services, such as YouTube, must be adaptive enough to deliver videos to different devices and browsers. Subtitles are always important for videos, especially on mobile platforms, where the support for subtitles is not thorough yet. With each updates or release of new versions of browsers, new codecs might be supported and old formats might be abandoned. WebVTT is currently is not well supported on mobile browsers and sp we are still waiting for some break-through.

V. Discussion of results

The different devices, operating systems and browsers all produced different results testing with the mobile phone version of Synote despite the fact that the pages have been designed in HTML5 with MediaElement.js player and jQuery mobile. The transcript, Synmarks and the PowerPoint slides all appear to display in the same way on the various browsers and devices but issues arose with videos and the MediaElement.js player[21]. YouTube videos are presented on iOS 5 (iPhone) where the controls work but the video is launched in the iOS player and Android where there are variations in the software used on different phones e.g. some HTC and Sony phones. On the latter the video is seen but cannot be played using an Opera browser but will play in the built in Android Webkit browser – on the former an error message appears within the Chrome browser as it cannot load the player. There is no mobile Firefox for HTC Wildfire. The HTC OneXL with a 4-inch screen will run Synote Mobile in the same way as the tablet and desktop version. Audio is not a problem as the transcript can appear below the player in this case. Synmarks can also be seen with a separate tab. The view of the video page has titles, tags and description and the options page allows the user to go to the video or audio page, transcript, slides or Synmarks. The transcript has the thumbnail picture from the video and the start and end times for each transcript block. The slides can be displayed in a row or line with start times and actual slides, the Synmarks are listed with title, description and tags plus start and end time and a thumbnail of the YouTube video for that time. At present there is also a huge variation between the way mobile phones show the captions as taken from the transcript because the players react differently to the selection of the caption button. The iPhone would not play the YouTube closed captions whether we used Chrome, Opera or Safari despite having closed caption settings on and this appears to be a known problem[22].

A group of users provided feedback on the prototype of Synote Mobile using 7 and 9.7 inch tablets with Safari, Opera

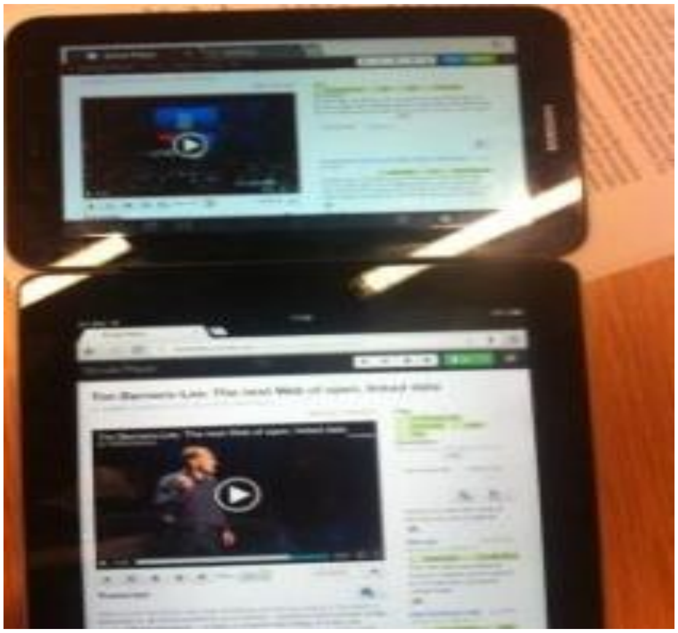


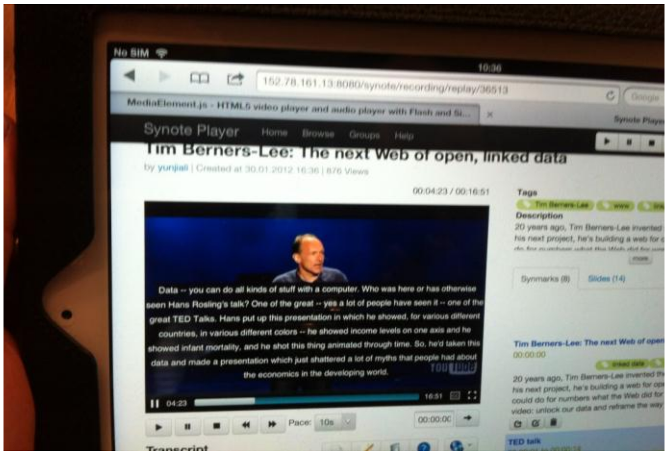
Figure 2. Comparing Galaxy and iPad presentation of Synote mobile

[20] http://dev.w3.org/html5/webvtt/

[21] http://mediaelementjs.com/

[22] http://audio-accessibility.com/news/2012/02/youtube-apples-lack-of-caption-support/

and Chrome browsers (see Figure. 2). CSS3 media query and viewing port size[23] is used to decide when working with multimedia materials how they will be displayed on devices so the user automatically experiences a comfortable presentation. In terms of Synote Mobile this means that when the viewing window is below 768 pixels the two columns that hold the transcript and Synmarks (annotations) collapse into one. When a video or audio is played the Synmarks will be on one tab with the closed captions allowing you to see the transcript across the window. The user can also choose just to read the transcript via that tab. One issue that was discovered by taking screen size as a method for changing presentation of videos and transcripts is that the CSS3 media query is dependent on screen size and videos are dependent on pixel widths. It is a tricky problem to explain as it appears that CSS media queries result in different behaviours for continuous media and paged media.[24] There is also an inconsistency in behaviour across devices and browsers. The controls are dependent on viewing modes, devices and browsers so when using landscape mode on the iPad it senses that it is in desktop mode. When playing the video and viewing the transcript the player controls are always seen but when using Portrait mode the CSS media query senses the reduced width and represents the web page without the player controls as you scroll down. It is possible to solve the problem on some devices, but because this does not work for all devices it has been decided that 'width' is probably the best method for defining when the controls will be shown. Bootstrap[25] is being used, which is one of the most consistent environments across all devices when it comes to presenting media data. When discussing other user issues such as the look and feel, the Synote Mobile website was considered to be easy to use and attractive but there were one or two login issues on the iPhone. One user suggested that the highlights of Synmarks (annotations) should be different from the transcript highlights as the video or audio plays. A caption longer than a few lines when viewed on the video will overlay too much of the picture (Figure. 3). YouTube have set their automatic speech recognition captioning to always generate 2 lines. If the transcript is put in manually the size can be reduced to two lines for each timed section but this can result in captions appearing for too short a time to be read. By using YouTube as the upload vehicle it is possible to have an automated transcript that can be corrected and annotated.


No SIM
10:36
152.78.161.13:8080/synote/recording/replay/36513
Google
MediaElement.js - HTML5 video player and audio player with Flash and Si...
Synote Player
Synote Player
Home
Browse
Groups
Help
Tim Berners-Lee: The next Web of open, linked data
by yunjiali | Created at 30.01.2012 16:36 | 876 Views
00:04:23 / 00:16:51
Tags
Tim Berners-Lee
www
Description
more
Synmarks (8)
Slides (14)
Tim Berners-Lee: The next Web of open
00:00:00
linked data
04:23
16:51
Pace:
10s
00:00:00
Transcript
TED talk

Data -- you can do all kinds of stuff with a computer. Who was here or has otherwise seen Hans Rosling's talk? One of the great -- yes a lot of people have seen it -- one of the great TED Talks. Hans put up this presentation in which he showed, for various different countries, in various different colors -- he showed income levels on one axis and he showed infant mortality, and he shot this thing animated through time. So, he'd taken this data and made a presentation which just shattered a lot of myths that people had about the economics in the developing world.

20 years ago, Tim Berners-Lee invented his next project, he's building a web for could do for numbers what the Web did for video: unlock our data and reframe the way

Figure 3. Captions obscuring video

VI. Conclusion

Synote Mobile[26] is a web application capable of working in the majority of browsers on mobile devices running an Android, Windows, Apple iOS system. It was found that one of the easiest ways to make videos suitable for Synote Mobile was to use YouTube with its captioning and transcription service that allowed for timed stamped data to be annotated and shared with others[27]. There were numerous issues relating to media types, presentation of transcriptions and annotations that needed to be resolved during the project. These were discussed in a series of blogs the most important being the development of a matrix to show which browsers could play videos within a browser[28]. While it might have been easier to have made a series of device specific apps offering more usability features in terms of button sizes, menu options and player modes, this would not have allowed for as many other access requirements, as was possible through the development of an HTML 5 version. A future approach could be a responsive delivery on the fly from a cloud storage system to a cloud transcoding service but this would once again come at a cost. The fall forward approach of MediaElement.js means that should the HTML5 player fail on the device Synote Mobile will present the user with Silverlight or Flash and vice versa in a fall back situation. In our tests we used some popular phones and browsers and it will be interesting in future work to compare the HTML5 video tag in all browsers across all different types and sizes of phones and tablets. The chosen solution appears to work with most browsers despite the lack of player access alongside the transcription on smaller mobile devices. It allows for accessibility with captions and transcriptions where the technology has allowed and provides the user with a way of interacting with others whilst working with video and audio files. This allows for video captured lectures to be not only more accessible to those who have hearing impairments but also allows all students to go back over content in a way that may suit their learning preferences whether they are in the university, at home or when travelling.

References

[1] Li, Y., Wald, M., Wills, G., Khoja, S., Millard, D., Kajaba, J., Singh, P. and Gilbert, L. (2011) Synote: development of a Web-based tool for synchronized annotations. *New Review of Hypermedia and Multimedia*. pp.1-18

[23] http://www.lgloss.com/tutorials/learnmod2/mediaqueries.php
[24] http://h30565.www3.hp.com/t5/Feature-Articles/CSS-From-Screen-to-Print-and-Beyond/ba-p/1752
[25] http://twitter.github.com/bootstrap/

[26] http://linkeddata.synote.org/
[27] http://linkeddata.synote.org/synote/recording/replay/52593
[28] http://access.ecs.soton.ac.uk/blog/synotemobile/2012/09/03/html-5-video-and-support-in-mobile-browsers