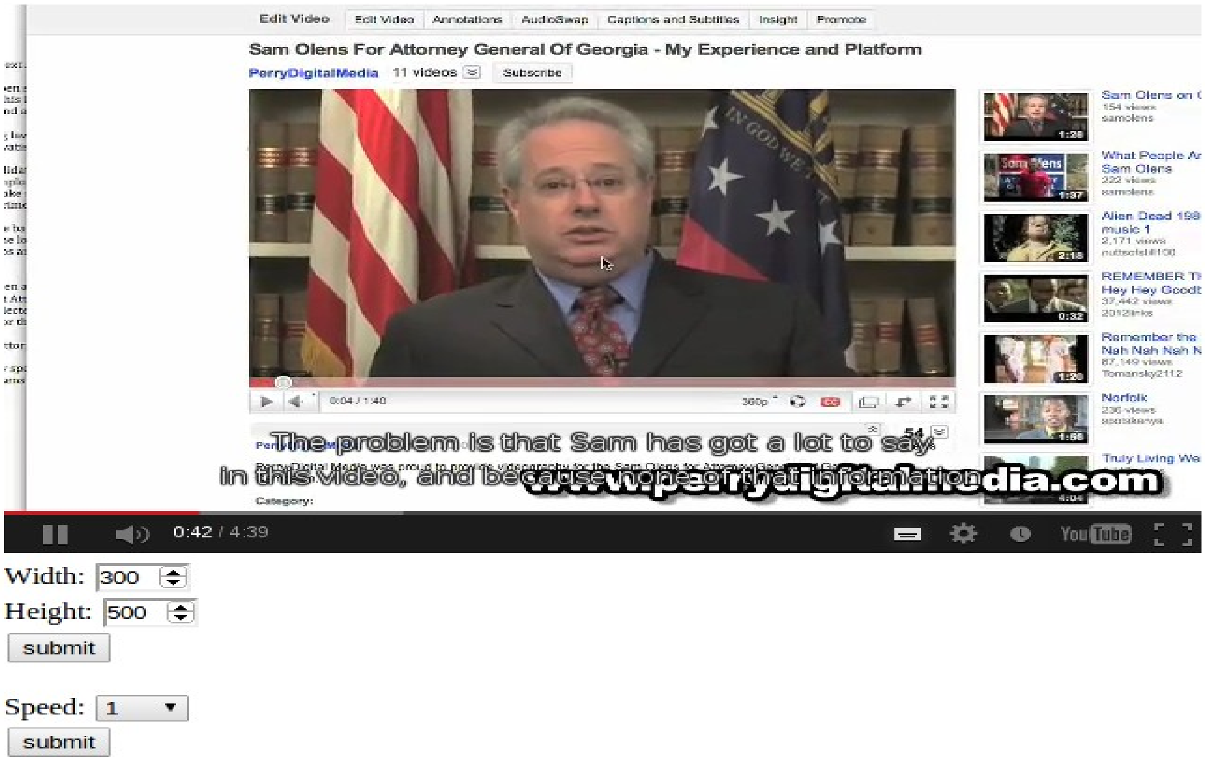
You tube Player Customization

- resizing player
- changing Playback Rate
- Quiz during video
- transcript display
- interaction between transcript and video
- subtitles display

Resizing Player

input : width , height
output : current video will be resized with the provided width and height



Sam Olens For Attorney General Of Georgia - My Experience and Platform

The problem is that Sam has got a lot to say in this video, and because none of that information

www.perrydigitalmedia.com

Width: 300

Height: 500

submit

Speed: 1 ▾

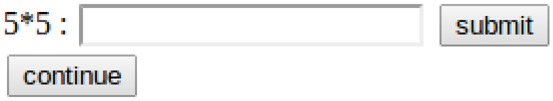
submit

Changing Playback Rate

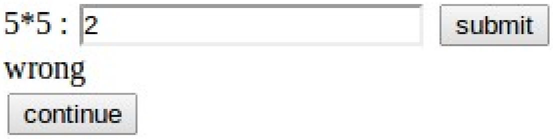
input : speed
output : playback Rate of video will be changed according to speed.

Quiz During Video

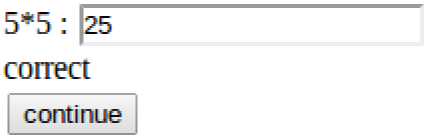
It automatically goes to quiz (without reloading).
After quiz completion video plays accordingly where user left.



5*5 : submit
continue

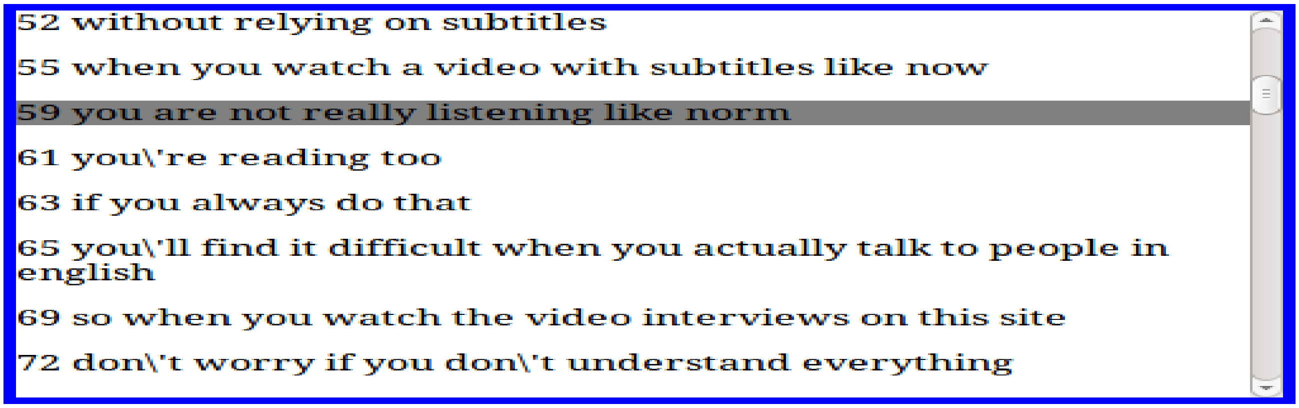


5*5 : 2 submit
wrong
continue



5*5 : 25
correct
continue

Transcript Display



52 without relying on subtitles

55 when you watch a video with subtitles like now

59 you are not really listening like norm

61 you\'re reading too

63 if you always do that

65 you\'ll find it difficult when you actually talk to people in english

69 so when you watch the video interviews on this site

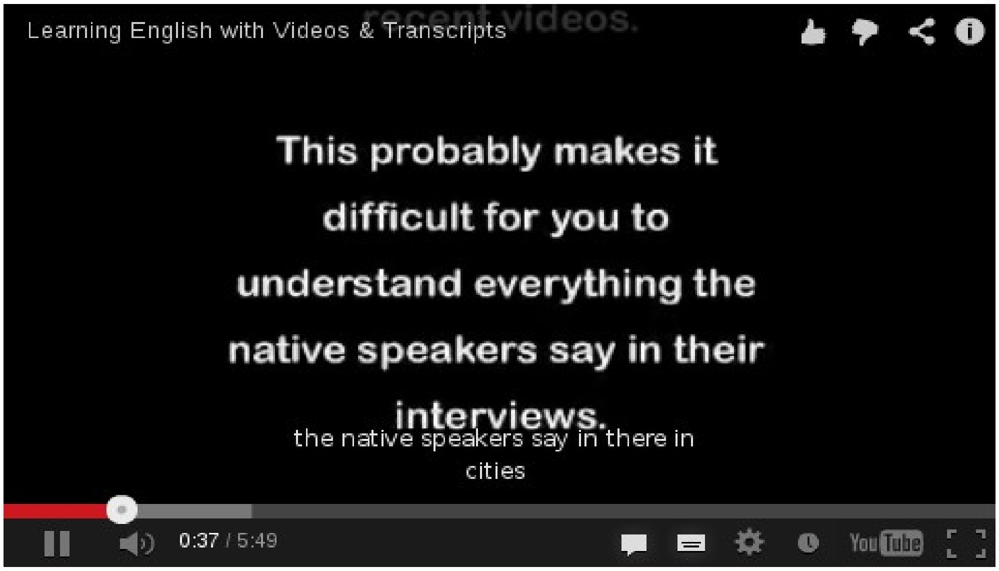
72 don\'t worry if you don\'t understand everything

Transcript displayed for more understanding and interaction with the video

Interaction between transcript and video

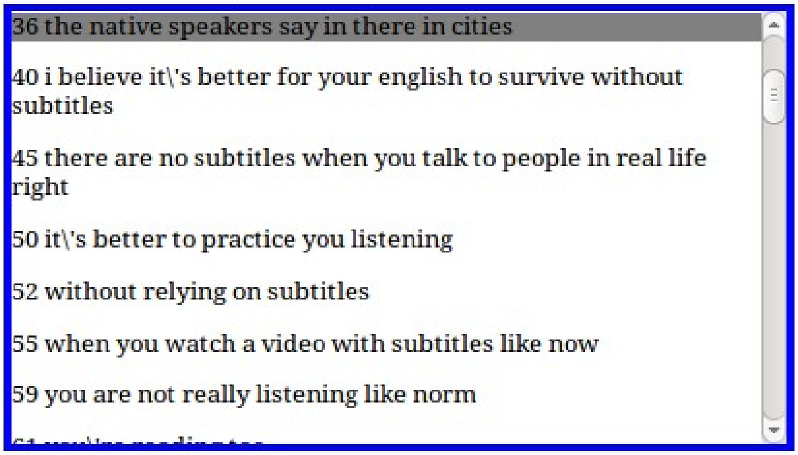
Functionalities:

1. transcript automatically follows video
2. Video follows transcript (when user clicks particular sentence)
3. It highlights the transcript while synchronizing with video.



This probably makes it difficult for you to understand everything the native speakers say in their interviews.

the native speakers say in there in cities



36 the native speakers say in there in cities

40 i believe it\'s better for your english to survive without subtitles

45 there are no subtitles when you talk to people in real life right

50 it\'s better to practice you listening

52 without relying on subtitles

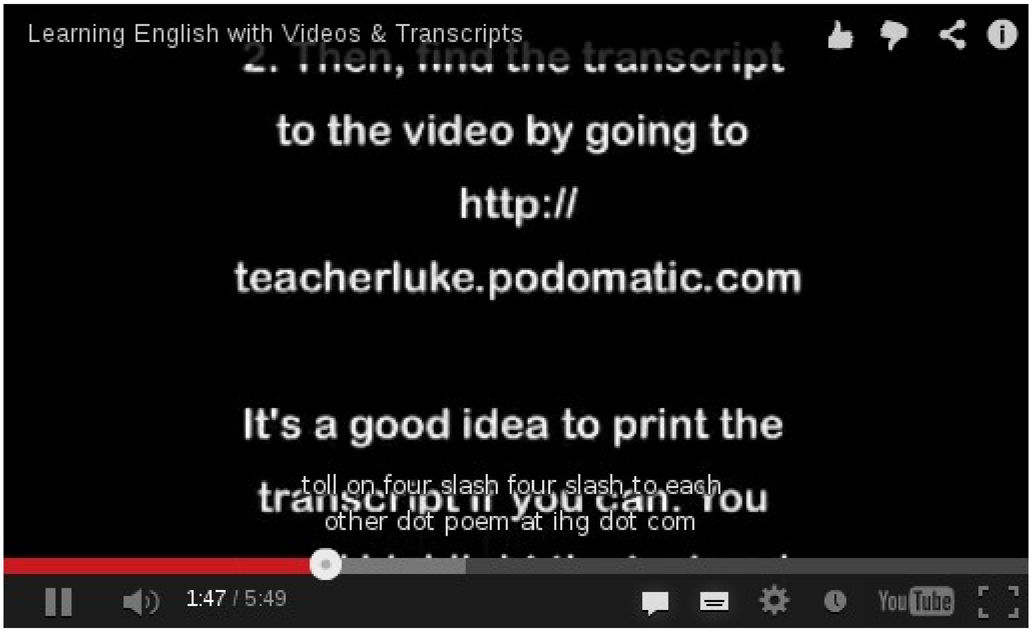
55 when you watch a video with subtitles like now

59 you are not really listening like norm

61 you\'re reading too

Subtitles display

Subtitle are displayed for better user understanding



Learning English with Videos & Transcripts

2. Then, find the transcript to the video by going to http:// teacherluke.podomatic.com

It's a good idea to print the transcript if you can. You

toll on four slash four slash to each other dot poem at ihg dot com

Coding

Quiz:

```
<!DOCTYPE html>
<html>
  <body>
    <!-- 1. The <iframe> (and video player) will replace this <div> tag. -->
    <div id="player"></div>
    <div id="playerDetails"></div>
    <div id="exam">
                        5*5 : <input id="result" type="text"/>
                        <input id="b1" type="button" value="submit" onclick="test()"/>
                        <div id="disp"></div>
                        <div id="cont"><input type='button' value='continue' onclick='change()'/></div>
    </div>
    <script>
                        function test(){
                                    if(document.getElementById("result").value=="25"){
                                                document.getElementById("disp").innerHTML="correct";
                                                document.getElementById("b1").style.display="none";
                                    }
                                    else{
                                                document.getElementById("disp").innerHTML="wrong";
                                    }
                        }
                        function change(){
                                    document.getElementById("exam").style.display="none";
                                    document.getElementById("player").style.display="block";
                        }
                        //KfWDcfYePm4,WDDgoxvQsrQ
      // 2. This code loads the IFrame Player API code asynchronously.
      var tag = document.createElement('script');
                        document.getElementById("exam").style.display="none";
      tag.src = "https://www.youtube.com/iframe_api";
      var firstScriptTag = document.getElementsByTagName('script')[0];
      firstScriptTag.parentNode.insertBefore(tag, firstScriptTag);

      // 3. This function creates an <iframe> (and YouTube player)
      //    after the API code downloads.
      var player;
      function onYouTubeIframeAPIReady() {
        player = new YT.Player('player', {
          height: '390',
          width: '640',
          videoId: 'M7lc1UVf-VE',
          events: {
            'onReady': onPlayerReady,
            'onStateChange': onPlayerStateChange
          }
        });
      }

      // 4. The API will call this function when the video player is ready.
      function onPlayerReady(event) {
        event.target.playVideo();
      }

      // 5. The API calls this function when the player's state changes.
      //    The function indicates that when playing a video (state=1),
      //    the player should play for six seconds and then stop.
      var done = false;
      function onPlayerStateChange(event) {
        if (event.data == YT.PlayerState.PLAYING && !done) {
          setTimeout(stopVideo, 5000);
          //document.getElementById("playerDetails").innerHTML=player.getCurrentTime();
          done = true;
        }
      }
      function stopVideo() {
        player.pauseVideo();
        document.getElementById("player").style.display="none";
        document.getElementById("exam").style.display="block";
      }
    </script>
  </body>
</html>
```

Resizing and Playback Rate Change:

```
<!DOCTYPE html>
<html>
  <body>
    <!-- 1. The <iframe> (and video player) will replace this <div> tag. -->
    <div id="player"></div><br/>
    <div id="resizing">
                    Width: <input id="width" type="number" min="200" max="700"/><br/>
                    Height: <input id="height" type="number" min="200" max="700"/><br/>
                    <input id="b1" type="button" value="submit" onclick="resize()"/><br/><br/>
                    Speed:
                    <select id="speed">
                            <option value="0.25">0.25</option>
                            <option value="0.5">0.5</option>
                            <option value="1">1</option>
                            <option value="1.5">1.5</option>
                            <option value="2">2</option>
                    </select>
                    <br/>
                    <input id="b1" type="button" value="submit" onclick="changeSpeed()"/><br/>
                    <div id="disp"></div>
    </div>
    <script>

                    //KfWDcfYePm4,WDDgoxvQsrQ
      // 2. This code loads the IFrame Player API code asynchronously.
      var tag = document.createElement('script');
      document.getElementById("resizing").style.display="none";
      tag.src = "https://www.youtube.com/iframe_api";
      var firstScriptTag = document.getElementsByTagName('script')[0];
      firstScriptTag.parentNode.insertBefore(tag, firstScriptTag);

      // 3. This function creates an <iframe> (and YouTube player)
      //    after the API code downloads.
      var player;
      function onYouTubeIframeAPIReady() {
        player = new YT.Player('player', {
          height: '390',
          width: '640',
          videoId: 'ULNrcqdjAt0',
          events: {
            'onReady': onPlayerReady,
            'onStateChange': onPlayerStateChange,
           // 'onPlaybackRateChange':onRateChange

          }
        });
      }

      // 4. The API will call this function when the video player is ready.
      function onPlayerReady(event) {
        event.target.playVideo();
      }

      // 5. The API calls this function when the player's state changes.
      //    The function indicates that when playing a video (state=1),
      //    the player should play for six seconds and then stop.
      var done = false;
      function onPlayerStateChange(event) {
        if (event.data == YT.PlayerState.PLAYING && !done) {
          //setTimeout(stopVideo, 3000);
          document.getElementById("resizing").style.display="inline";
        }
      }
      //function onRateChange(){
                         // document.write("hi");
              //}
      function resize()
      {

player.setSize(document.getElementById("width").value,document.getElementById("height").value);
              }
              function changeSpeed(){
                          var val=document.getElementById("speed").value;
                          player.setPlaybackRate(val);
              }
      function stopVideo() {
        player.stopVideo();
      }
    </script>
  </body>
</html>
```

Transcript Display and Synchronizing with player:

```
<!DOCTYPE html>
<html>
            <head>
                        <style>
                        #disp{
                                                width:500px;
                                                height:300px;
                                                border:5px solid blue;
                                                margin:10px;
                                                overflow:auto;
                                                //display:none;
                                    }
                        </style>
            </head>
  <body>
    <!-- 1. The <iframe> (and video player) will replace this <div> tag. -->
    <div id="player"></div>
    <!--<iframe src="parsexml.html" width="640" height="390" frameborder="0"></iframe>-->
    <div id="disp"></div>
    <div id="xml"></div>
    <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
    <script>



                        //KfWDcfYePm4,WDDgoxvQsrQ
      // 2. This code loads the IFrame Player API code asynchronously.
      var tag = document.createElement('script');
      tag.src = "https://www.youtube.com/iframe_api";
      var firstScriptTag = document.getElementsByTagName('script')[0];
      firstScriptTag.parentNode.insertBefore(tag, firstScriptTag);

      // 3. This function creates an <iframe> (and YouTube player)
      //    after the API code downloads.
      var player;
      function onYouTubeIframeAPIReady() {
        player = new YT.Player('player', {
          height: '390',
          width: '640',
          videoId: 'ar8U_q34FRs',
          events: {
            'onReady': onPlayerReady,
            'onStateChange': onPlayerStateChange
          }
        });
    }
        prev=undefined;
    arr=new Array();
    i=0;
    // 4. The API will call this function when the video player is ready.
    function onPlayerReady(event) {
     event.target.playVideo();
     $.ajax({
                type : 'GET',
                dataType : 'xml',
                url : 'http://query.yahooapis.com/v1/public/yql?q=select%20*%20from%20xml
%20where%20url%3D%22http%3A%2F%2Fvideo.google.com%2Ftimedtext%3Flang%3Den%26v
%3Dar8U_q34FRs%22',
                success : function(xhtml) {
                $(xhtml).find('text').each(function(){
                        var str=$(this).attr('start');
                        $('#disp').append('<p>'+str.substr(0,str.indexOf("."))+' '+$(this).text()+'</p>');
                        arr[i]=str.substr(0,str.indexOf("."));
                        var p = document.getElementsByTagName("p")[i];
                        p.setAttribute('start',arr[i]);
                        p.onclick=function(){
                                console.log("working!!");
                                player.seekTo(p.getAttribute("start"),true);
                                p.style.backgroundColor="gray";
                                if(prev==undefined){
                                        prev=p;
                                }
                                else{
                                        prev.style.backgroundColor="white";
                                        prev=p;
                                }
                        };
                        i++;
                        });
                        setInterval(function(){
                                var videoDuration=player.getDuration();
                                var currentTime=player.getCurrentTime().toString();
                                var ct=currentTime.substr(0,currentTime.indexOf("."));
                                var k=0;
                                while(k<arr.length){
                                        var p=document.getElementsByTagName("p")[k];
                                        if(p.getAttribute('start')==ct){
                                                p.scrollIntoView();
                                                p.style.backgroundColor="gray";

                                                if(prev==undefined){
                                                        prev=p;
                                                }
                                                else{
```

```
                                        prev.style.backgroundColor="white";
                                        prev=p;
                                   }
                             }
                             k++;
                       }
                 },999);
            }
            })
      }

      // 5. The API calls this function when the player's state changes.
      //    The function indicates that when playing a video (state=1),
      //    the player should play for six seconds and then stop.
      var done = false;
      function onPlayerStateChange(event) {
       if (event.data == YT.PlayerState.PLAYING && !done) {
        done = false;
       }
      }

    </script>
  </body>
</html>
```

Prepared By

Mahendra Mundru,
MOOC Team.