Using Google Text-To-Speech in Javascript

Asked 7 years, 6 months ago Active

Thanks for the feedback! Votes cast by those with less than 15 reputation are recorded, but do not change the publicly displayed post score.





I need to play **Google text-to-speech** in JavaScript.

The idea is to use the web service:





And play it on a certian action, e.g. a button click.

43

But it seems that it is not like loading a normal wav/mp3 file:

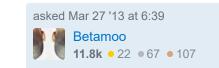
```
<audio id="audiotag1" src="audio/example.wav" preload="auto"></audio>

<script type="text/javascript">
    function play() {
        document.getElementById('audiotag1').play();
    }
</script>
```

http://translate_google.com/translate_tts?tl=en&q=This%20is%20just%20a%20test

How can I do this?

javascript google-text-to-speech



Active

Oldest

Votes

6 Answers



Another option now may be HTML5 text to speech, which is in Chrome 33+ and many others.

1/12

Here is a sample:



```
var msg = new SpeechSynthesisUtterance('Hello World');
window.speechSynthesis.speak(msg);
```

With this, perhaps you do not need to use a web service at all.

edited Apr 25 '17 at 12:33

answered Mar 6 '14 at 19:24



- The great thing about the HTML5 Speech Synthesis is that you can adjust the voice pitch, fluctuation, etc Jake Wilson Nov 15 '14 at 3:01
- 1 Supported by Safari, IE? Jasper May 5 '16 at 13:34

You should add properties such as pitch, rate, speed, and etc. – user8903269 Mar 15 '18 at 0:17

@Rejaul Works in Chrome 76 - Brian M. Hunt Jul 29 '19 at 18:43

This works on Chrome, Firefox, Edge, Opera and not on IE – Nanju Mar 18 at 5:44



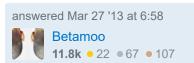
Here is the code snippet I found:



```
var audio = new Audio();
audio.src ='http://translate.google.com/translate_tts?ie=utf-8&tl=en&q=Hello%20World.';
audio.play();
```







- 3 It seems that if you remove the 'dot' at the end it works fine, otherwise it's not playing the sound. Diego Sep 17 '13 at 15:53
- 5 Keep in mind that Google Translate has a limit of ~100 letters. niutech Jan 8 '14 at 1:25
- 7 It seems that Google will ban requests with a Referrer in the HTTP header. Is there a way to bypass that issue? jichi Jan 31 '14 at 20:03

@jichi Take a look on this answer and comments to this answer. – Piotr Sobczyk Aug 3 '14 at 7:13

15 As of today this doesn't seem to be working any more; I get 302 and then 403. – icedwater Jan 28 '16 at 13:59

9

Then you only need to call say(msg) when using it.

Update: Look at Google's Developer Blog that is about Voice Driven Web Apps Introduction to the Web Speech API.



answered Mar 2 '18 at 19:10 user8903269

are you the same user as the JudahRR who wrote say.js? If so, you should disclose that affiliation. Also, probably no need to consume an external library for a 10 line function that just consumes the browser's native API anyway and doesn't need all of the default settings. – KyleMit Dec 23 '18 at 1:01



Very easy with responsive voice. Just include the js and voila!

14

```
<script src='https://code.responsivevoice.org/responsivevoice.js'></script>
<input onclick="responsiveVoice.speak('This is the text you want to speak');"
type='button' value='@ Play' />
```

answered Oct 16 '18 at 22:06

ManuelC

141 • 1 • 2

Important note here: Although this has interesting immediate results, it is using their API via https://code.responsivevoice.org/getvoice.php to generate voices and send them back as rendered audio. Their service is however not free for non commercial projects. – Jankapunkt Sep 5 '19 at 10:50



Run this code it will take input as audio(microphone) and convert into the text than audio play.

```
4
```



```
<!doctype HTML>
<head>
<title>MY Echo</title>
<script src="http://code.responsivevoice.org/responsivevoice.js"></script>
<link rel="stylesheet" type="text/css"</pre>
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.6.1/css/font-
awesome.min.css" />
<style type="text/css">
    body {
        font-family: verdana;
    }
    #result {
        height: 100px;
        border: 1px solid #ccc;
        padding: 10px;
        box-shadow: 0 0 10px 0 #bbb;
        margin-bottom: 30px;
        font-size: 14px;
        line-height: 25px;
    }
    button {
        font-size: 20px;
        position: relative;
        left: 50%;
</style>
```

Speech to text converter in JS var r = document.getElementById('result');

```
function startConverting() {
   if ('webkitSpeechRecognition' in window) {
        var speechRecognizer = new webkitSpeechRecognition();
        speechRecognizer.continuous = true;
        speechRecognizer.interimResults = true;
        speechRecognizer.lang = 'en-IN';
        speechRecognizer.start();
        var finalTranscripts = '';
        speechRecognizer.onresult = function(event) {
            var interimTranscripts = '';
            for (var i = event.resultIndex; i < event.results.length; i++) {</pre>
                var transcript = event.results[i][0].transcript;
                transcript.replace("\n", "<br>");
                if (event.results[i].isFinal) {
                    finalTranscripts += transcript;
                    var speechresult = finalTranscripts;
```

edited Dec 2 '19 at 6:50

Varun Rajkumar

127 • 1 • 2 • 10

answered Sep 6 '16 at 9:27



Gajender Singh 831 ● 8 ● 10



I don't know of Google voice, but using the javaScript speech SpeechSynthesisUtterance, you can add a click event to the element you are reference to. eg:







```
const listenBtn = document.getElementById('myvoice');
listenBtn.addEventListener('click', (e) => {
    e.preventDefault();

    const msg = new SpeechSynthesisUtterance(
        "Hello, hope my code is helpful"
    );
    window.speechSynthesis.speak(msg);
});

    cbutton type="button" id='myvoice'>Listen to me</putton>

O Run code snippet

Copy snippet to answer

Expand snippet
```



answered Sep 9 at 23:28

iamnonso
11 • 3