

## Text to Speech

Description: This feature converts text to speech that is read back to the caller. this is useful for development or saying dynamic text that is difficult to pre-record. The feature offers different options for voices.

Web.config settings:

```
<add key="DidNumber" value="1XXXXXXXXXX"/>
<add key="SmsApiBaseUrl"
value="https://api.didforsale.com/didforsaleapi/index.php/api/SMS/send/" />
<add key="ApiKey" value="Get from your DIDforsale account portal"/>
<add key="WebsitePath" value="http://yourdomain.com"/>
```

Coding Description:

DIDTextToSpeech\Controllers\TextToSpeechController.cs

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Configuration;
using System.Web.Mvc;

namespace DIDTextToSpeech.Controllers
{
    public class TextToSpeechController : Controller
    {
        // GET: TextToSpeech
        public string didNumber =
WebConfigurationManager.AppSettings["DidNumber"].ToString();
        public string apiKey = WebConfigurationManager.AppSettings["ApiKey"].ToString();
        public string voiceApiBaseUrl =
WebConfigurationManager.AppSettings["VoiceApiBaseUrl"].ToString();
        public string websitePath =
WebConfigurationManager.AppSettings["WebsitePath"].ToString();
        public ActionResult Index()
        {
            return View();
        }
        [HttpPost]
        public ActionResult Index(string phonenumber, string text)
        {
            Dictionary<string, string> values = new Dictionary<string, string>();
            values.Add("from", didNumber);
            values.Add("to", phonenumber);
            values.Add("apikey", apiKey);
            // if your project is live use url domain/TextToSpeech/Speech
```

```

        values.Add("url", websitePath + Url.Action("Speech", new { text = text }));
        var result = HttpPostRequest(voiceApiBaseUrl, values);
        ViewBag.Result = result;
        return View();
    }
    public ActionResult Speech(string text)
    {
        string response = "<Response><Say>" + text + "</Say></Response>";
        return this.Content(response, "text/xml");
    }
    public string HttpPostRequest(string url, Dictionary<string, string>
postParameters)
    {
        string result = string.Empty;

        //DIDforsale api only accept json data format
        ServicePointManager.SecurityProtocol = SecurityProtocolType.Ssl3 |
SecurityProtocolType.Tls12 | SecurityProtocolType.Tls11 | SecurityProtocolType.Tls;
        var httpRequest = (HttpRequest)WebRequest.Create(url);
        httpRequest.ContentType = "application/json";
        httpRequest.Method = "POST";
        using (var streamWriter = new
StreamWriter(httpRequest.GetRequestStream()))
        {
            var json = MyDictionaryToJson(postParameters);
            streamWriter.Write(json);
            streamWriter.Flush();
            streamWriter.Close();
        }
        var httpResponse = (HttpWebResponse)httpRequest.GetResponse();
        using (var streamReader = new StreamReader(httpResponse.GetResponseStream()))
        {
            result = streamReader.ReadToEnd();
        }
        return result;
    }
    public string MyDictionaryToJson(Dictionary<string, string> dict)
    {
        var entries = dict.Select(d => string.Format("\"{0}\": \"{1}\"", d.Key,
string.Join(",", d.Value)));
        return "{" + string.Join(",", entries) + "}";
    }
}
}

```

DIDTextToSpeech\Views\TextToSpeech\Index.cshtml

```

@{
    Layout = null;
}
<!DOCTYPE html>
<html>

```

```

<head>
  <meta name="viewport" content="width=device-width" />
  <title>Index</title>
</head>
<body>
  <form name="authform" method="post" action="/">
    <table id="phone">
      <tr>
        <td>Phonenumber:</td>
        <td><input type="text" name="phonenumber" maxlength="15" id="phonenumber"
/></td>
      </tr>
      <tr>
        <td>Message:</td>
        <td><input type="text" name="text" id="text" /></td>
      </tr>
    </table>
    <input type="Submit" name="callme" value="Call Me" />
  </form>
  <div>
    <span>
      @ViewBag.Result
    </span>
  </div>
</body>
</html>

```

User will initiate call from his home page by entering phone number and message. Our code will initiate the call to the provided number with message url(<http://yourdomain.com/TextToSpeech/Speech?text=message>). Url will play the message text in voice format to user.