

Outbound Call Conference with DIDforsale api.

Description: With DID, you can create conference calls . Launch as many simultaneous conferences as you need. Select a moderator, chose your hold music.

In Outbound call conference, participants can be added by calling from DID number.

By default, DID conference rooms enable a number of useful features:

1. Conferences do not start until at least two participants join.
2. While waiting, customizable background music is played.
3. When participants join and leave, notification sounds are played to inform the other participants.

Web.config settings:

```
<add key="DidNumber" value="1XXXXXXXXXX"/>
<add key="ApiKey" value="Get from your DIDforsale account portal"/>
<add key="WebsitePath" value="http://yourdomain.com"/>
<add key="VoiceApiBaseUrl"
value="https://api.didforsale.com/didforsaleapi/index.php/api/V2/CallTermination/" />
```

Coding Description:

DIDTextToSpeech\Controllers\OutboundController.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 DIDConferenceCall.Controllers
{
    public class OutboundController : Controller
    {
        // GET: Outbound
        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)
    {
        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("JoinConference"));
        var result = HttpPostRequest(voiceApiBaseUrl, values);
        ViewBag.Result = result;
        return View();
    }
    public ActionResult JoinConference()
    {
        string response = "<Response><Dial><Conference
startConferenceOnEnter=\"false\">My conference</Conference></Dial></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 httpWebRequest = (HttpWebRequest)WebRequest.Create(url);
        httpWebRequest.ContentType = "application/json";
        httpWebRequest.Method = "POST";
        using (var streamWriter = new
StreamWriter(httpWebRequest.GetRequestStream()))
        {
            var json = MyDictionaryToJson(postParameters);
            streamWriter.Write(json);
            streamWriter.Flush();
            streamWriter.Close();
        }
        var httpResponse = (HttpWebResponse)httpWebRequest.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) + "}";
    }
}

```

}

In this feature user will go to the HTML page and will add his phone number to start conference with DID number, after that he can add other numbers by entering in this page, all entered number will be connected to the call conference.