

Search and Buy DID Number.

Description: In this feature, designed code will search for DID Number, User will search for Country Code in which he want to buy a number then will search for phone numbers in that country.

After receiving a list of DID numbers user will select a DID number and buy the number.

Web.config settings:

```
<add key="ApiKey" value="Get from your DIDforsale portal account"/>
<add key="SearchDidNumberApi"
value="http://api.didforsale.com/didforsaleapi/index.php/api/v2/ListDID/" />
<add key="BuyDidNumberApi"
value="http://api.didforsale.com/didforsaleapi/index.php/api/v2/BuyDID/" />
```

Coding Description:

DID2FA.Controllers/[HomeController.cs](#)

```
using DID2FA.Models;
using Newtonsoft.Json;
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 DID2FA.Controllers
{
    public class HomeController : Controller
    {
        public string apiKey = WebConfigurationManager.AppSettings["ApiKey"].ToString();
        public string searchDidNumberApi =
WebConfigurationManager.AppSettings["SearchDidNumberApi"].ToString();
        public string buyDidNumberApi =
WebConfigurationManager.AppSettings["BuyDidNumberApi"].ToString();
        public ActionResult SearchDID()
        {
```

```

        ListDID listDID = new ListDID();
        string searchResult = string.Empty;
        HttpRequest request = (HttpRequest)WebRequest.Create(searchDidNumberApi
+ apiKey + "?npa=216&nxx=373&show=10");
        var asa = Request.Params[""];
        using (HttpResponse response = (HttpResponse)request.GetResponse())
        using (Stream stream = response.GetResponseStream())
        using (StreamReader reader = new StreamReader(stream))
        {
            searchResult = reader.ReadToEnd();
        }
        var jsonResult = JsonConvert.DeserializeObject<ListDID>(searchResult);
        if (jsonResult.status == "success")
        {
            listDID = jsonResult;
        }

        return View(listDID);
    }
    [HttpPost]
    public ActionResult BuyDID(numbers number)
    {
        string result = string.Empty;
        if (!string.IsNullOrEmpty(number.number))
        {
            HttpRequest request =
(HttpHttpRequest)WebRequest.Create(buyDidNumberApi + apiKey + "?ratecenter=" +
number.ratecenter + "&state=" + number.state + "&did=" + number.number + "&reference_id="
+ number.reference_id + "&didtype=" + number.metered);
            using (HttpResponse response = (HttpResponse)request.GetResponse())
            using (Stream stream = response.GetResponseStream())
            using (StreamReader reader = new StreamReader(stream))
            {
                result = reader.ReadToEnd();
            }
        }
        return Json(result);
    }
}
}
}

```

Get values from web.config

```

public string apiKey = WebConfigurationManager.AppSettings["ApiKey"].ToString();
public string searchDidNumberApi =
WebConfigurationManager.AppSettings["SearchDidNumberApi"].ToString();
public string buyDidNumberApi =
WebConfigurationManager.AppSettings["BuyDidNumberApi"].ToString();

```

Search DID Number List Code:

```

ListDID listDID = new ListDID();
string searchResult = string.Empty;

```

```

+ apiKey + "?npa=216&nxx=373&show=10");
var asa = Request.Params[""];
using (HttpWebResponse response = (HttpWebResponse)request.GetResponse())
using (Stream stream = response.GetResponseStream())
using (StreamReader reader = new StreamReader(stream))
{
    searchResult = reader.ReadToEnd();
}
var jsonResult = JsonConvert.DeserializeObject<ListDID>(searchResult);
if (jsonResult.status == "success")
{
    listDID = jsonResult;
}

return View(listDID);

```

DID2FA\Views\Home\SearchDID.cshtml

```

@{
    ViewBag.Title = "Home Page";
}
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>

@model DID2FA.Models.ListDID
<div class="jumbotron">
    @using (Html.BeginForm("/"))
    {
        <div class="row">
            <div class="col-md-8">
                <h2>Available DID Numbers for npa=216 and nxx=373</h2>
            </div>
            <div class="col-lg-offset-10">
                <input type="submit" class="btn btn-block btn-default" name="Buy"
value="Buy" id="buydid" />
            </div>
        </div>

        <table class="table">
            <thead>
                <tr>
                    <th>#</th>
                    <th>Number</th>
                    <th>Ratecenter</th>
                    <th>State</th>
                    <th>RefId</th>
                    <th>Type</th>
                </tr>
            </thead>
            <tbody>
                @foreach (var item in Model.numbers)
                {
                    <tr>
                        <td>

```

```

        <input type="checkbox" class="selected" />
    </td>
    <td class="number">
        @Html.DisplayFor(modelItem => item.number)
    </td>
    <td class="ratecenter">
        @Html.DisplayFor(modelItem => item.ratecenter)
    </td>
    <td class="state">
        @Html.DisplayFor(modelItem => item.state)
    </td>
    <td class="reference_id">
        @Html.DisplayFor(modelItem => item.reference_id)
    </td>
    <td class="metered">
        @Html.DisplayFor(modelItem => item.metered)
    </td>
</tr>
    }
</tbody>
</table>
}
</div>
<script>
$(document).on("click", 'input[type="checkbox"]', function () {
    $(document).find('input[type="checkbox"]').not(this).each(function () {
        $(this).removeAttr("checked");
    })
})
$('#buydid').click(function (e) {
    e.preventDefault();
    debugger;
    var chekedRow = $(document).find('input[type="checkbox"]:checked').closest('tr');
    var _number = chekedRow.find('.number').text().trim();
    var ratecenter = chekedRow.find('.ratecenter').text().trim();
    var state = chekedRow.find('.state').text().trim();
    var reference_id = chekedRow.find('.reference_id').text().trim();
    var metered = chekedRow.find('.metered').text().trim();
    if (metered = "y")
    {
        metered = "metered";
    }

    var number = { number: _number, ratecenter: ratecenter, state: state,
reference_id: reference_id, metered: metered }
    $.ajax({
        method: 'POST',
        url: '/',
        data: { number: number },
        datatype: "JSON",
        success: function (result) {
            alert(result);
        }
    })
});
</script>

```

Select a number from list and click to buy button. And post the data using ajax to controller.

DID2FA\Models>ListDID.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace DID2FA.Models
{
    public class ListDID
    {
        public string status { get; set; }
        public List<numbers> numbers { get; set; }
        public string code { get; set; }
    }
    public class numbers
    {
        public string number { get; set; }
        public string ratecenter { get; set; }
        public string state { get; set; }
        public string reference_id { get; set; }
        public string metered { get; set; }
        public string channelized { get; set; }
        public string flatrate { get; set; }
        public string metered_tier1 { get; set; }
        public string metered_tier2 { get; set; }
    }
}
}
```

Define DID search api response parameter as property in model.

```
//Buy DID Number
string result = string.Empty;
if (!string.IsNullOrEmpty(number.number))
{
    HttpWebRequest request =
(HttpWebRequest)WebRequest.Create(buyDidNumberApi + apiKey + "?ratecenter=" +
number.ratecenter + "&state=" + number.state + "&did=" + number.number + "&reference_id="
+ number.reference_id + "&didtype=" + number.metered);
    using (HttpWebResponse response = (HttpWebResponse)request.GetResponse())
    using (Stream stream = response.GetResponseStream())
    using (StreamReader reader = new StreamReader(stream))
    {
        result = reader.ReadToEnd();
    }
}
```