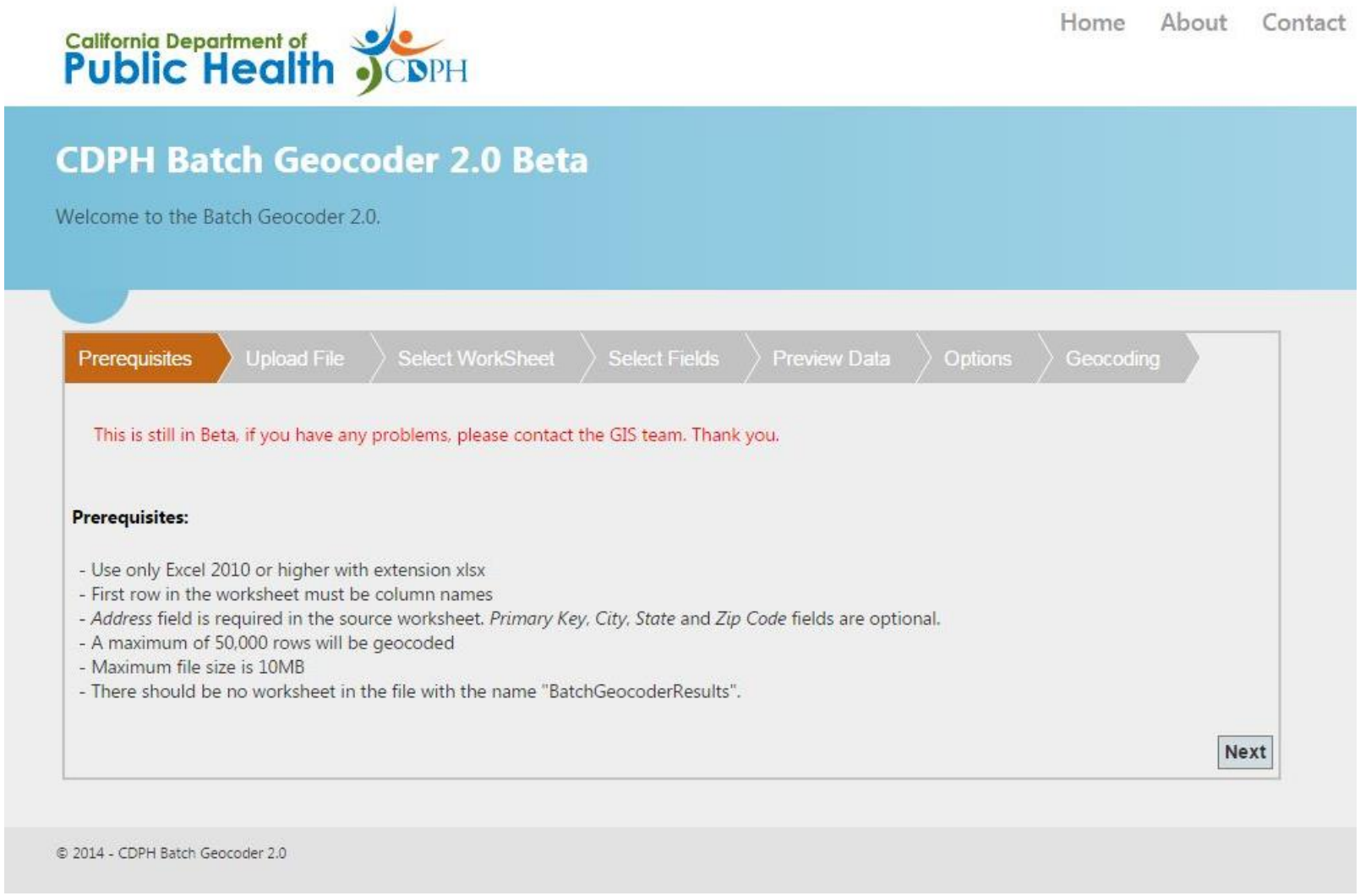
Developing a Browser geocoder using REST API and ASP.NET

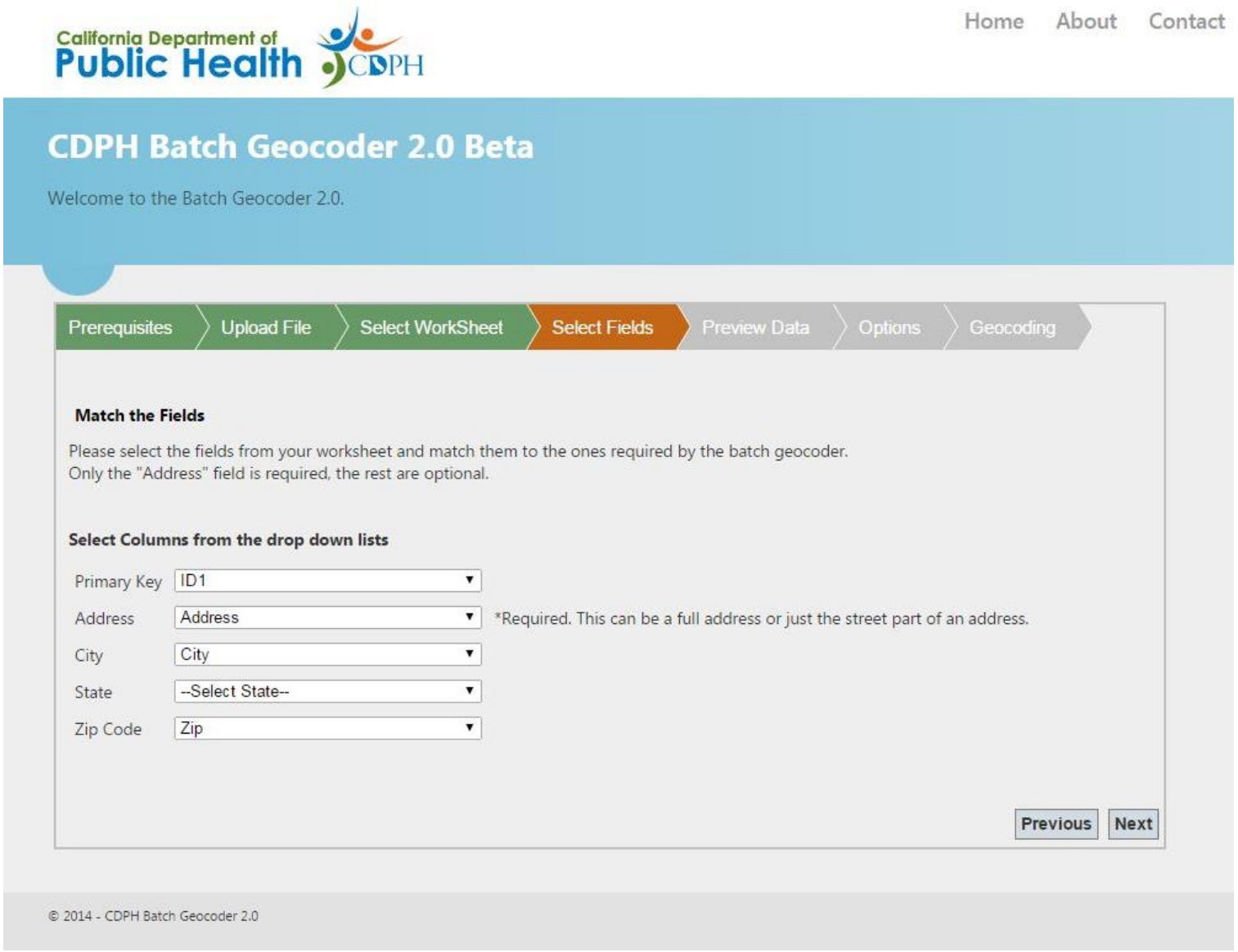
Bhavdeep S Sachdev

The Browser Geocoder

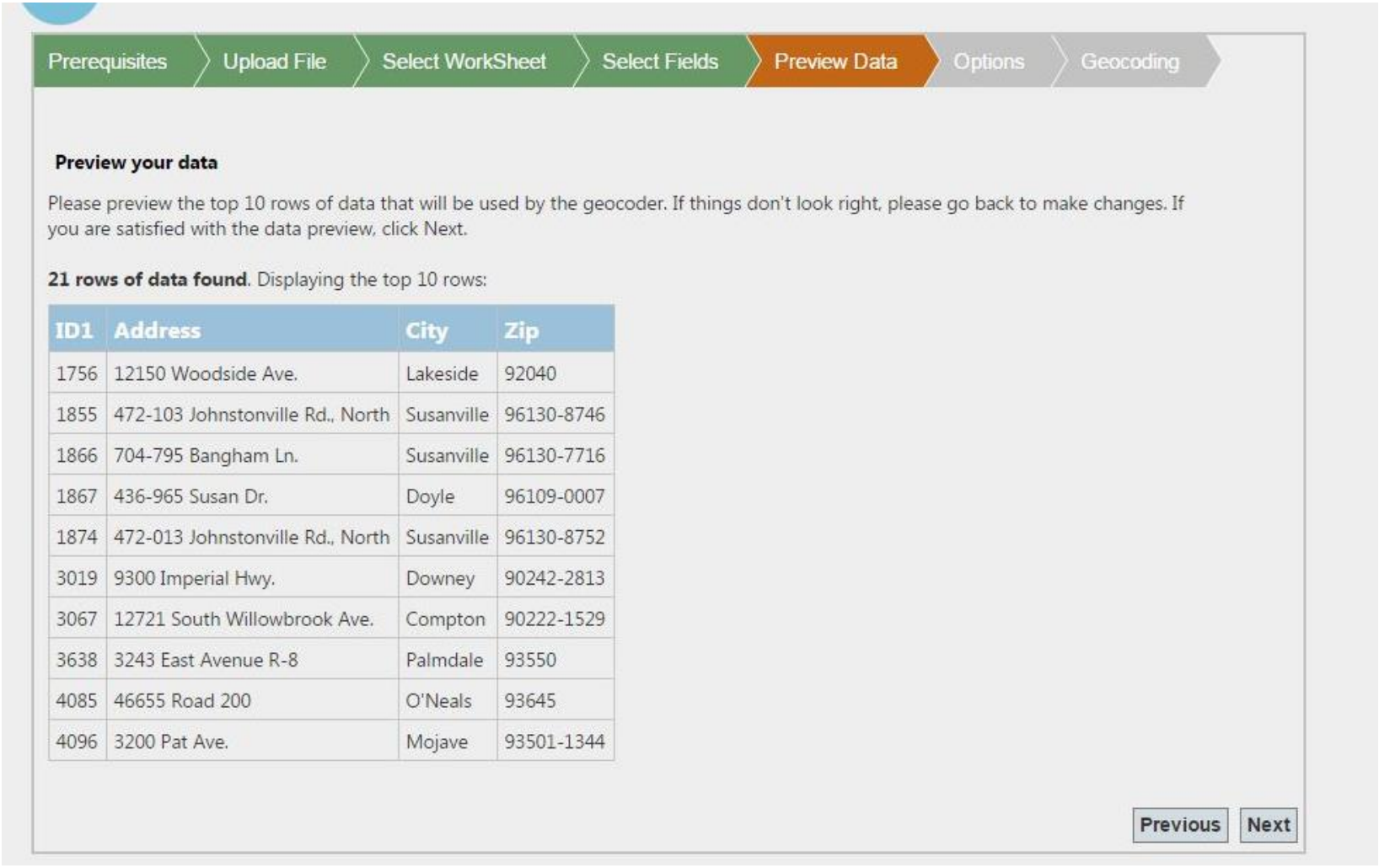


The Browser Geocoder

The Browser Geocoder

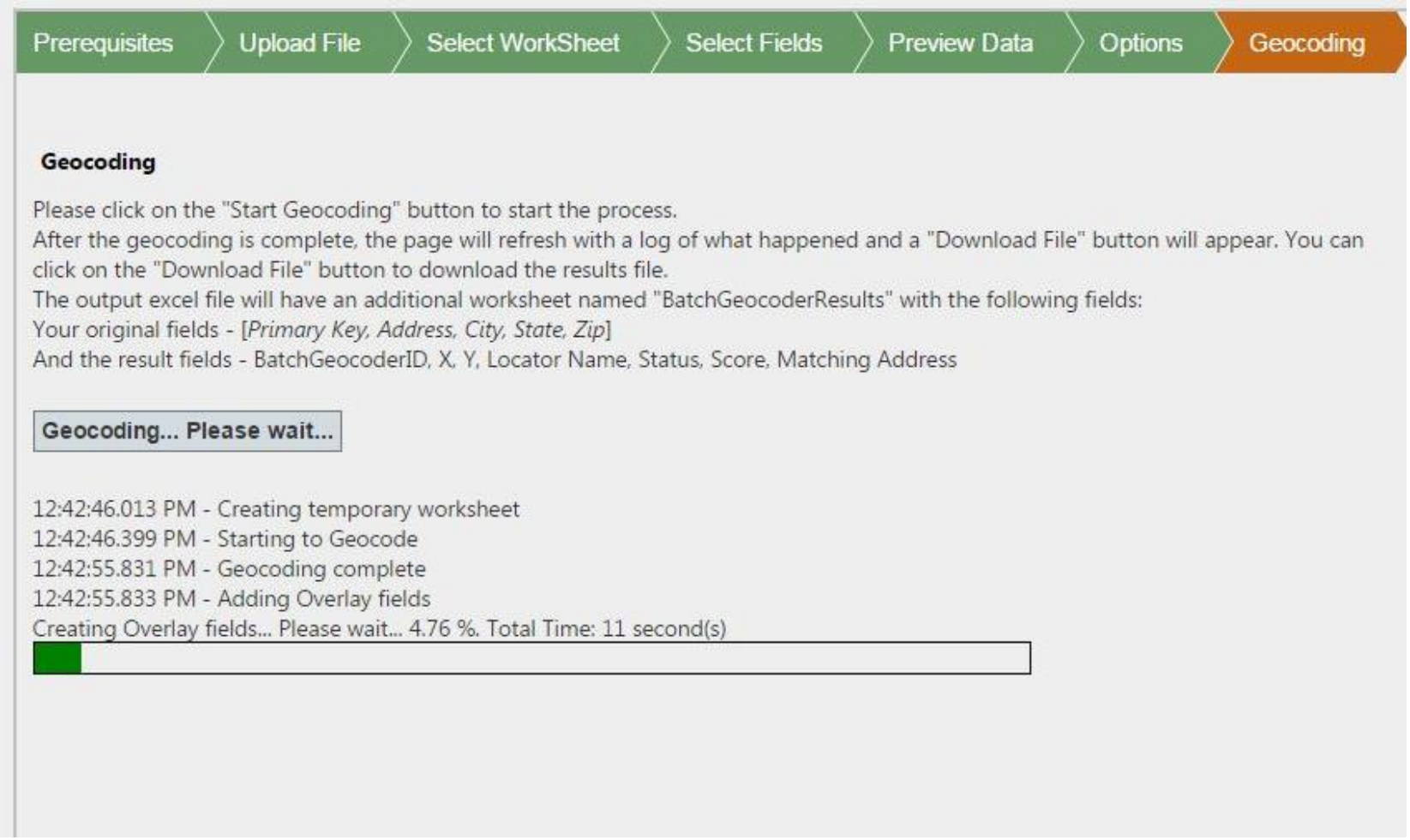


The Browser Geocoder

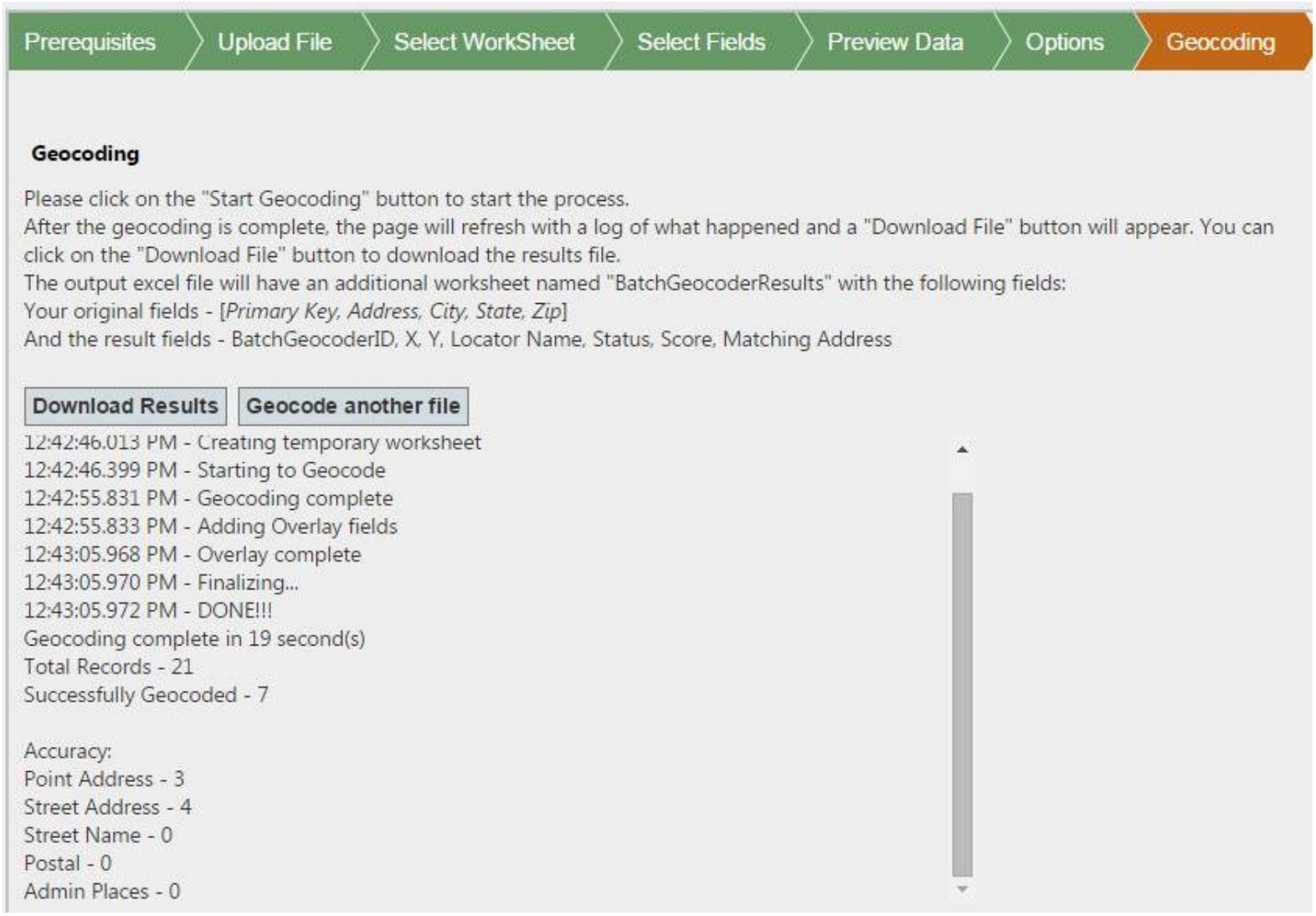


The Browser Geocoder

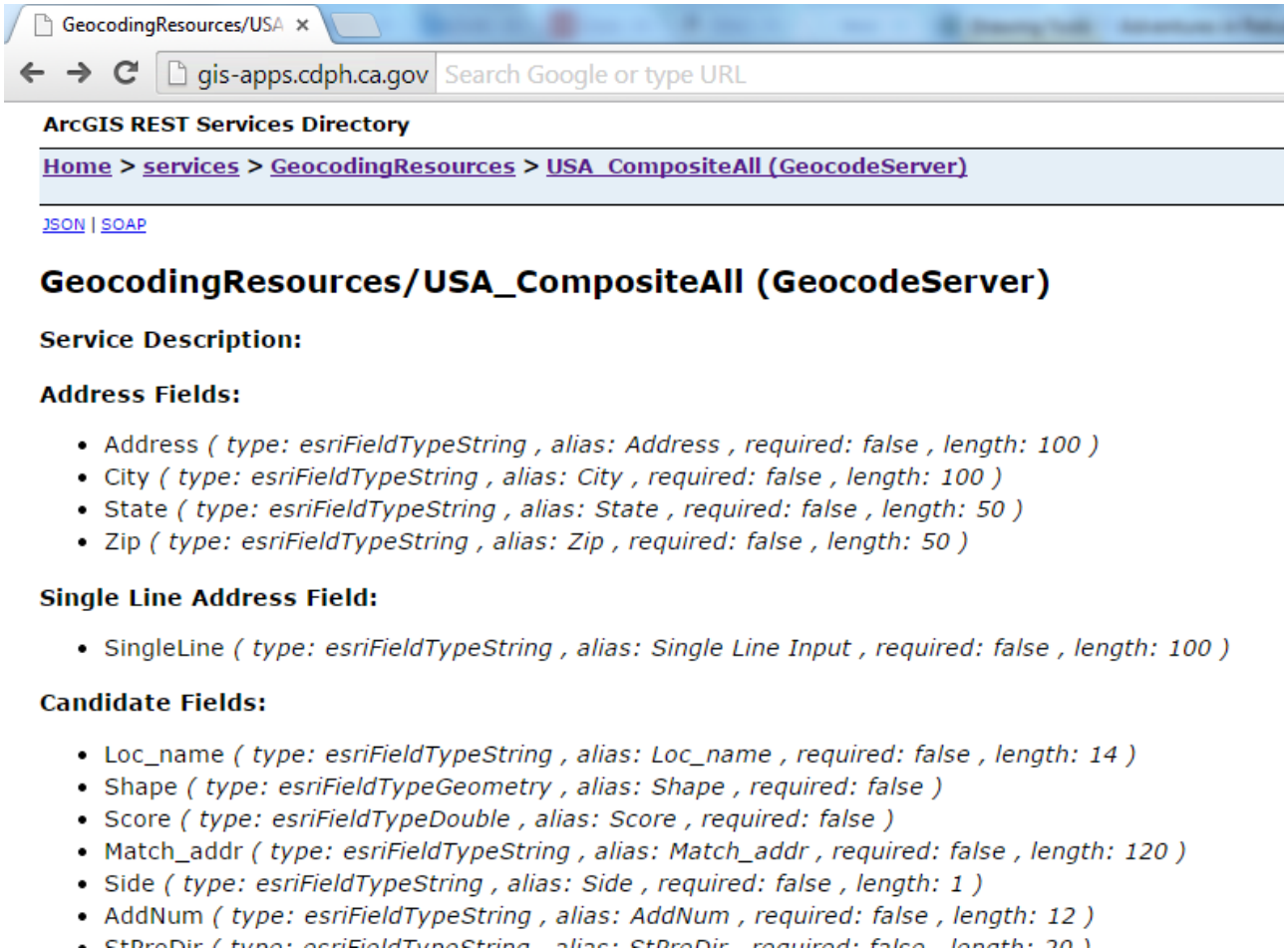
The Browser Geocoder



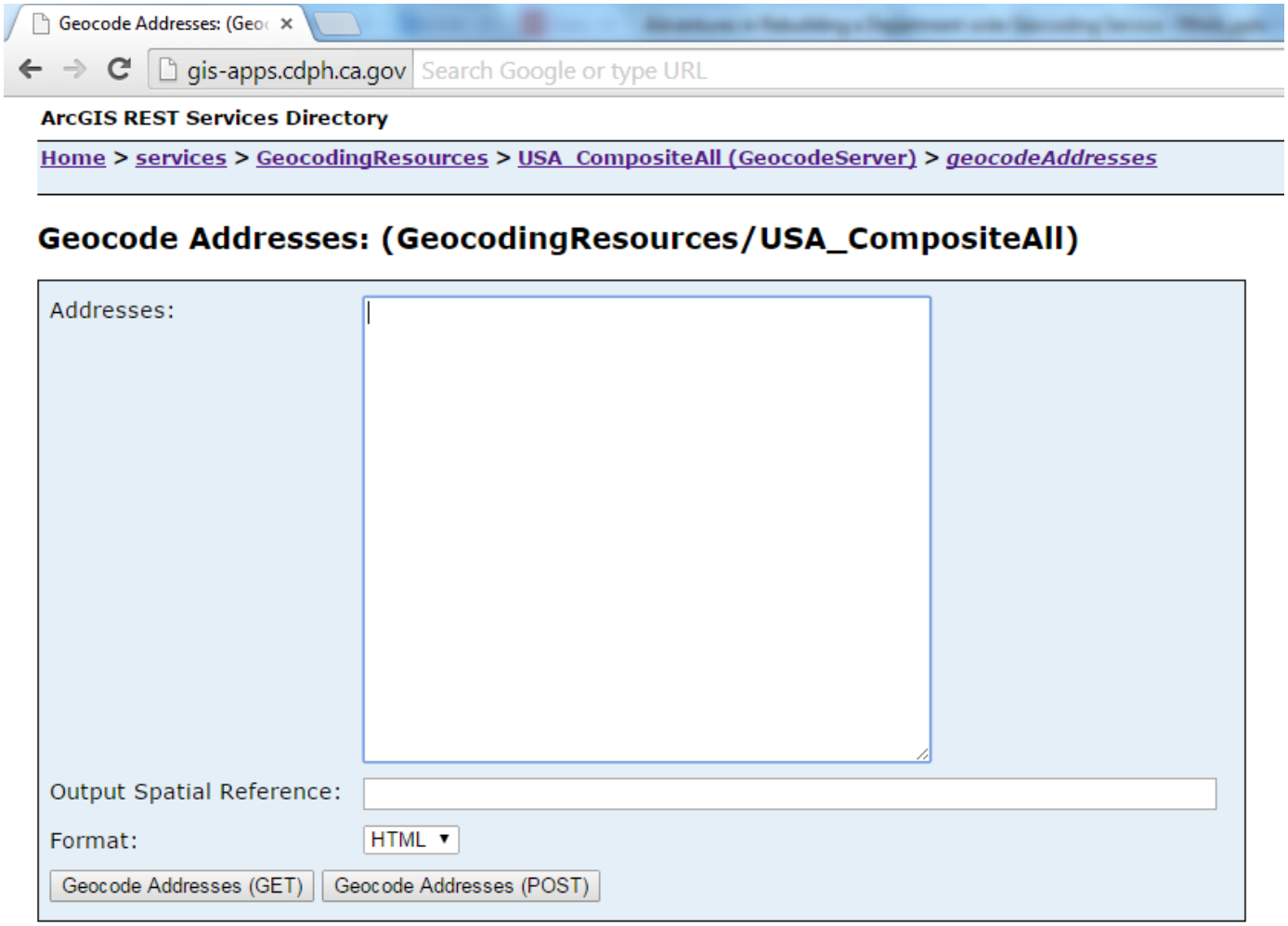
The Browser Geocoder



The REST Endpoint



The REST Endpoint



The REST Endpoint - json

```
//Sample JSON for Geocoding
StringBuilder sb = new StringBuilder();
sb.AppendLine(@"{ ""records"": [");
sb.AppendLine(@"   {");
sb.AppendLine(@"     ""attributes"": {");
sb.AppendLine(@"         ""OBJECTID"": 1,");
sb.AppendLine(@"         ""Address"": ""1616 Capitol Ave"",");
sb.AppendLine(@"        ""City"": ""Sacramento"",");
sb.AppendLine(@"        ""State"": ""CA"",");
sb.AppendLine(@"        ""Zip"": ""95814""");
sb.AppendLine(@"     }");
sb.AppendLine(@"   },");
sb.AppendLine(@"   {");
sb.AppendLine(@"     ""attributes"": {");
sb.AppendLine(@"         ""OBJECTID"": 2,");
sb.AppendLine(@"         ""Address"": ""4000 J Street"",");
sb.AppendLine(@"        ""City"": ""Sacramento"",");
sb.AppendLine(@"        ""State"": ""CA""");
sb.AppendLine(@"     }");
sb.AppendLine(@"   }");
sb.AppendLine(@"]");
sb.AppendLine(@"}");
txtGeocoderJSONInput.Text = sb.ToString();
```

The REST Endpoint – request/response

```
//Build Post data
StringBuilder postData = new StringBuilder();
postData.Append("addresses=" + HttpUtility.UrlEncode(txtGeocoderJSONInput.Text) + "&");
postData.Append("outSR=4326&");
postData.Append("f=pjson");

string postDataString = postData.ToString();

//Send the Post request
HttpWebRequest webRequest = (HttpWebRequest)WebRequest.Create(url);
webRequest.Method = "POST";
webRequest.ContentType = "application/x-www-form-urlencoded";
webRequest.ContentLength = postDataString.Length;

using (StreamWriter requestWriter = new StreamWriter(webRequest.GetRequestStream()))
{
    requestWriter.Write(postDataString);
}

//Read the response
string responseData = string.Empty;

using (StreamReader responseReader = new StreamReader(webRequest.GetResponse().GetResponseStream()))
{
    responseData = responseReader.ReadToEnd();
}
```

The REST Endpoint - Output

Input JSON for Geocoding:


```
{ "records": [
   {
     "attributes": {
         "OBJECTID": 1,
         "Address": "1616 Capitol Ave",
        "City": "Sacramento",
        "State": "CA",
        "Zip": "95814"
     }
   },
   {
     "attributes": {
         "OBJECTID": 2,
         "Address": "4000 J Street",
        "City": "Sacramento",
        "State": "CA"
     }
   }
]
}
```


Submit to Geocoding Service

Output JSON from Geocoder:


```
{
 "spatialReference": {
  "wkid": 4326,
  "latestWkid": 4326
 },
 "locations": [
  {
   "address": "1616 Capitol Ave, Sacramento, Cal
   "location": {
    "x": -121.48594163399849,
    "y": 38.574701897575608
   },
   "score": 100,
   "attributes": {
    "ResultID": 1,
    "Loc_name": "PointAddress",
    "Status": "M",
    "Score": 100,
    "Match_addr": "1616 Capitol Ave, Sacramento,
    "Side": "R",
```

DEMO

Reading Excel Spreadsheets

```
oledbConn = new OleDbConnection(@"Provider=Microsoft.ACE.OLEDB.12.0; Data Source=" +
    hdnUploadedFileNameWithPath.Value + "; Extended Properties='Excel 12.0;HDR=YES;IMEX=1;';");
oledbConn.Open();

DataTable dt = oledbConn.GetOleDbSchemaTable(System.Data.OleDb.OleDbSchemaGuid.Tables, null);
bool bOutputSheetFound = false;
for (int i = 0; i < dt.Rows.Count; i++)
{
    string sTemp = dt.Rows[i]["TABLE_NAME"].ToString();
    if (sTemp == "BatchGeocoderResults")
        bOutputSheetFound = true;
}

if (bOutputSheetFound)
{
    UploadStatusLabel.Text = "You cannot have any worksheets with the name \"BatchGeocoderResult
    UploadStatusLabel.ForeColor = System.Drawing.Color.Red;
    oledbConn.Close();
    return;
}

ddlWorksheets.Items.Clear();

ddlWorksheets.DataSource = dt.DefaultView;
ddlWorksheets.DataTextField = "TABLE_NAME";
ddlWorksheets.DataValueField = "TABLE_NAME";
ddlWorksheets.DataBind();
ListItem li = new ListItem("--Select Worksheet--", "-1");
ddlWorksheets.Items.Insert(0, li);
```

Read columns and preview data

```
oledbConn.Open();

OleDbCommand cmd = new OleDbCommand(); ;
OleDbDataAdapter oleda = new OleDbDataAdapter();
DataSet ds = new DataSet();

cmd.Connection = oledbConn;
cmd.CommandType = CommandType.Text;
cmd.CommandText = "SELECT top 1 * FROM [" + ddlWorksheets.SelectedValue + "]";
oleda = new OleDbDataAdapter(cmd);
oleda.Fill(ds, "dsSlno");
```

```
sCommandText = "";
sCommandText = "SELECT top 10 ";
if (bIDColumn) sCommandText += "[" + ddlObjectID.SelectedValue + "], ";
sCommandText += "[" + ddlAddress.SelectedValue + "]";
if (bCityColumn) sCommandText += ", [" + ddlCity.SelectedValue + "]";
if (bStateColumn) sCommandText += ", [" + ddlState.SelectedValue + "]";
if (bZipColumn) sCommandText += ", [" + ddlZip.SelectedValue + "]";
sCommandText += " FROM [" + ddlWorksheets.SelectedValue + "]";
```

Define fields in Output Sheet

```
string sCommand = "Create table [BatchGeocoderResults] ";
sCommand += " (BatchGeocoderID number,";
if (ddlObjectID.SelectedValue != "-1")
    sCommand += " [" + ddlObjectID.SelectedValue + "] string,";

sCommand += " [" + ddlAddress.SelectedValue + "] string,";

if (ddlCity.SelectedValue != "-1")
    sCommand += " [" + ddlCity.SelectedValue + "] string,";
if (ddlState.SelectedValue != "-1")
    sCommand += " [" + ddlState.SelectedValue + "] string,";
if (ddlZip.SelectedValue != "-1")
    sCommand += " [" + ddlZip.SelectedValue + "] string,";
sCommand += " X number, Y number, [Locator Name] string, Status string, Score number,
for (int j = 0; j < selectedLI.Count; j++)
{
    sCommand += ",[" + selectedLI[j].Text + "] string";
}
```

Start Geocoding

```
AddToGeocodingLog("\n" + DateTime.Now.ToString("hh:mm:ss.fff tt") + " - Starting to Geocode");
```

```
sqlCmd.CommandType = CommandType.StoredProcedure;
sqlCmd.CommandText = "UpdateGeocodingLog";
sqlCmd.Parameters.Add("BatchID", SqlDbType.Int).Value = Convert.ToInt32(hdnBatchGeocoderID.Value);
sqlCmd.Parameters.Add("StringToAdd", SqlDbType.VarChar, 1000).Value = sStringToAdd;
```

```
    if (iTotalNumberOfRecords < 1000)
        iNumberOfRecordsToGeocodeAtOnce = 50;
    else if (iTotalNumberOfRecords < 5000)
        iNumberOfRecordsToGeocodeAtOnce = 100;
    else if (iTotalNumberOfRecords < 20000)
        iNumberOfRecordsToGeocodeAtOnce = 200;
```

Request/Response

```
// create the POST request
HttpWebRequest webRequest = (HttpWebRequest)WebRequest.Create(url);
webRequest.Method = "POST";
webRequest.ContentType = "application/x-www-form-urlencoded";
webRequest.ContentLength = postDataString.Length;

using (StreamWriter requestWriter2 = new StreamWriter(webRequest.GetRequestStream()))
{
    requestWriter2.Write(postDataString);
}

//  This actually does the request and gets the response back
HttpWebResponse resp = (HttpWebResponse)webRequest.GetResponse();

string responseData = string.Empty;

using (StreamReader responseReader = new StreamReader(webRequest.GetResponse().GetResponseStream()))
{
    // dumps the HTML from the response into a string variable
    responseData = responseReader.ReadToEnd();
}

dynamic outputObject = JsonConvert.DeserializeObject(responseData);

UpdateResultsSheet(outputObject);
```

Parsing the Response Object

```
for (int i = 0; i < outputObject.locations.Count; i++)
{

    cmd = new OleDbCommand();
    cmd.Connection = oledbConn;
    cmd.CommandType = CommandType.Text;
    cmd.CommandText = @"UPDATE [BatchGeocoderResults] SET"
        + " X = " + outputObject.locations[i].location.x + ", "
        + " Y = " + outputObject.locations[i].location.y + ", "
        + " [Locator Name] = '" + outputObject.locations[i].attributes.Loc_name + "', "
        + " Status = '" + outputObject.locations[i].attributes.Status + "', "
        + " Score = " + outputObject.locations[i].attributes.Score + ", "
        + " [Matching Address] = '" + outputObject.locations[i].attributes.Match_addr + "' "
        + " WHERE BatchGeocoderID = " + outputObject.locations[i].attributes.ResultID + "";
    cmd.ExecuteNonQuery();

    iSuccessfullyGeocoded++;
```

Add Overlay Output

```
using (StreamReader responseReader = new StreamReader(webRequestOverlay.GetResponse().GetResponseStream()
{
    // dumps the HTML from the response into a string variable
    responseDataOverlay = responseReader.ReadToEnd();
}
outputOverlayObject = JsonConvert.DeserializeObject(responseDataOverlay);

UpdateResultsSheetWithOverlayData(outputOverlayObject);


for (int i = 0; i < outputObject.results[0].value.features.Count; i++)
{
    string sSelectedOptions = "";
    for (int j = 0; j < selectedLI.Count; j++)
    {
        if (sSelectedOptions != "")
            sSelectedOptions += ",";
        string sTemp = outputObject.results[0].value.features[i].attributes[selectedLI[j].Value];
        sSelectedOptions += " [" + selectedLI[j].Text + "] = '" + sTemp.Replace("'", "''") + "' ";
    }

    cmd = new OleDbCommand();
    cmd.Connection = oledbConn;
    cmd.CommandType = CommandType.Text;
    cmd.CommandText = @"UPDATE [BatchGeocoderResults] SET"
                            + sSelectedOptions
                        + " WHERE BatchGeocoderID = " + outputObject.results[0].value.features[i].attributes.
    cmd.ExecuteNonQuery();

}
```

Final Output

D	E	F	G	H	I	J	K	L
City	Zip	X	Y	Locator Name	Status	Score	Matching Address	Census 2010 Tracts
ıkeside	92040	-116.9302927	32.85619511	USA_PointAddress	M	100	0 Woodside Ave, Lakeside, California, 9	06073016702
sanville	96130-8746	0	0			0		
sanville	96130-7716	0	0			0		
Doyle	96109-0007	0	0			0		
sanville	96130-8752	0	0			0		
owney	90242-2813	-118.1286025	33.9169532	USA_PointAddress	M	100	0 Imperial Hwy, Downey, California, 90	06037551800
ompton	90222-1529	-118.2322575	33.91725951	USA_StreetAddress	M	100	Willowbrook Ave, Compton, California	06037541400
ılmdale	93550	-118.0708835	34.56560323	USA_StreetAddress	M	100	3 E Avenue R 8, Palmdale, California, 93	06037910605
'Neals	93645	0	0			0		
Mojave	93501-1344	-118.1897983	35.05005902	USA_StreetAddress	M	100	3200 Pat Ave, Mojave, California, 93501	06029005900
roville	95965	0	0			0		
'Neals	93645	0	0			0		
mecula	92592	-117.0952345	33.49804217	USA_StreetAddress	M	100	Meadows Pkwy, Temecula, California,	06065043264
ockton	95207	0	0			0		
ockton	95215-9113	0	0			0		
Palo Alto	94303	0	0			0		
ıgton Beach	92647	-118.0127373	33.73405412	USA_PointAddress	M	100	Glen Dr, Huntington Beach, California, 9	06059099605
taluma	94954-2301	0	0			0		
ıta Rosa	95401-4035	0	0			0		
akdale	95361	0	0			0		
oveland	95321-9338	0	0			0		

Future Enhancements

- Run as a Windows service
- Publish as an asynchronous service
- Publish and consume other services on the REST endpoint

```
protected void End_Of_Presentation()
{
    Console.Write("Any Questions");
}
```

Contact

- Bhavdeep S Sachdev
- Bhavdeep.Sachdev@gmail.com

- Get the code:
- https://github.com/bhavdeep-sachdev/CDPHBrowserGeocoder