

## Cloud Comp. Assignment 2

### Introduction:

The WeatherCloud webapp is designed to store cities and get the temperature for those cities and your location city using get requests. It uses services from <https://developer.yahoo.com/yql> service for the weather data, "<http://ip-api.com/json>" service for current location, and Facebook Developer API for Facebook user data, specifically the user's name.

It's hosted in azure platform and is located at <http://weathercloud1.azurewebsites.net>

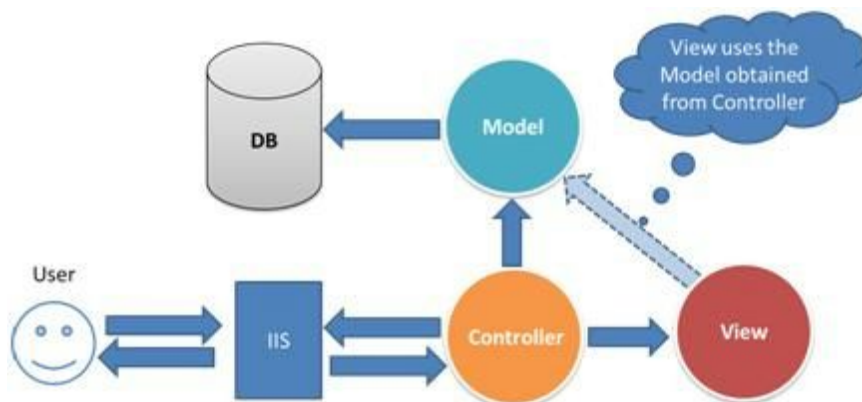
### Design:

The cloud website is designed using ASP.net(MVC), HTML/CSS and Javascript/JQuery. It uses SQL server database. It relies on Model View Controller architecture where is is separation between Data, Views, and the controllers. The server comprises of three methods: Index, add(city), and remove(city), in addition to getting location based on IP and "Login with Facebook" feature which uses "CheckAuthorization()" method.

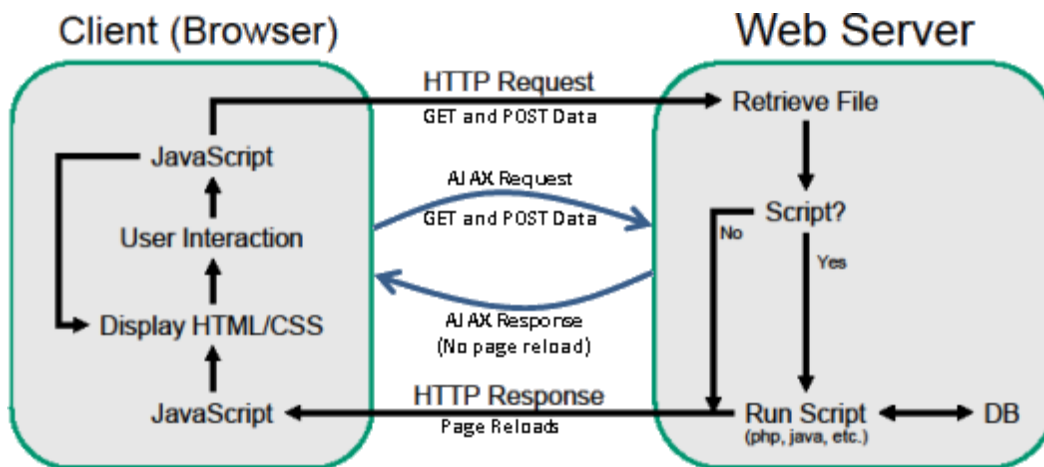
The interface is designed with AJAX technology meaning there will not be need for refresh when submitting action of add(city) or remove(city) to server. The client, after, listing cities from server database, will ask "<https://developer.yahoo.com/yql>" service, returning a JSON object, parsing it for the weather for each city listed, and display the temprature beside each city.

The Web app is improved by showing your current city based on your IP along with its temprature using the service "<http://ip-api.com/json>". It also uses Facebook API to getting the User's Facebook name, showing it beside the Welcome message, and then allowing the user to logout.

The Web App uses the default ASP.net Template, and its logic is design in the files "script.js", "HomeController.cs", the View (.html) files.



## MVC Architecture [1]



Client Request - Server interaction using Javascript [2]

## Instructions:

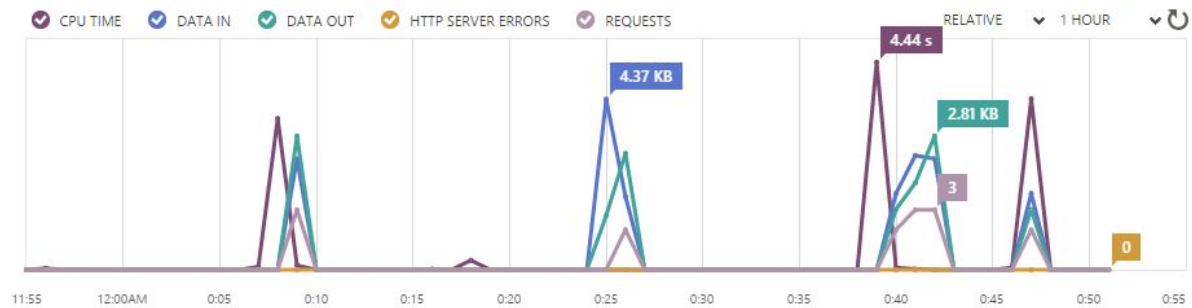
- Enter a city and Click “add” to add a city to store it in the database.
- Click “Login With Facebook” button to login using Facebook and Show your name beside welcoming message. You can logout by clicking “Logout” link.

## Scalability:

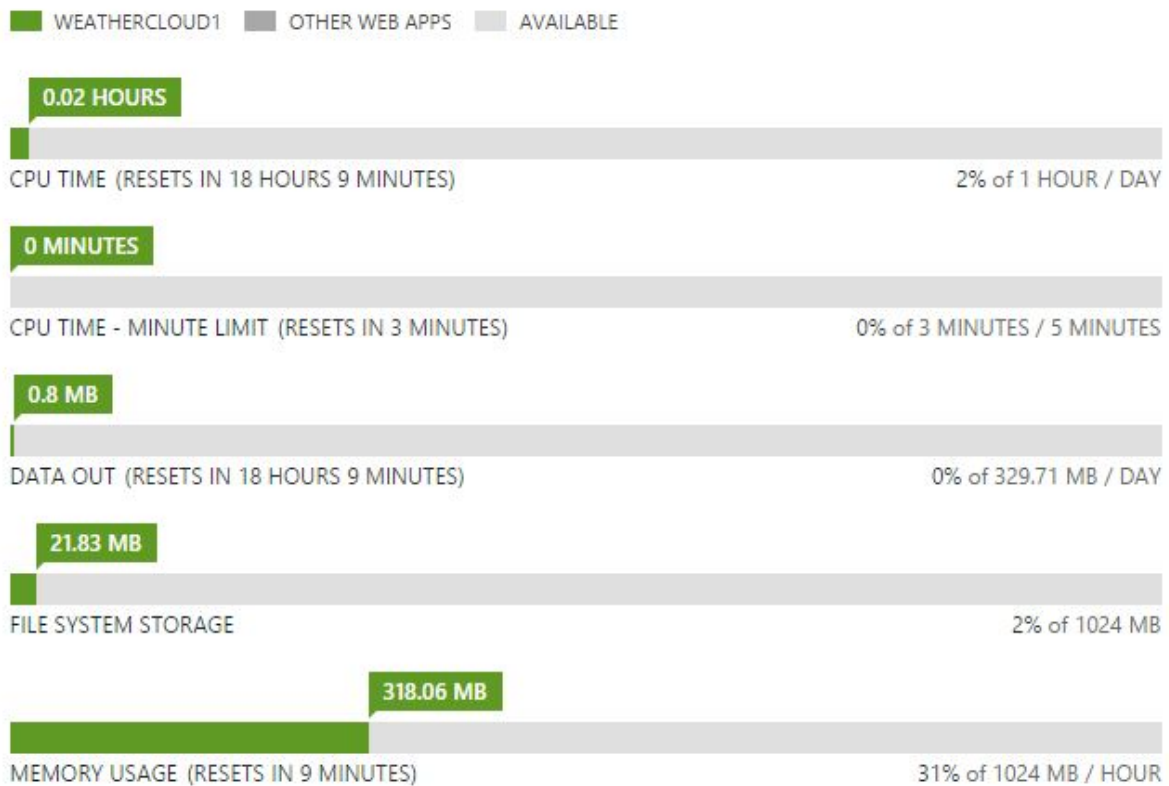
Microsoft Azure autoscaling feature which provides a very powerful tools to help scaling your application. You can have many instances of your Web App and scale down on them depending on rules you can set, such as the your site traffic. Moreover, you can have more instances running if you experience high traffic, but low number of instances if you have low traffic. It also ensures that your Web App is always running [3]

The free version of Azure, however, doesn't allow these powerful tools such as having more than one instance.

## Web App usage Data



Web App using in one hour



Web App Metrics in one hour

HTTP Request.jmx (C:\Users\100488982\Desktop\apache-jmeter-2.13\bin\examples\HTTP Request.jmx) - Apache JMeter (...)

File Edit Search Run Options Help

Test Plan  
Thread Group  
HTTP Request  
HTTP Request Defaults  
Response Assertion  
View Results in Table  
WorkBench

### View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename:  Browse...

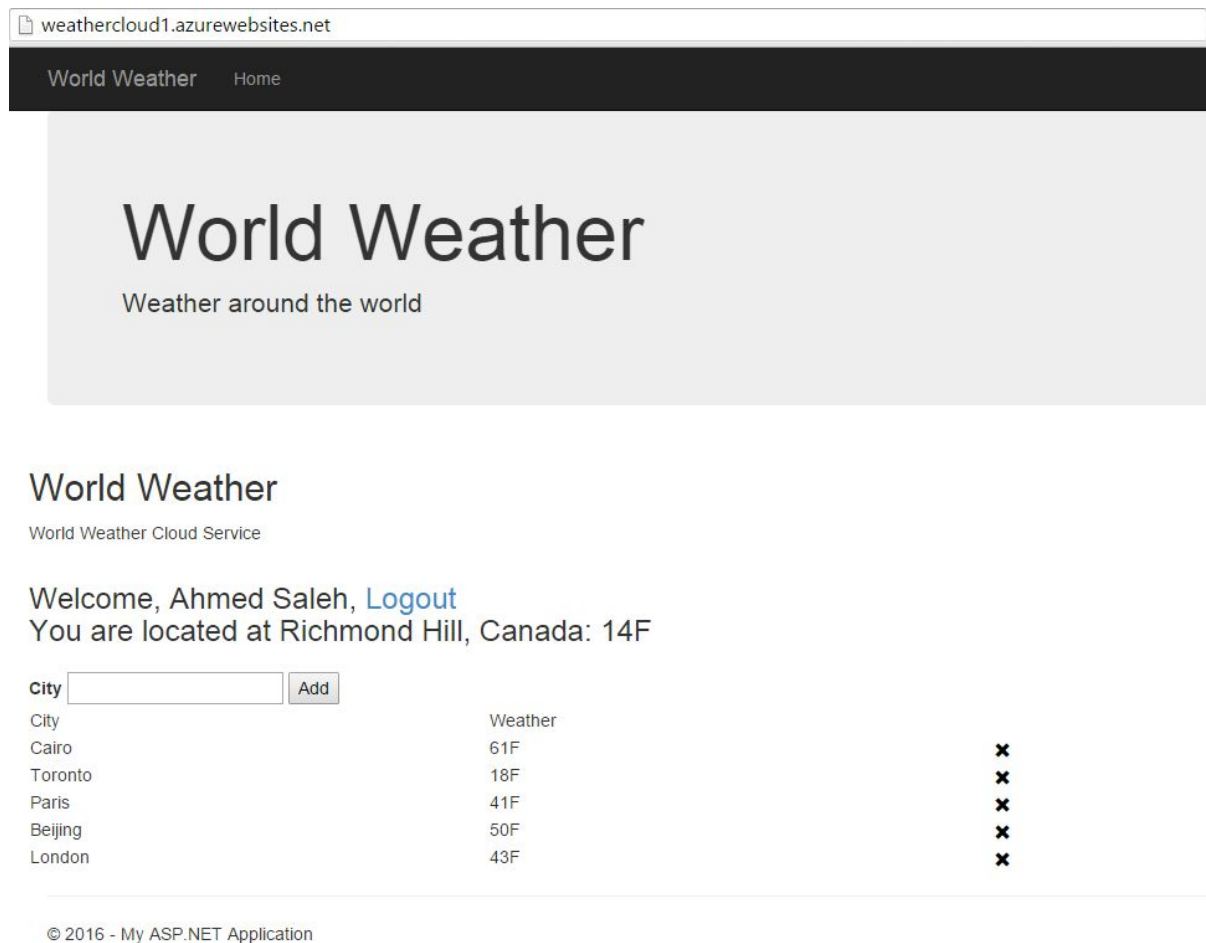
Log/Display Only: ☐ Errors ☐ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Latency	Connect Time(ms)
1	01:29:01.516	Thread Group 1-1	HTTP Request	0		946	0	0
2	01:29:01.529	Thread Group 1-2	HTTP Request	0		946	0	0
3	01:29:01.544	Thread Group 1-3	HTTP Request	0		946	0	0
4	01:29:01.549	Thread Group 1-4	HTTP Request	0		946	0	0
5	01:29:01.559	Thread Group 1-5	HTTP Request	0		946	0	0
6	01:29:01.569	Thread Group 1-6	HTTP Request	0		946	0	0
7	01:29:01.580	Thread Group 1-7	HTTP Request	0		946	0	0
8	01:29:01.594	Thread Group 1-8	HTTP Request	0		946	0	0
9	01:29:01.600	Thread Group 1-9	HTTP Request	0		946	0	0
10	01:29:01.610	Thread Group 1-...	HTTP Request	0		946	0	0
11	01:29:01.620	Thread Group 1-...	HTTP Request	0		946	0	0
12	01:29:01.630	Thread Group 1-...	HTTP Request	0		946	0	0
13	01:29:01.641	Thread Group 1-...	HTTP Request	0		946	0	0
14	01:29:01.651	Thread Group 1-...	HTTP Request	0		946	0	0
15	01:29:01.661	Thread Group 1-...	HTTP Request	0		946	0	0
16	01:29:01.673	Thread Group 1-...	HTTP Request	0		946	0	0
17	01:29:01.681	Thread Group 1-...	HTTP Request	0		946	0	0
18	01:29:01.691	Thread Group 1-...	HTTP Request	0		946	0	0
19	01:29:01.702	Thread Group 1-...	HTTP Request	0		946	0	0
20	01:29:01.712	Thread Group 1-...	HTTP Request	0		946	0	0
21	01:29:01.722	Thread Group 1-...	HTTP Request	0		946	0	0
22	01:29:01.734	Thread Group 1-...	HTTP Request	0		946	0	0
23	01:29:01.742	Thread Group 1-...	HTTP Request	0		946	0	0
24	01:29:01.752	Thread Group 1-...	HTTP Request	0		946	0	0
25	01:29:01.763	Thread Group 1-...	HTTP Request	0		946	0	0
26	01:29:01.774	Thread Group 1-...	HTTP Request	0		946	0	0
27	01:29:01.784	Thread Group 1-...	HTTP Request	0		946	0	0
28	01:29:01.794	Thread Group 1-...	HTTP Request	0		946	0	0
29	01:29:01.804	Thread Group 1-...	HTTP Request	0		946	0	0
30	01:29:01.814	Thread Group 1-...	HTTP Request	0		946	0	0
31	01:29:01.824	Thread Group 1-...	HTTP Request	0		946	0	0
32	01:29:01.834	Thread Group 1-...	HTTP Request	0		946	0	0
33	01:29:01.844	Thread Group 1-...	HTTP Request	0		946	0	0
34	01:29:01.854	Thread Group 1-...	HTTP Request	0		946	0	0

☐ Scroll automatically? ☐ Child samples? No of Samples 100 Latest Sample 0 Average 0 Deviation 0

JMeter Test of main Page with 100 threads, and Response Assertion code of 200.

## Screen Shots:



## References:

- [1] M.Sukesh, "WebForms vs. MVC," - CodeProject. [Online]. Available at: <http://www.codeproject.com/articles/528117/webforms-vs-mvc>. [Accessed: 11-Feb-2016].
- [2] "About This Book," *JavaScript for Sheridan Students*. [Online]. Available at: <http://javascript.sheridanc.on.ca/>. [Accessed: 11-Feb-2016].
- [3] "4: Autoscaling and Microsoft Azure," *4: Autoscaling and Microsoft Azure*. [Online]. Available at: [https://msdn.microsoft.com/en-us/library/hh680945\(v=pandp.50\).aspx](https://msdn.microsoft.com/en-us/library/hh680945(v=pandp.50).aspx). [Accessed: 03-Mar-2016].