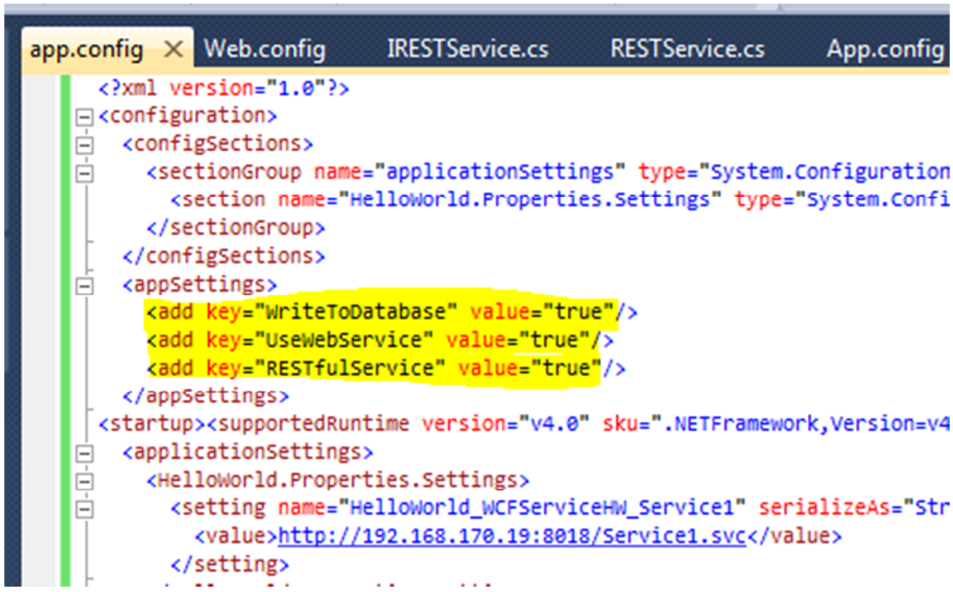
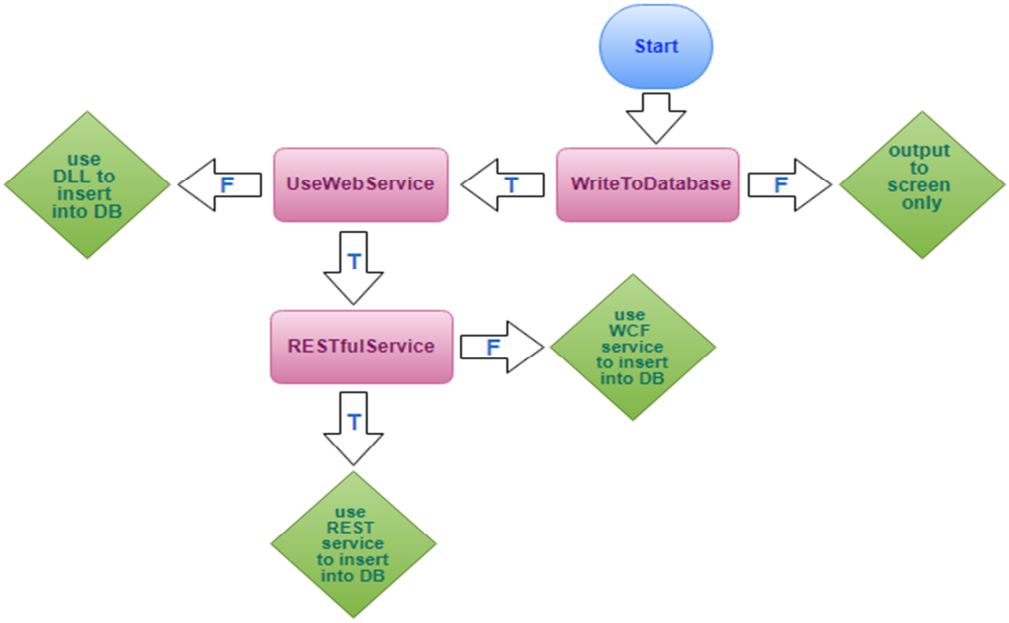
There are 5 projects in this solution:

1. Console app HelloWorld
2. DllTest
3. RESTFulDemo
4. WcfService1
5. UnitTestHelloWorld

I added 3 different APIs for this console app to use. There are options to select which API to use, or not to use any API, in app.config of HelloWorld project.

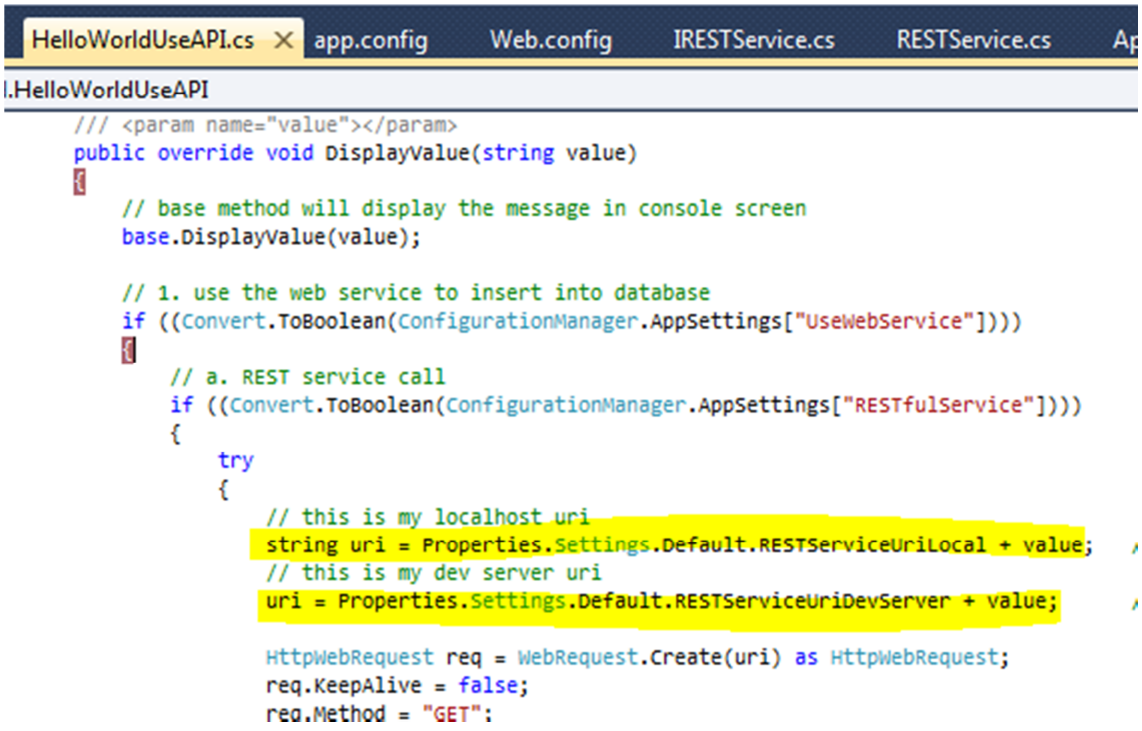


Flowchart below shows how to set the key values in app.config to call different APIs

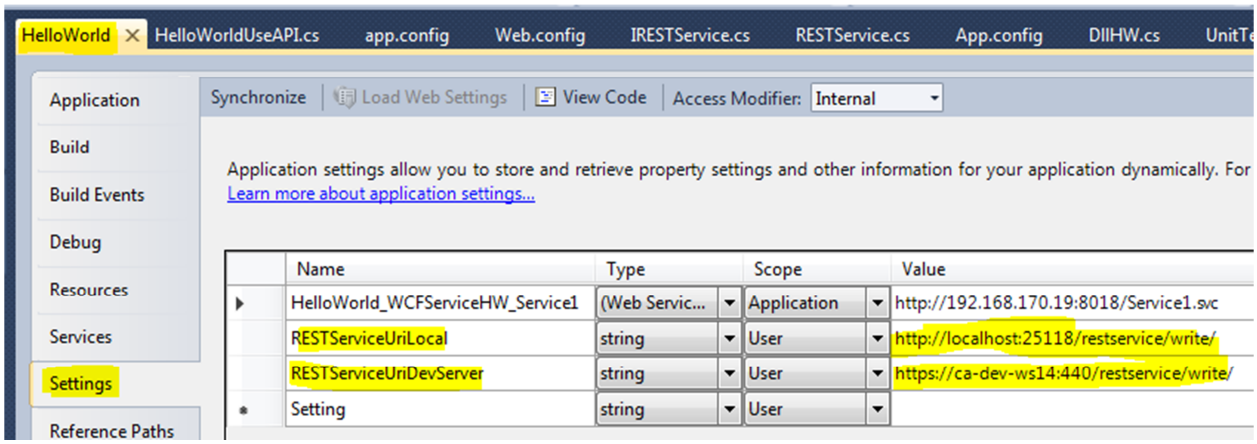


Some notes here:

1. In HelloWorldUseAPIs.cs, when the code calls REST service:

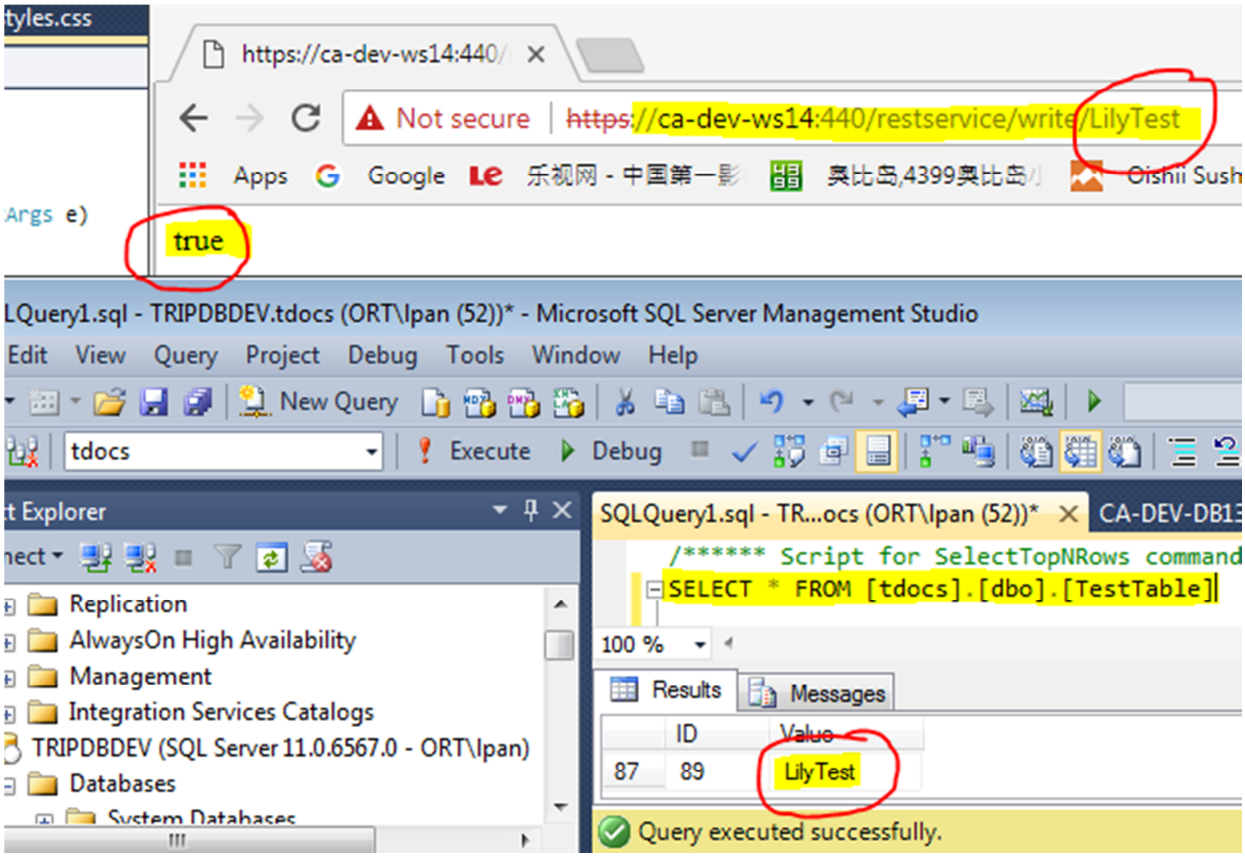


The uri values for localhost and dev server are set in:



These uri's are for my localhost and my dev server, please change to your uri values to test.
My dev server is internal so you will not be able to access it from internet.

A browser test result looks like this:
https://ca-dev-ws14:440/restservice/write/LilyTest has inserted the value into database table and returned “true” in the browser.
“Not secure” warning is because I made a self-signed certificate on my dev server which is not trusted by SSL.
This service works for HelloWorld console app too.

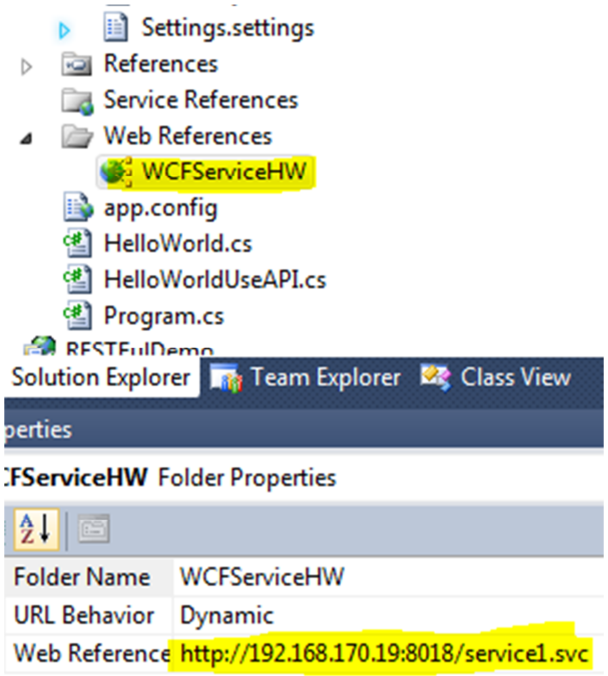


2. In HelloWorldUseAPI.cs, when the code calls WCF service, please comment one or the other, to try either the local reference or the web reference.

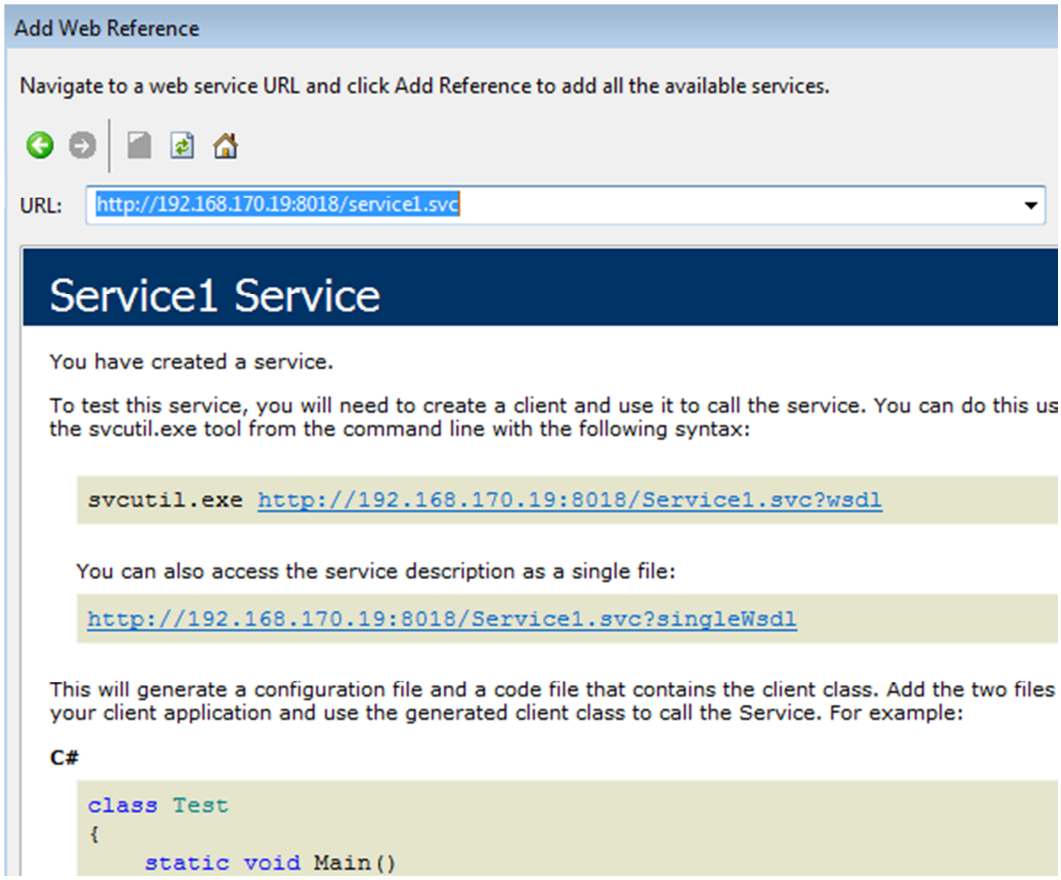
```
// b. WCF service call
else
{
    try
    {
        // when the service is added as a local reference
        WcfService1.Service1 service = new WcfService1.Service1();
        bool result = service.WriteValue(value);
        Console.WriteLine("Inserted into database by WCFService: " + result);
        Thread.Sleep(5000);

        // when the service is added as a web reference
        WCFServiceHW.Service1 svc = new WCFServiceHW.Service1();
        bool writeValueResult = false;
        bool writeValueResultSpecified = true;
        svc.WriteValue(value, out writeValueResult, out writeValueResultSpecified);
        Console.WriteLine("Inserted into database by WCFService: " + writeValueResult);
        Thread.Sleep(5000);
    }
```

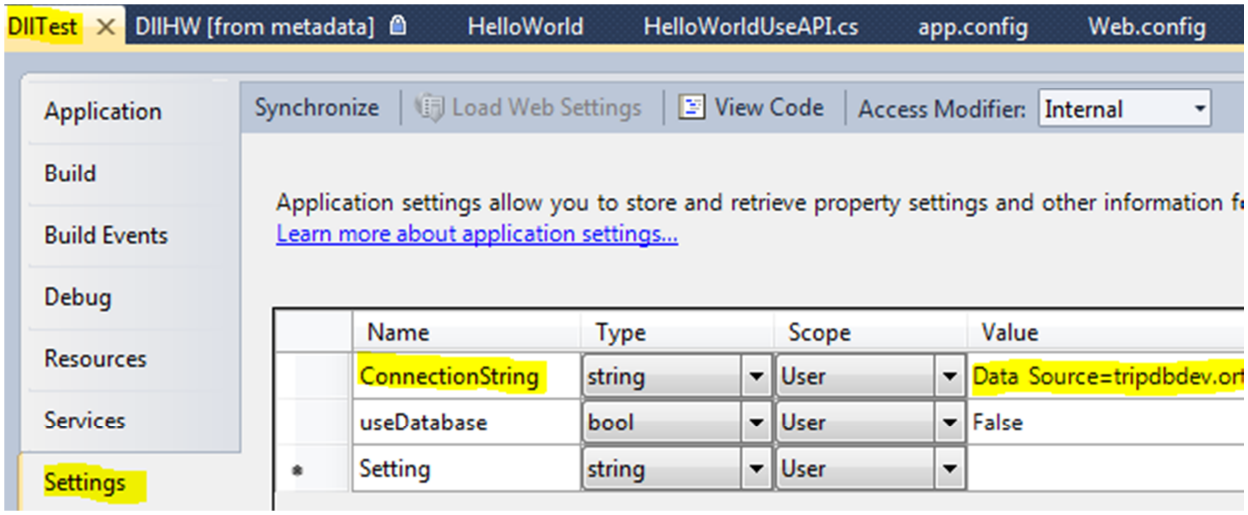
When you try the web reference, please not to use my web reference. Because I published it on my internal dev server which will not be connected from outside.



Please publish the WCF service on your server, and then go to “Add Web Reference” with your own service link.



3. The connection string for the DLL is set in here:
   I will delete my connection string value so you will put your own value in there.

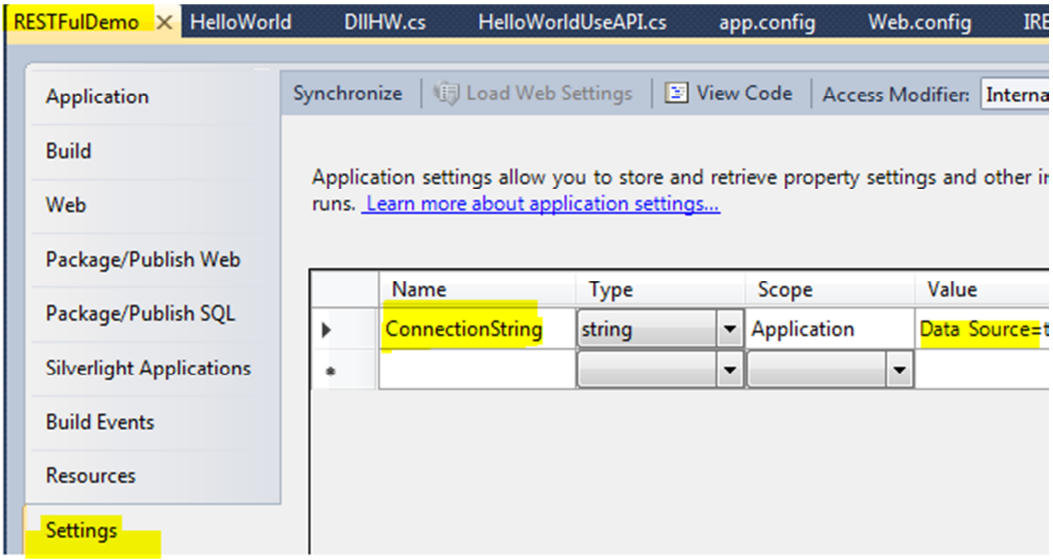


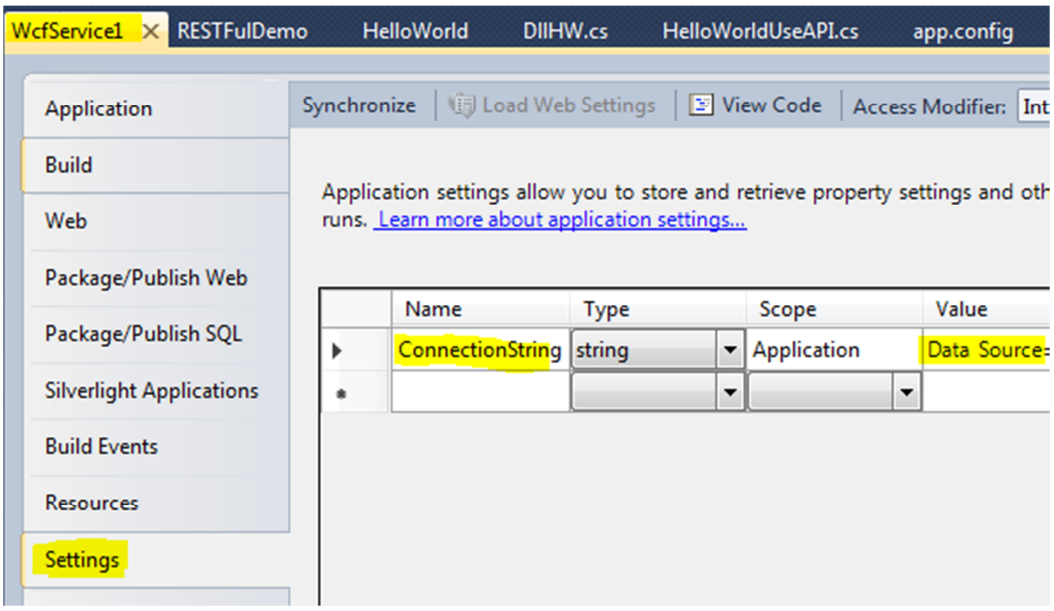
In DllHW.cs, if you would like to test the database insert method, this is where you put your own query.

I have created a TestTable in dev DB and it works with my code.

In the DLL I also added GetValue( ) method. I thought you also wanted to have options of either getting value from the database or just hard coding “Hello World”. Then I realized you just asked for writing to database, but not reading from database. Anyways I added the option of reading from database too, and it’s turned off for now in the config file: useDatabase is set to False.

4. I will also delete the connection string in RESTFulDemo and in WcfService1. Please add your own connection strings in there if you will test the code.





5. I did unit test for the DLL, to check if the return value is “Hello World”, and the database insert method. I did not do the unit test for the REST service and WCF serve, because the only method is to insert into the database, and they will be the same as the insert test for the DLL.

6. So I think the DLL is easy to use, but it has to be added locally, and if there are changes in the DLL then we will have to rebuild the DLL, also the application that uses the DLL will have to re reference the DLL and rebuild too. The WCF service can be consumed from the server, and it’s stable, but if there are changes in the service, the client app will also have to re reference and rebuild the code. Also we have to make sure all the configurations match on the server side and on client side, otherwise it will not work. Especially when SSL is enabled on the server, the configurations could be a little complicated for the client app. The REST service is easier for client app to use because we don’t need to add reference, if there are changes in the service, as long as the uri is not changed, the client side doesn’t have to change anything.