# Objective:

The purpose of this document is to demonstrate how to **migrate a simple on premise IIS based website to Azure Web Role (Cloud Service)**.

# Prerequisites:

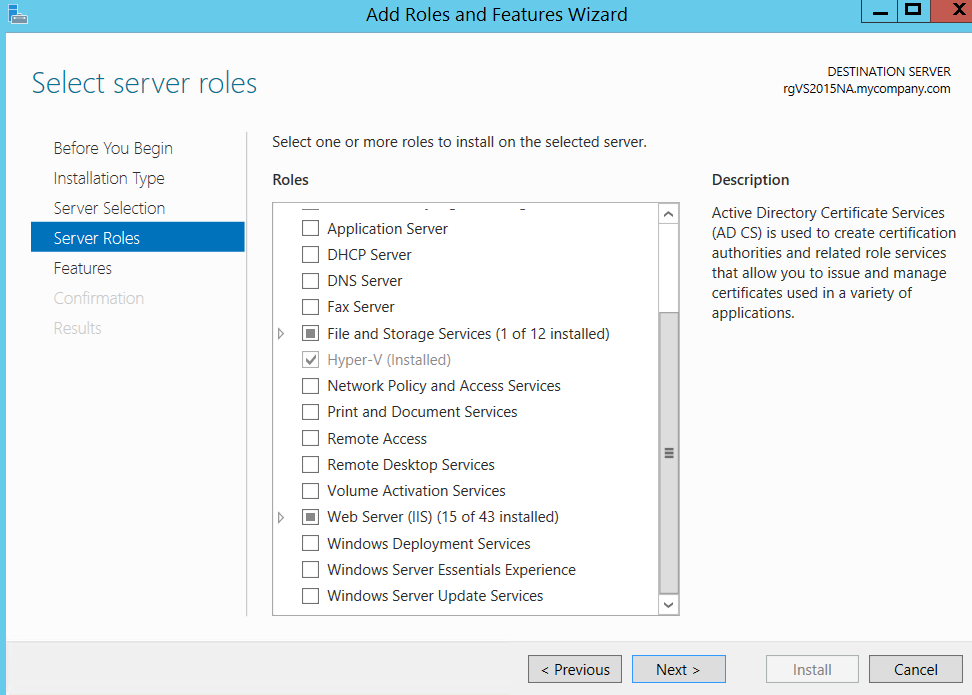
* Visual Studio 2015 with IIS
* Valid Azure Subscription
* This demonstration uses an already built ASP.NET web application named CloudShop. This app is located in the following zip **DemoApp\_CloudShop.zip**..
* This demonstration uses an already built SQL Server database named Adventure Works. You should configure this Database either on premises or in the Cloud. See docs **Demo-MigrateToAzureSqlDB** **or Demo-MigrateToAzureSqlVM** to learn how to setup the database.

# Steps Overview:

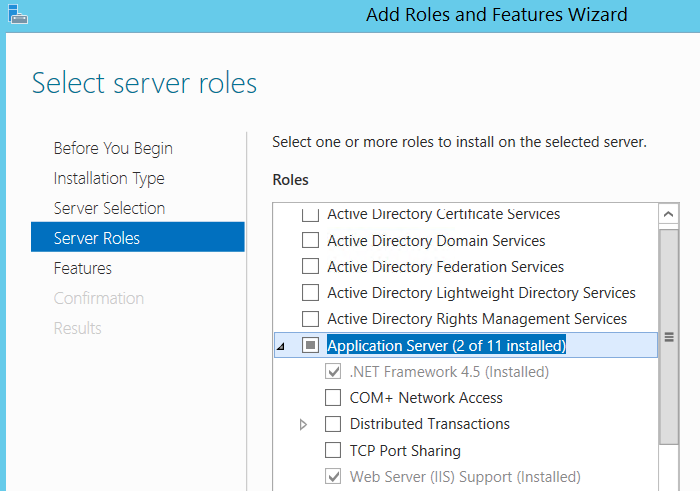
* Setting up On-Premise Website in IIS
* Connecting Visual Studio with Azure
* Associate CloudShop with a new Cloud Service Project
* Publish website to Azure Web Role

# Setting up On-Premise Website in IIS

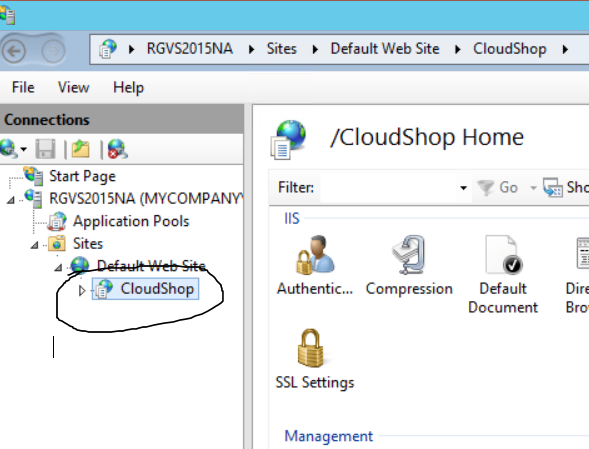
* Make sure that IIS is installed on the local server. On server 2012 R2 you can enable it as role as shown below.



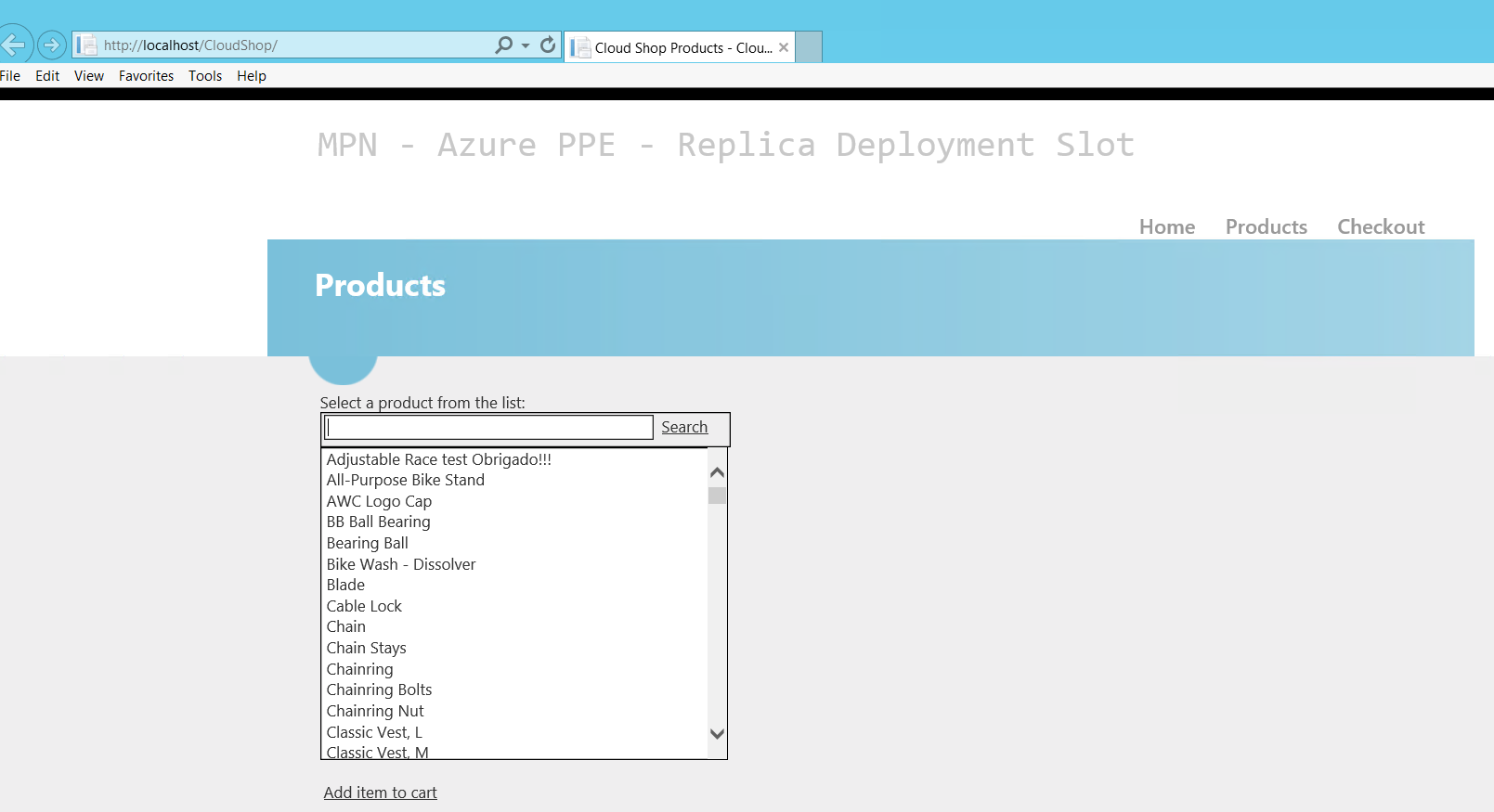
* If this is a fresh machine, installing IIS alone won’t be enough (you’ll face config errors when running). On server 2012 R2, you also need to make sure that you’ve installed “Application Server” role as well as the optional “Web Server (IIS) support” feature under Application server as shown below:



* Also make sure to give permissions to the CloudShop folder to “Everyone” and also make sure that the folder/files are NOT “Read-Only”.
* Extract CloudShop to a local folder.
* Launch VS 2015 in administrative mode and open the Solution file (CloudShopSolution.sln).
* VS 2015 will automatically setup IIS and host CloudSolution to it. You should see something like below.



* Run the application from Visual Studio and you should be able to see the app running on localhost as seen below.



* If you see an ADO.NET error, then that’s probably because the project uses AdventureWorks Databases hosted at sw7heo19qc.database.windows.net and uses connection string mentioned in web.config a sshown below.



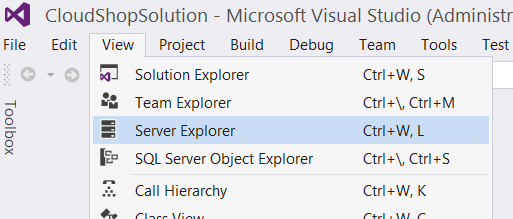
If for some reason, this database is unavailable, please host [AdventureWorks](http://msftdbprodsamples.codeplex.com/releases/view/37304) Database in your own publicly accessible SQL Database and replace the connection string.

* Congratulations! CloudShop is now setup to run on an IIS VM.

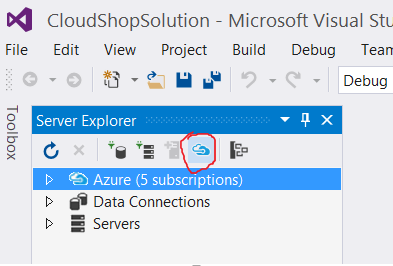
# Connecting Visual Studio with Azure

Before we could migrate on premise CloudShop to Azure Web Role, we need to connect Visual Studio to Microsoft Azure. Below steps demonstrate how to do so.

* Open “Server Explorer” in VS 2015 as shown below.



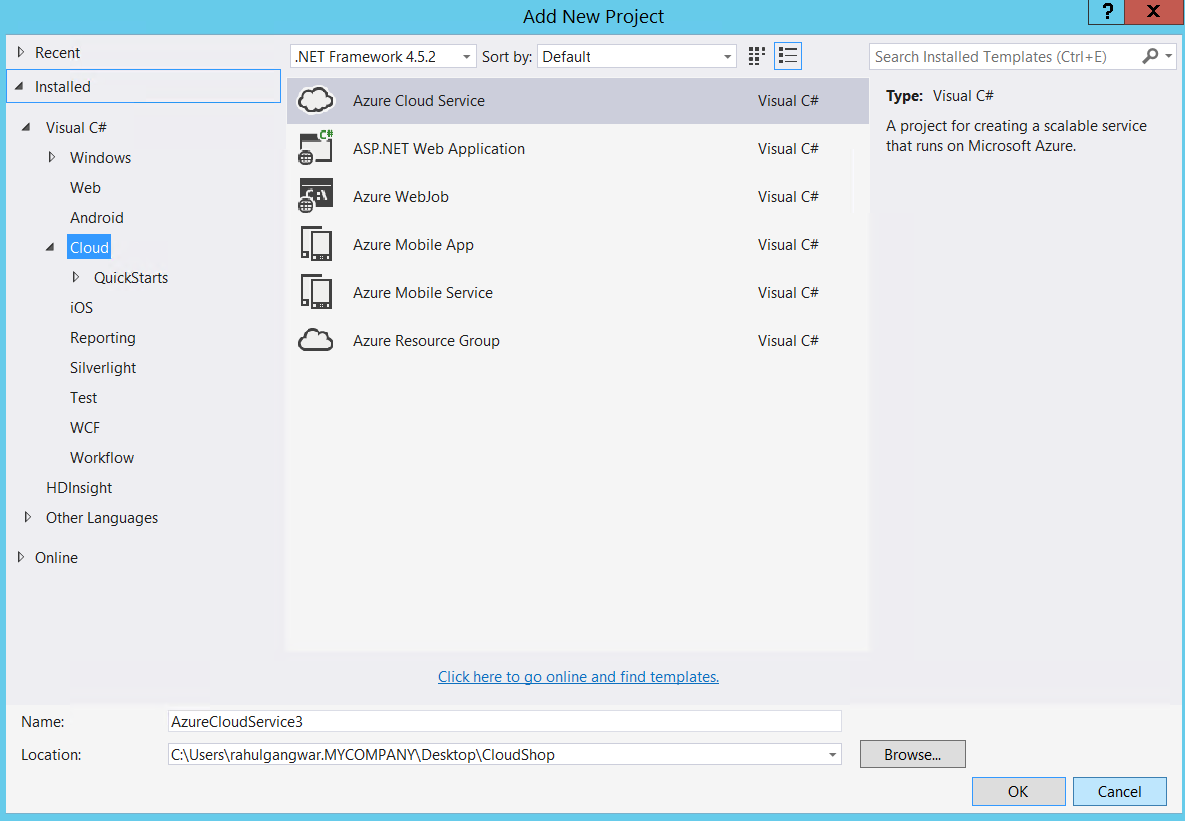
* From Server Explorer, click on icon to connect to Microsoft Azure Subscription as shown below.



* Follow the wizard and input your credentials for your Azure Subscription.
* Congratulations! Your Visual Studio is now connected to your Azure Subscription.

# Associate CloudShop with a new Cloud Service Project

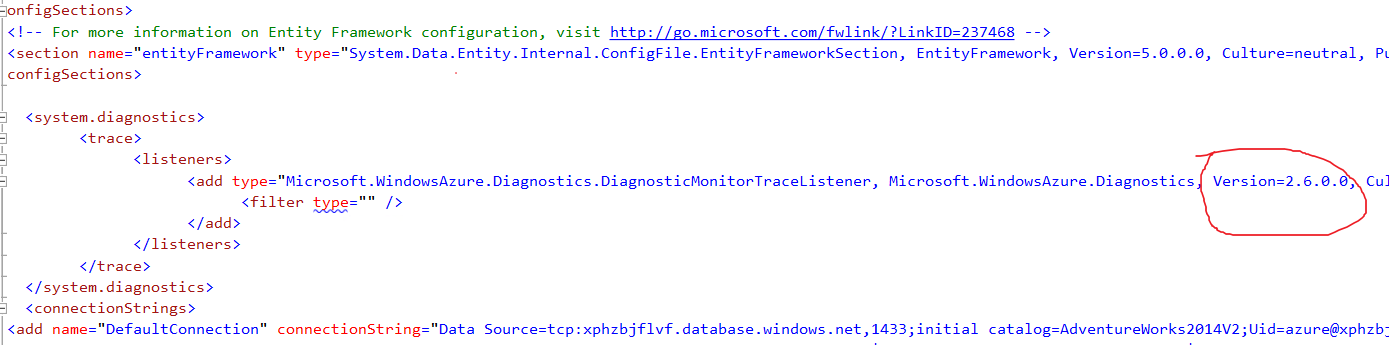
* From Solution Explorer, right click your “**Solution**” and select Add🡪 New Project.
* Select Cloud Service as shown below



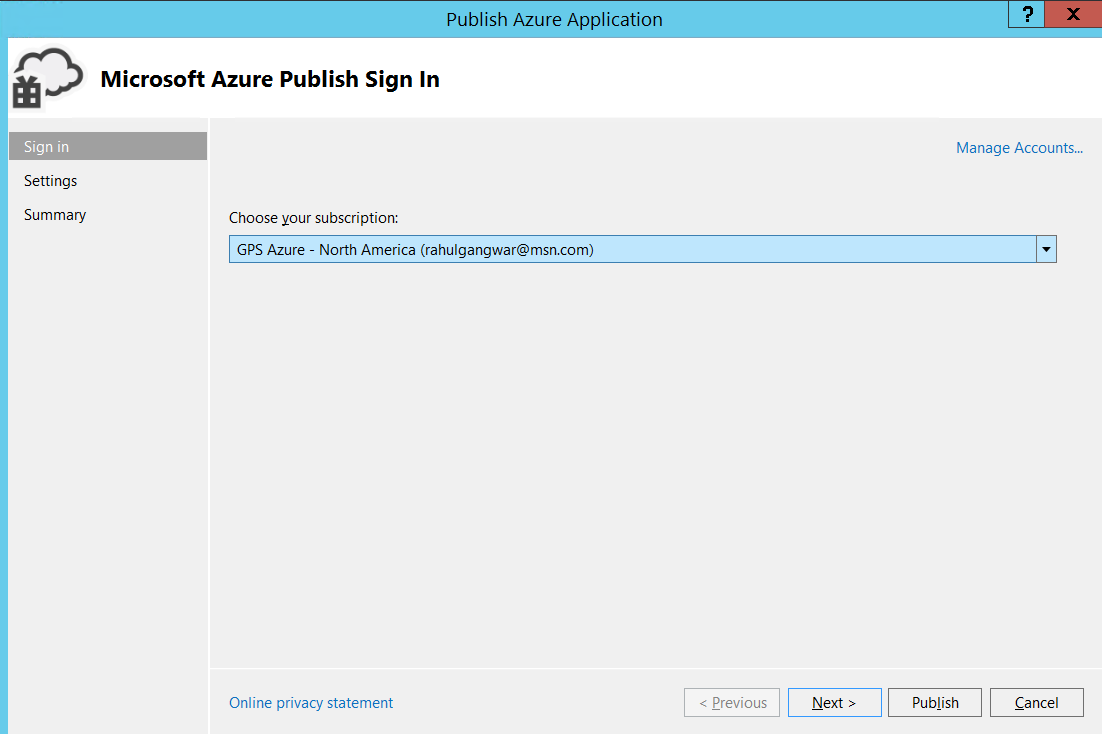
* Click OK on next screen. Remember NOT to add any web role at this stage.
* Right Clck “Roles” and select Add🡪Web Role Project In Solution. In subsequent screen, select ”CloudShop” and click OK.

# Publish website to Azure Web Role

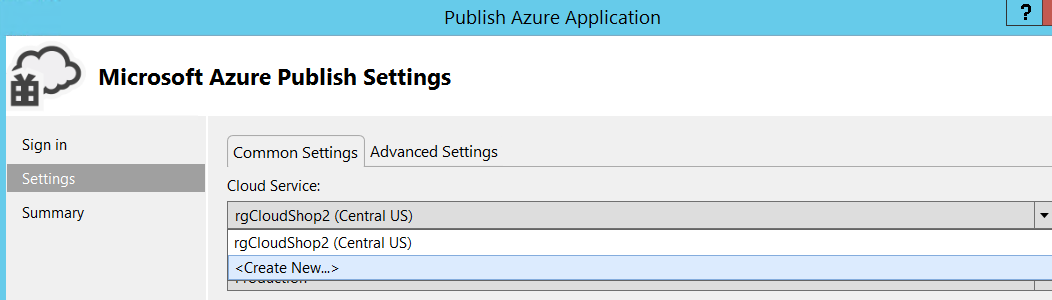
* Before we publish website to web role, we need to accommodate SDK difference between VS2015 and CloudShop application. Cloud Shop application is built using Azure SDK 2.5 while we are using SDK version 2.6. This makes Microsoft.WindowsAzure.Diagnostics dll in our project obsolete. Follow below bullet point to fix the discrepancy.
* Open web.config and replace the version of Diagnostics dll from 2.5.0.0 to 2.8.0.0 as shown below.



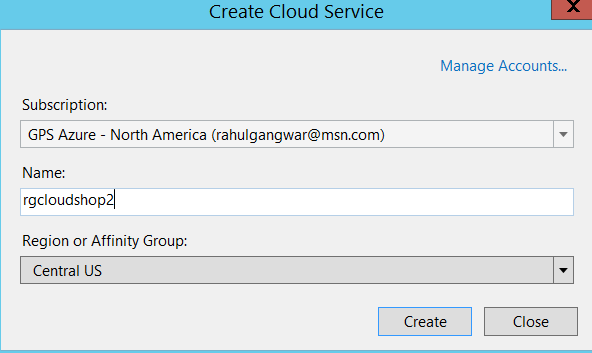
* Now, we are ready to continue with publishing.
* From Solution Explorer, right click the cloud service project (not cloud shop) and click “Publish”.
* In subsequent screen select subscription and click next.



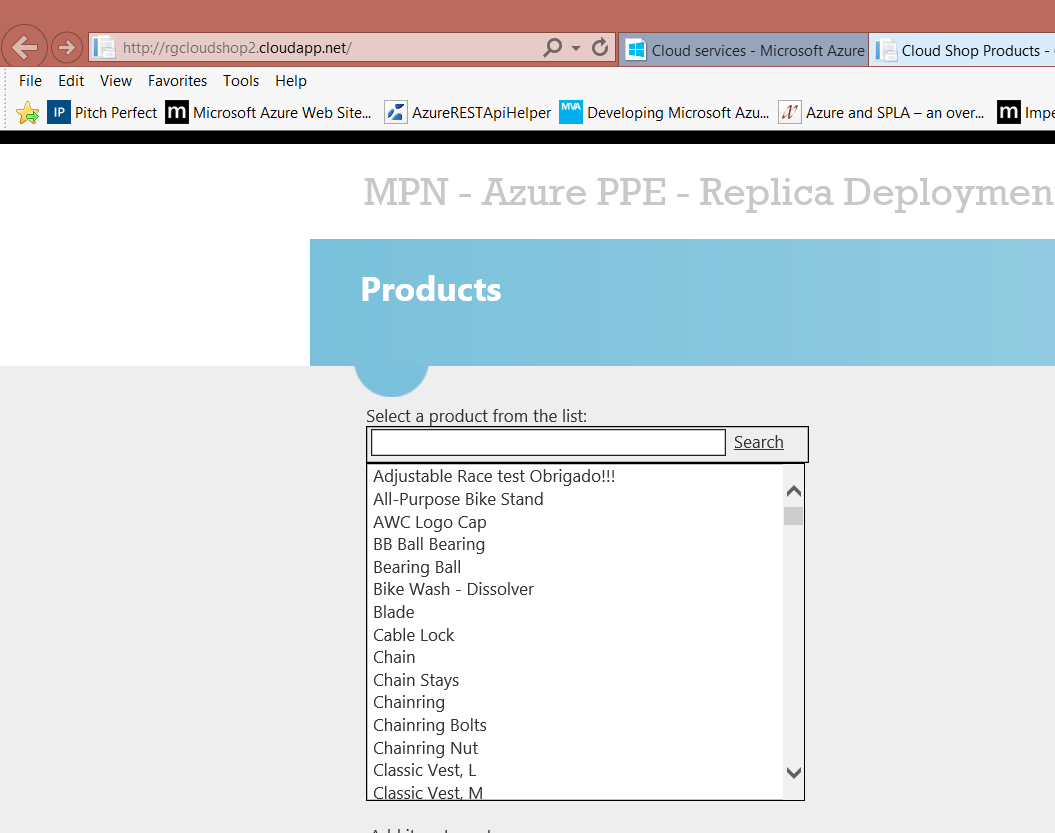
* From Cloud Service dropdown, select “<Create New…>” as shown below



* Give a globally available name, select region and click “Create” as shown below



* Once cloud service is created, click “Publish”.
* Congratulations! You have successfully published your IIS website to Azure Web Role. You can now navigate to it publicly as shown below.



Lab Completed!