**Deploy Web MVC EF with Authentication**

Reference: <http://www.c-sharpcorner.com/UploadFile/4b0136/getting-started-with-iis-host-and-publish-in-mvc-5/>

**Introduction**

This article explains how to host your MVC 5 App in IIS Express and store the data and information in the database created in SQL Server. Visual Studio uses the ASP.NET Development Server as a default and here I am using IIS Server to host a MVC 5 App. As you know, in Visual Studio 2013 Preview the ASP.NET Web Application uses a MVC5 project template **and** MVC5 uses the Bootstrap theme to display the application, so here I am using the Visual Studio 2013 Preview to develop an application using the MVC project template.

**Prerequisites**

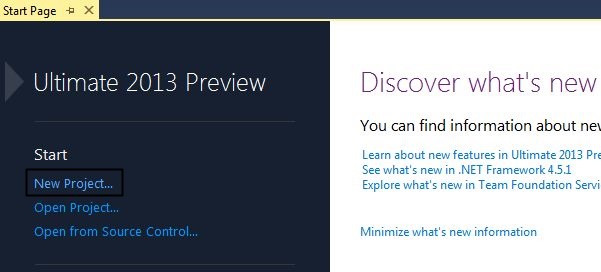
The following are the prerequisites to develop a MVC 5 application:

* Visual Studio 2013 Preview
* SQL Server 2008 or 2012

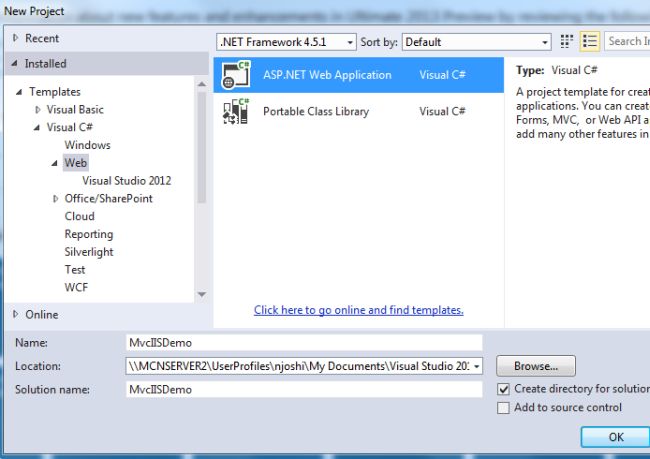
**Creating MVC 5 Application**

Let's start to develop an application in MVC 5 using the following procedure.

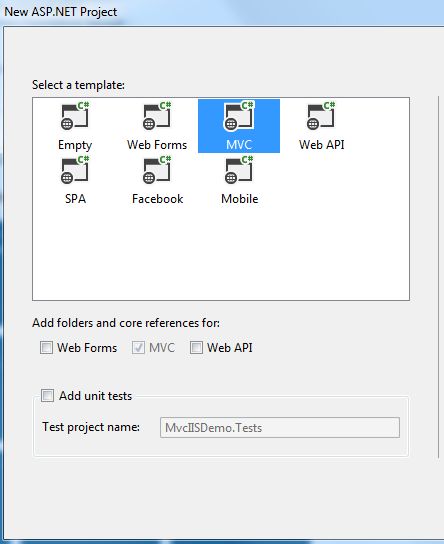
**Step 1:**Open Visual Studio 2013 Preview



**Step 2:**Click on "New Project" and enter the application name.



**Step 3:**Select MVC project template to create the project.



**Step 4:**In your Solution Explorer, open the Web.Config file.

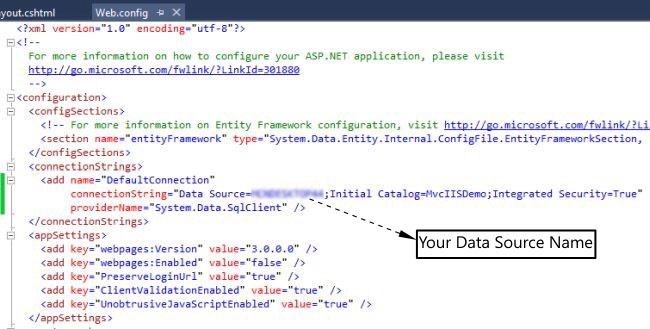
As you can see, the connection property is available on my file:

<add name="DefaultConnection"

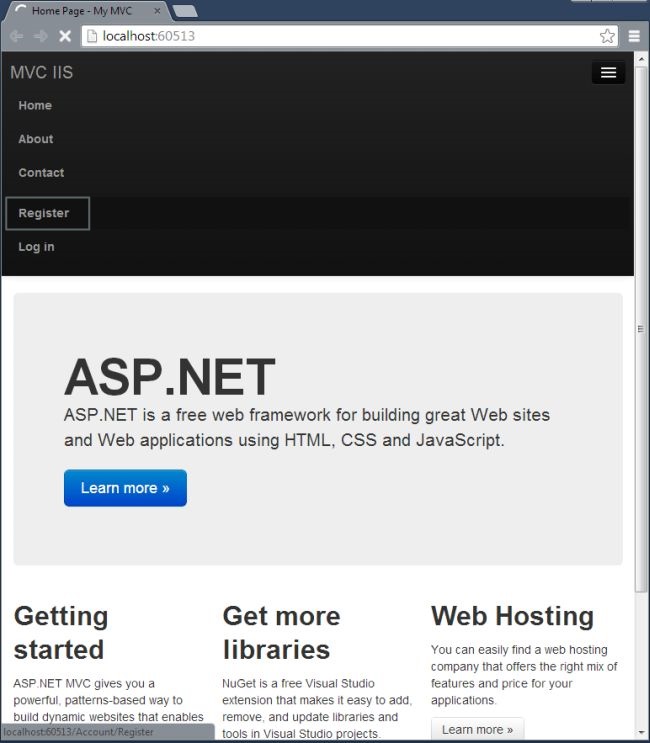
     connectionString="Data Source=(LocalDb)\v11.0;AttachDbFilename=|DataDirectory|\aspnet-MvcIISDemo-20130911053147.mdf;Initial Catalog=aspnet-MvcIISDemo-20130911053147;Integrated Security=True"

     providerName="System.Data.SqlClient" />

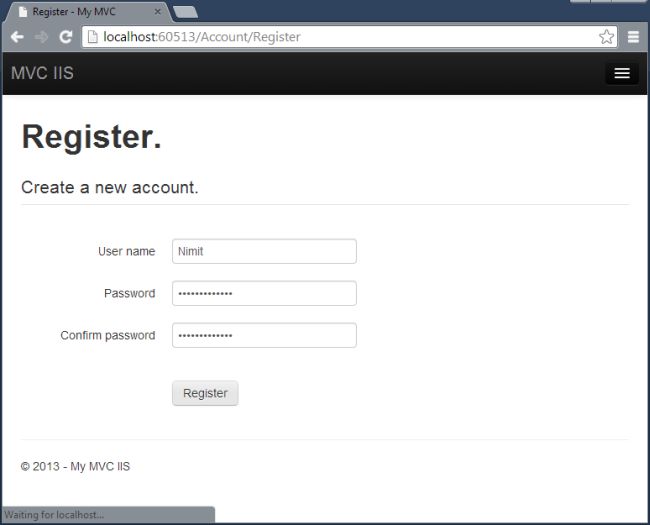
Now, change the code above as shown below:



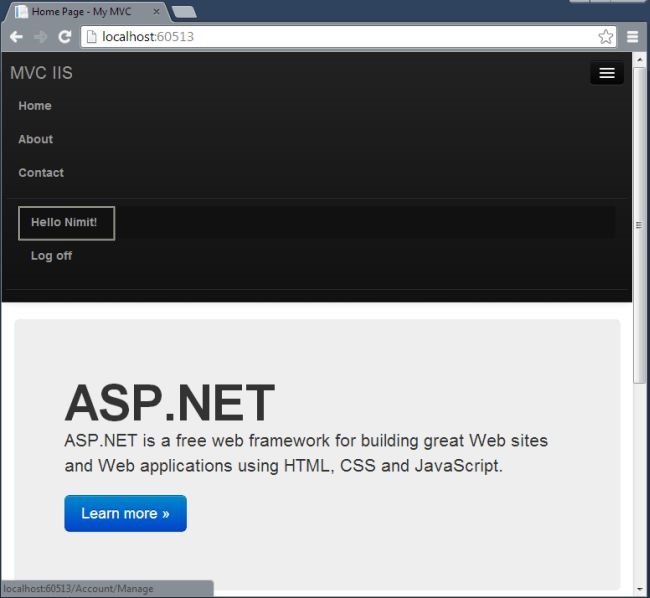
**Step 5:**Debug your application and click on the "Register Link" as shown below:



**Step 6:**Enter the details and click on the "Register" button.



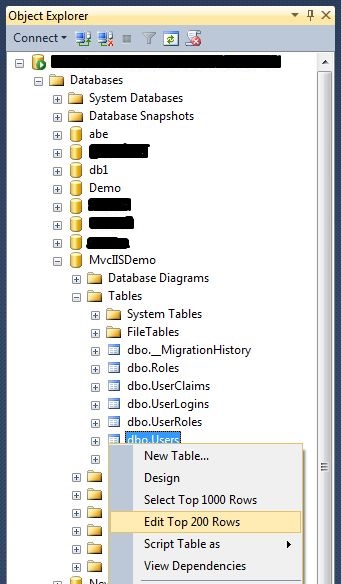
You will see your UserName in your next page window:



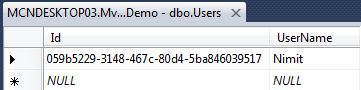
**Verify in SQL Server 2012**

As you can see, I did not create a database nor any table in my SQL Server 2012. When I declared my connection string, it automatically created a database and table in my SQL Server. You do not need to worry about that.

Let's see my database in my SQL Server Object Explorer.



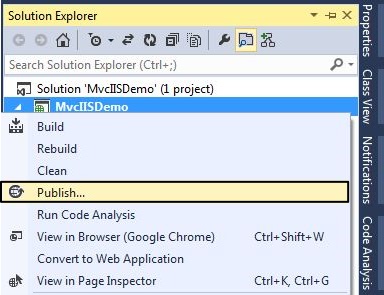
Select your database and open the Users Table. You can see the user information as shown below:



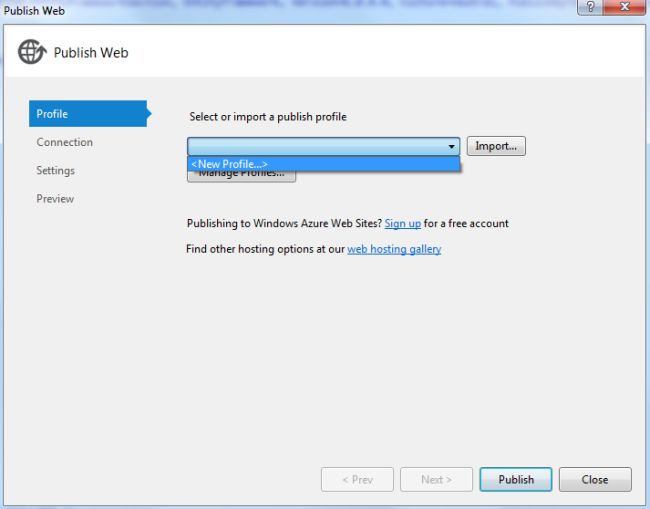
**Publish Application**

The next step is to host in IIS 8 (IIS Express). For that use the following procedure:

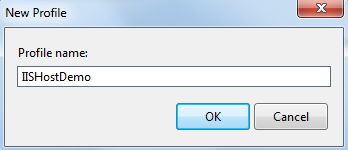
**Step 1:**In your Solution Explorer select your project and right-click to select Publish.



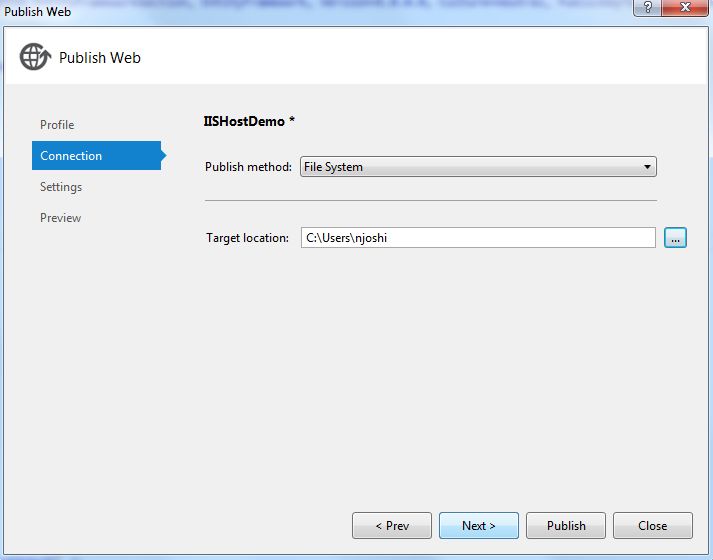
**Step 2:**Select NewProfile and click Next as shown below:



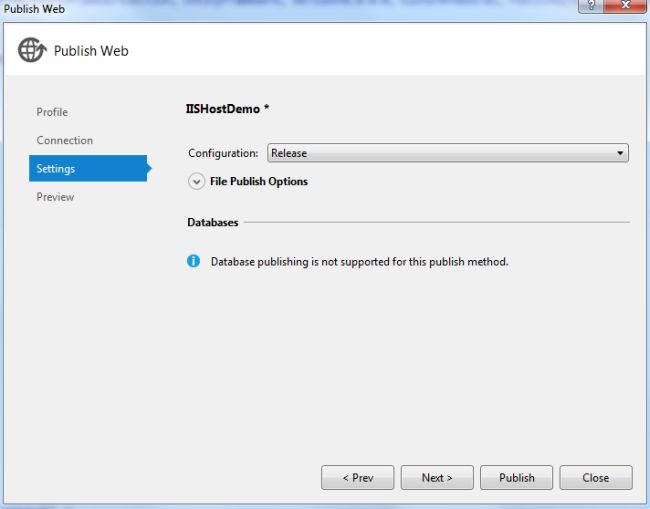
**Step 3:**Enter Profile Name and click OK.



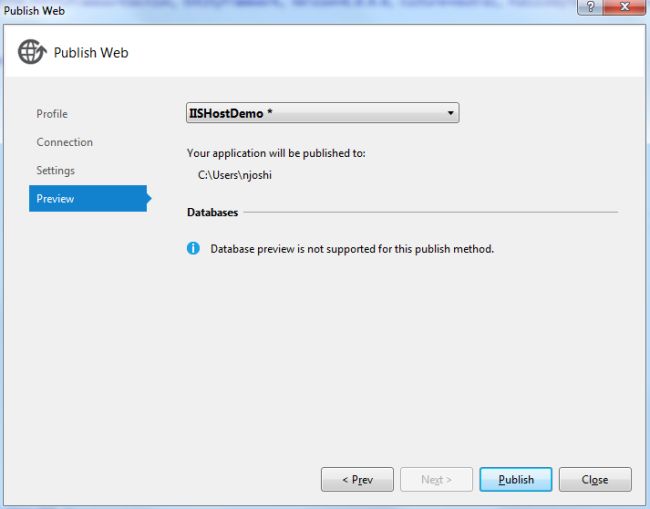
**Step 4:**Select File System in Publish Method and select your preferred location in the Target Folder.



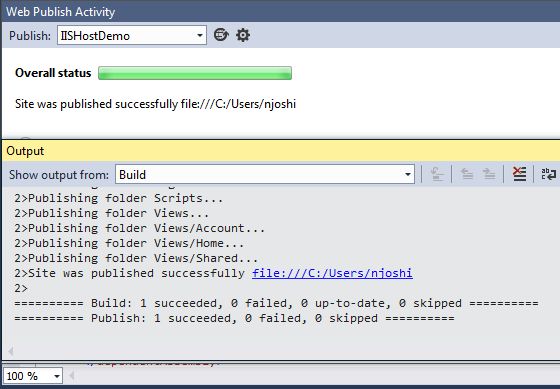
**Step 5:**Select Release in Configuration in the next window and click "Next".



**Step 6:** Click on "Publish" in the next wizard.



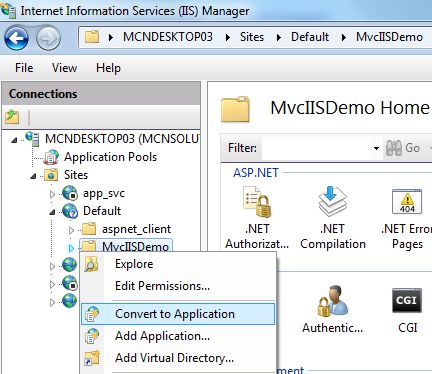
You will see that the application will publish successfully.



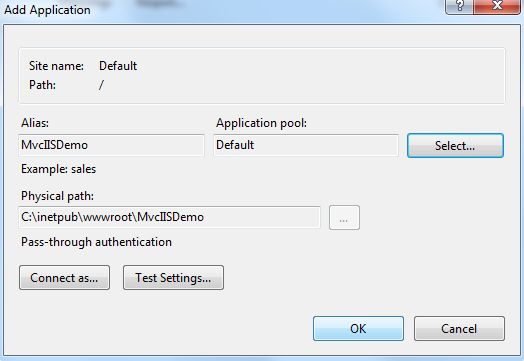
**IIS Host**

Open the Internet Information Services (IIS) Manager and follow the procedure given below.

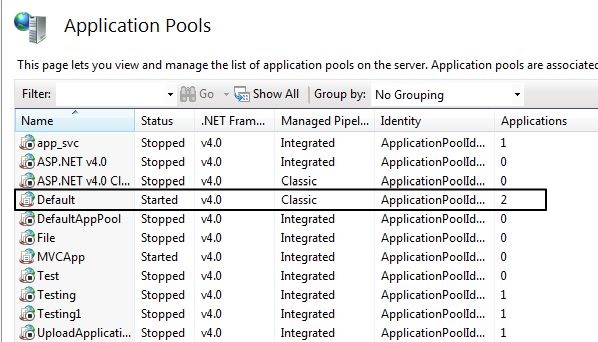
**Step 1:**Select your Profile in the Default Site.



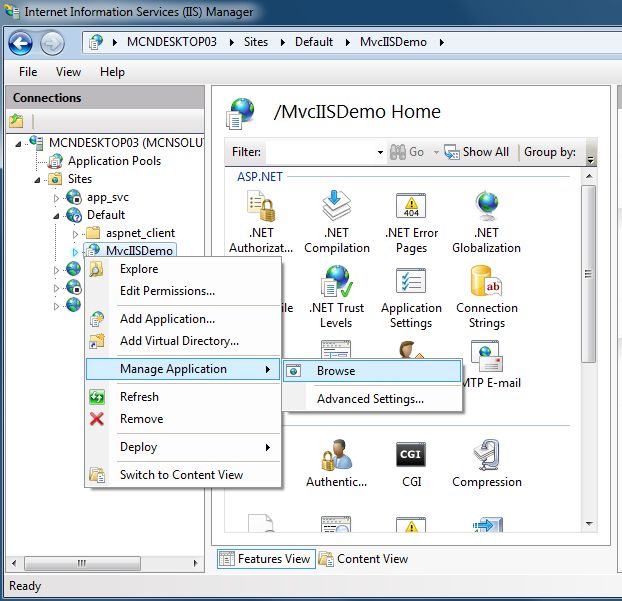
**Step 2:**Just click "OK" in the next Add Application wizard.

****

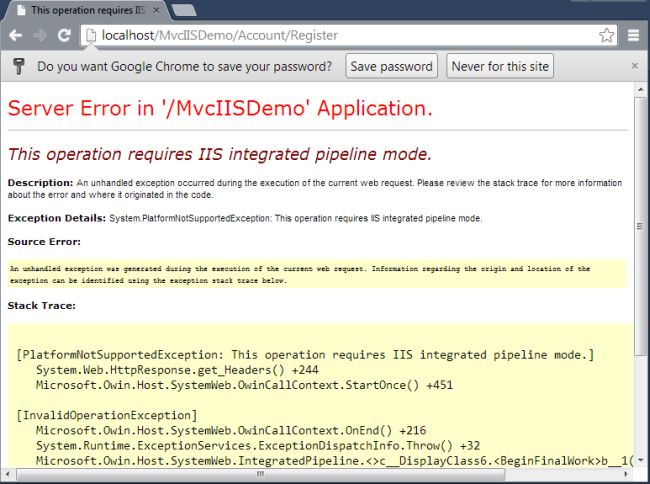
**Step 3:**Ensure your Default Site Status is Started in Application Pools.



**Step 4:**Right-click your Profile in the Select Browse in Manage Application.

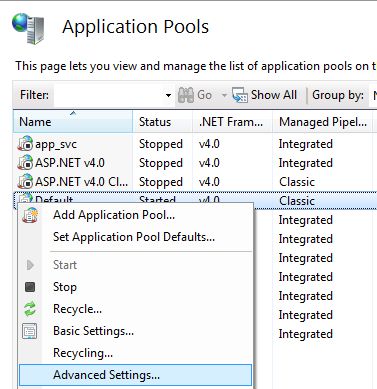


**Step 5:**You will see an error such as "This Operation requires IIS integrated pipeline mode" in your browser.

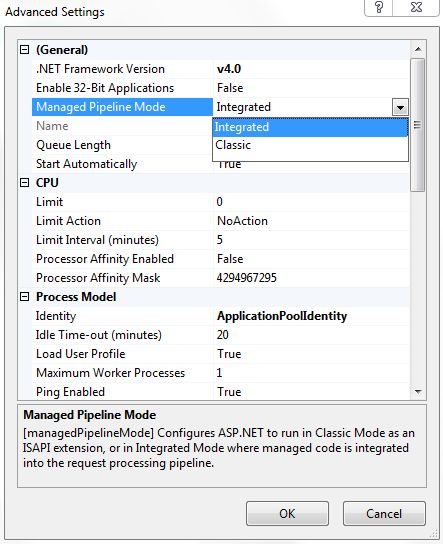


**Note:**Please don't close the browser**.**

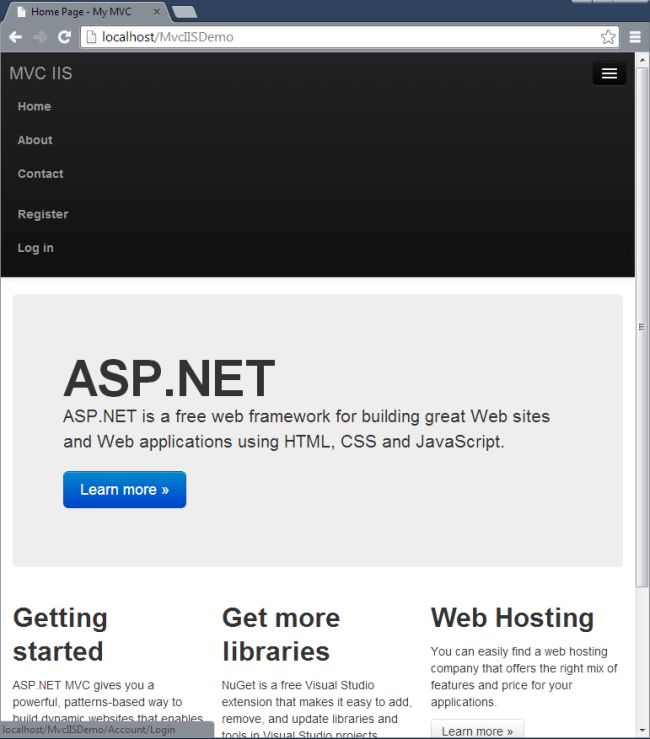
**Step 6:**Select "Advanced Settings" of your site in the Application Pool by right-clicking and selecting "Advanced Settings".



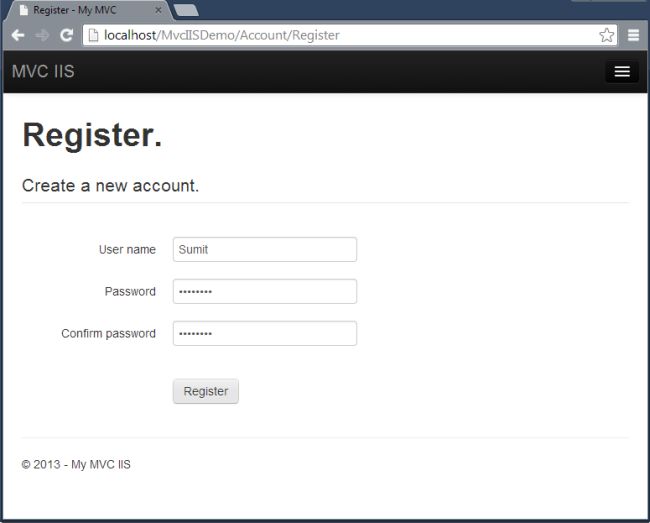
**Step 7:**Select Integrated in the Managed Pipeline Mode.



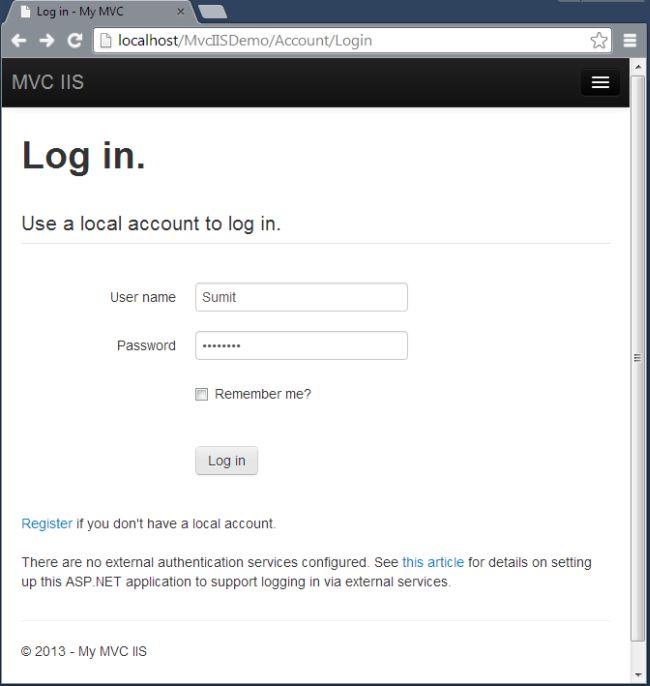
**Step 8:**Reload your browser and you will see the Home Page.

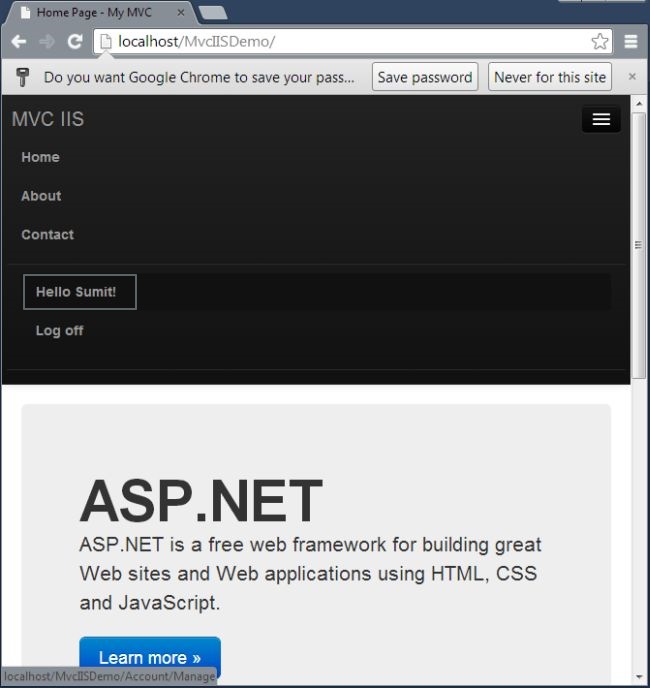


**Step 9:**Now click on "Register" and enter the details.



**Step 10:**After registration you can login.





That's it.

**Summary**

So far, this article will help you to learn to create an MVC 5 Application to store the data and information in SQL Server directly and you don't need to worry about creating the database manually. With this article you can also learn to publish your article and browse your application from IIS Manager and the app will run on the IIS Server. So just go for it and let me know if you have any problem.

Thanks for reading my article.

Some problem raise when to deploy:

Reference: <http://stackoverflow.com/questions/2806438/login-failed-for-user-domain-machinename>

[

*NETWORK SERVICE and LocalSystem will authenticate themselves always as the correpsonding account locally (builtin\network service and builtin\system) but both will authenticate as the machine account remotely.*

*If you see a failure like Login failed for user 'DOMAIN\MACHINENAME$' it means that a process running as NETWORK SERVICE or as LocalSystem has accessed a remote resource, has authenticated itself as the machine account and was denied authorization.*

*Typical example would be an ASP application running in an app pool set to use NETWORK SERVICE credential and connecting to a remote SQL Server: the app pool will authenticate as the machinerunning the app pool, and is this machine account that needs to be granted access.*

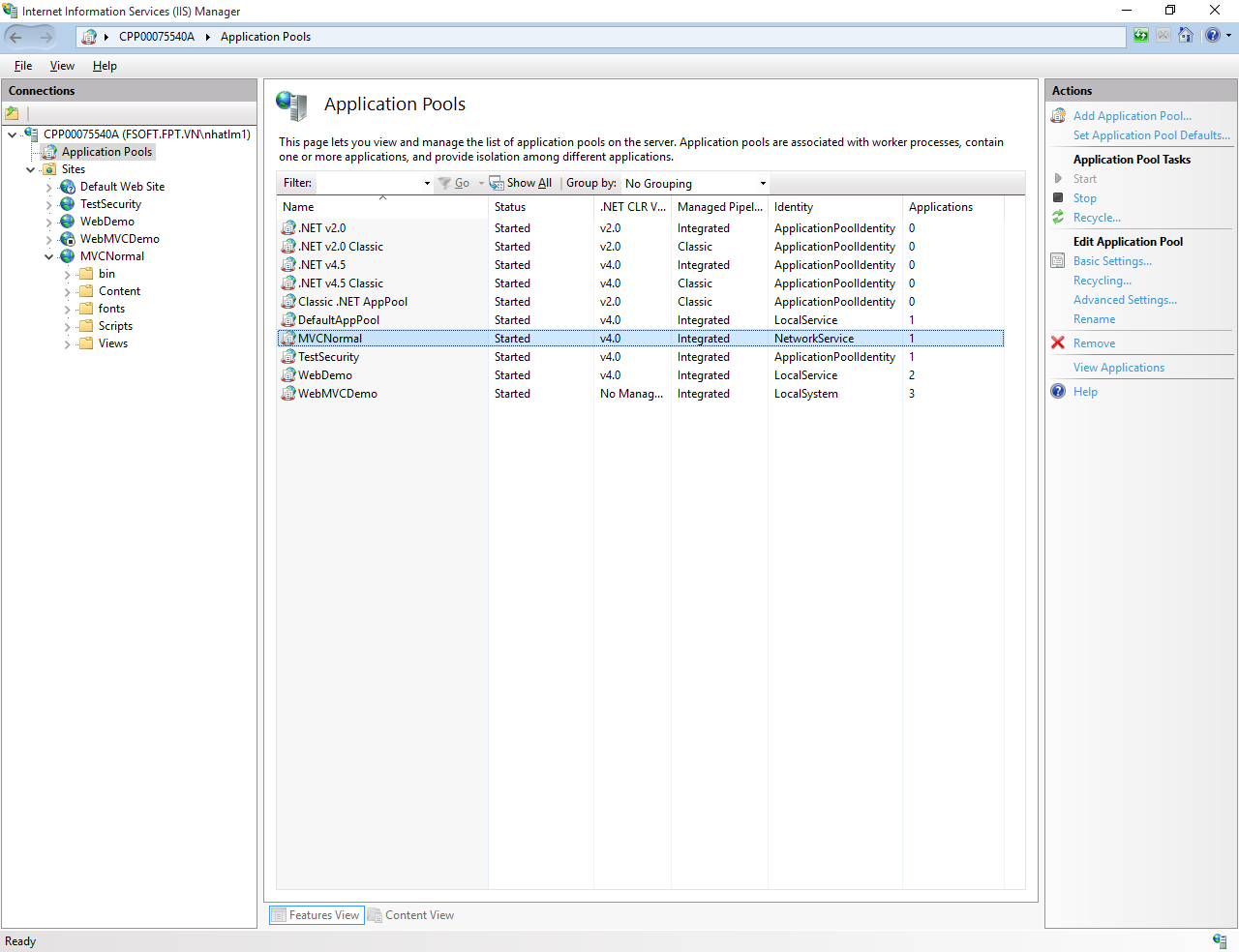
*When access is denied to a machine account, then access must be granted to the machine account. If the server refuses to login 'DOMAIN\MACHINE$', then you must grant login rights to 'DOMAIN\MACHINE$' not to NETWORK SERVICE. Granting access to NETWORK SERVICE would allow a****local****process running as NETWORK SERVICE to connect, not a remote one, since the remote one will authenticate as, you guessed, DOMAIN\MACHINE$.*

*If you expect the asp application to connect to the remote SQL Server as a SQL login and you get exceptions about DOMAIN\MACHINE$ it means you use Integrated Security in the connection string. If this is unexpected, it means you screwed up the connection strings you use.*

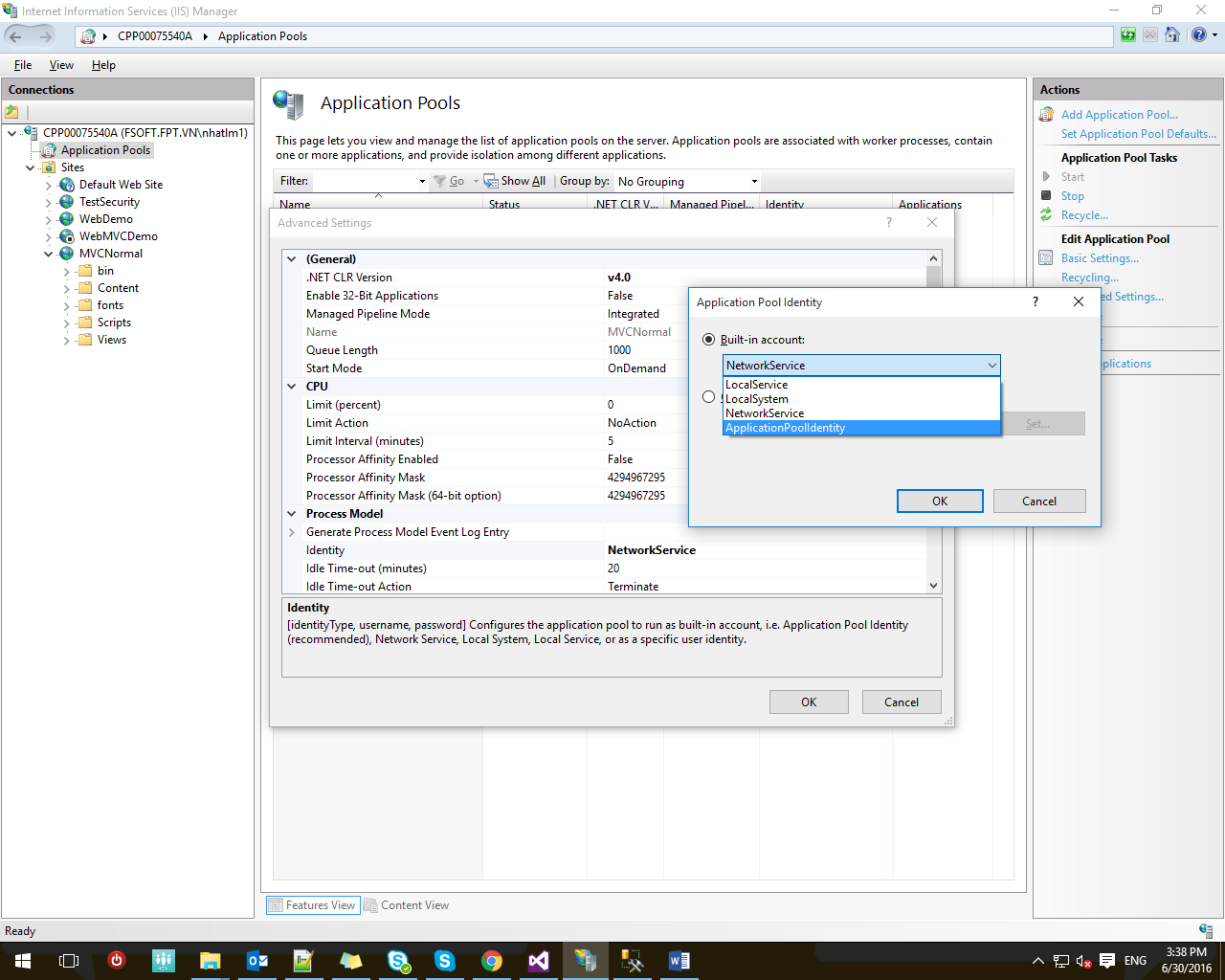
]

Example:

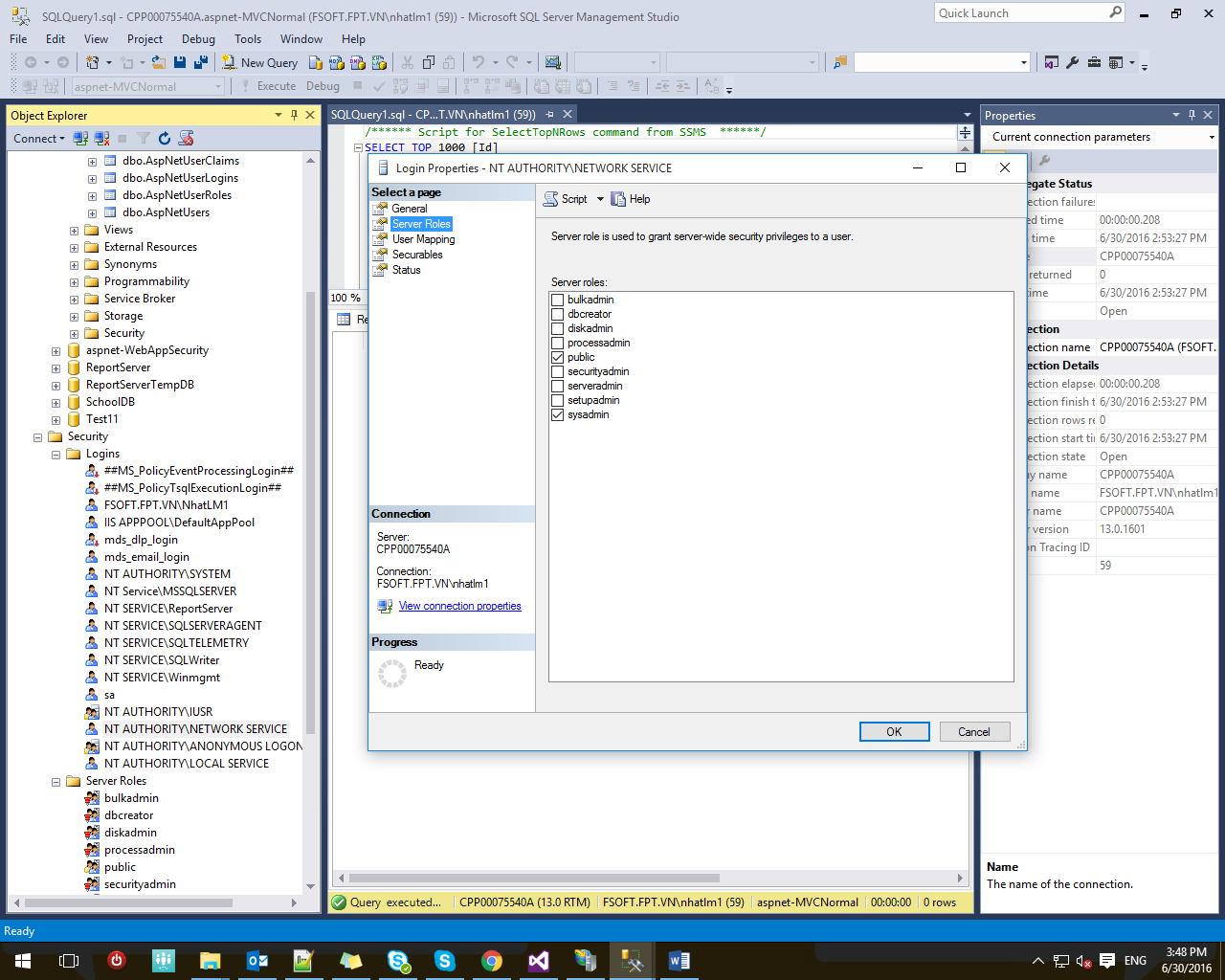
**Case 1: If you select identity is Network service**

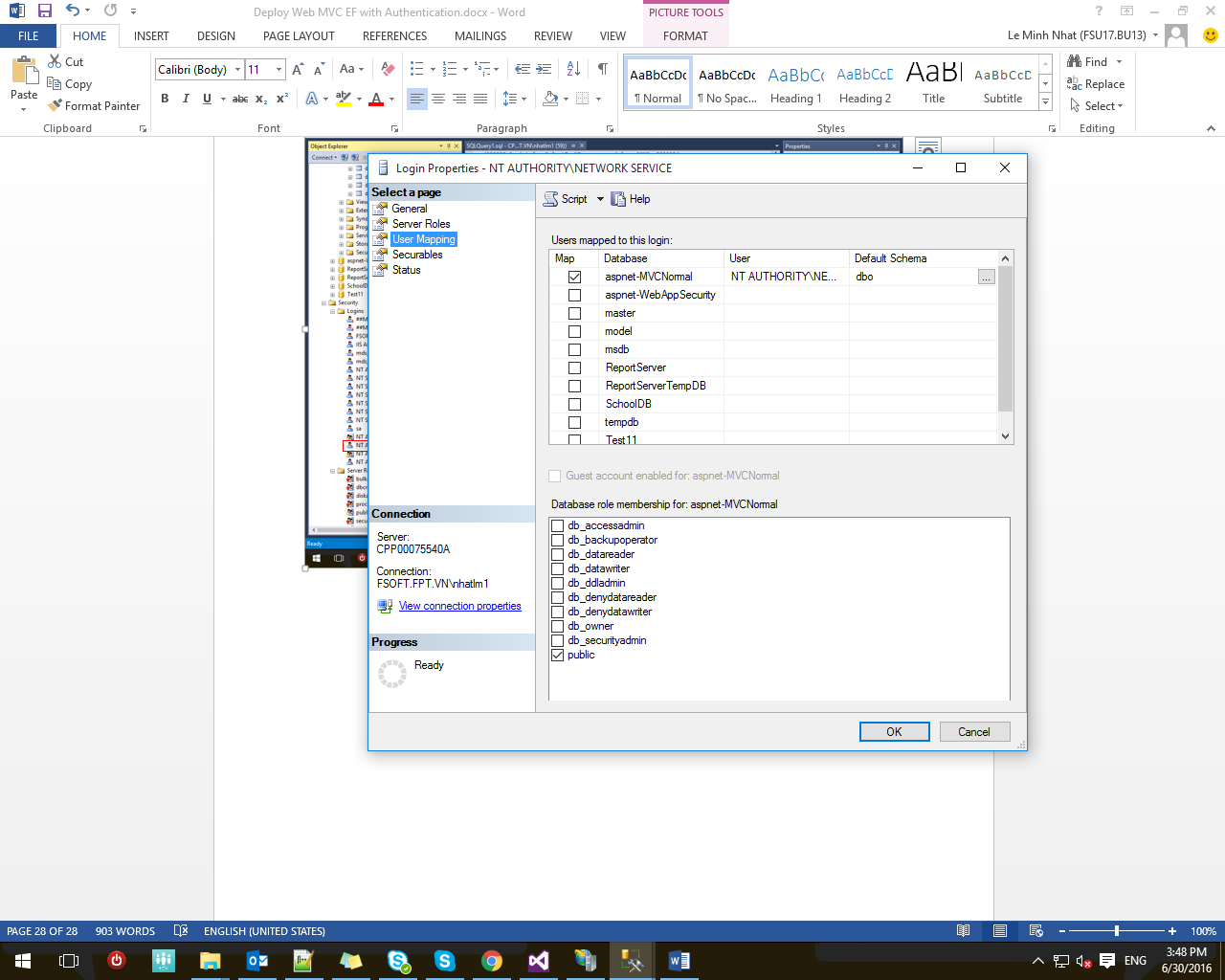


**Select [Advanced setting] to change identity for application pool**

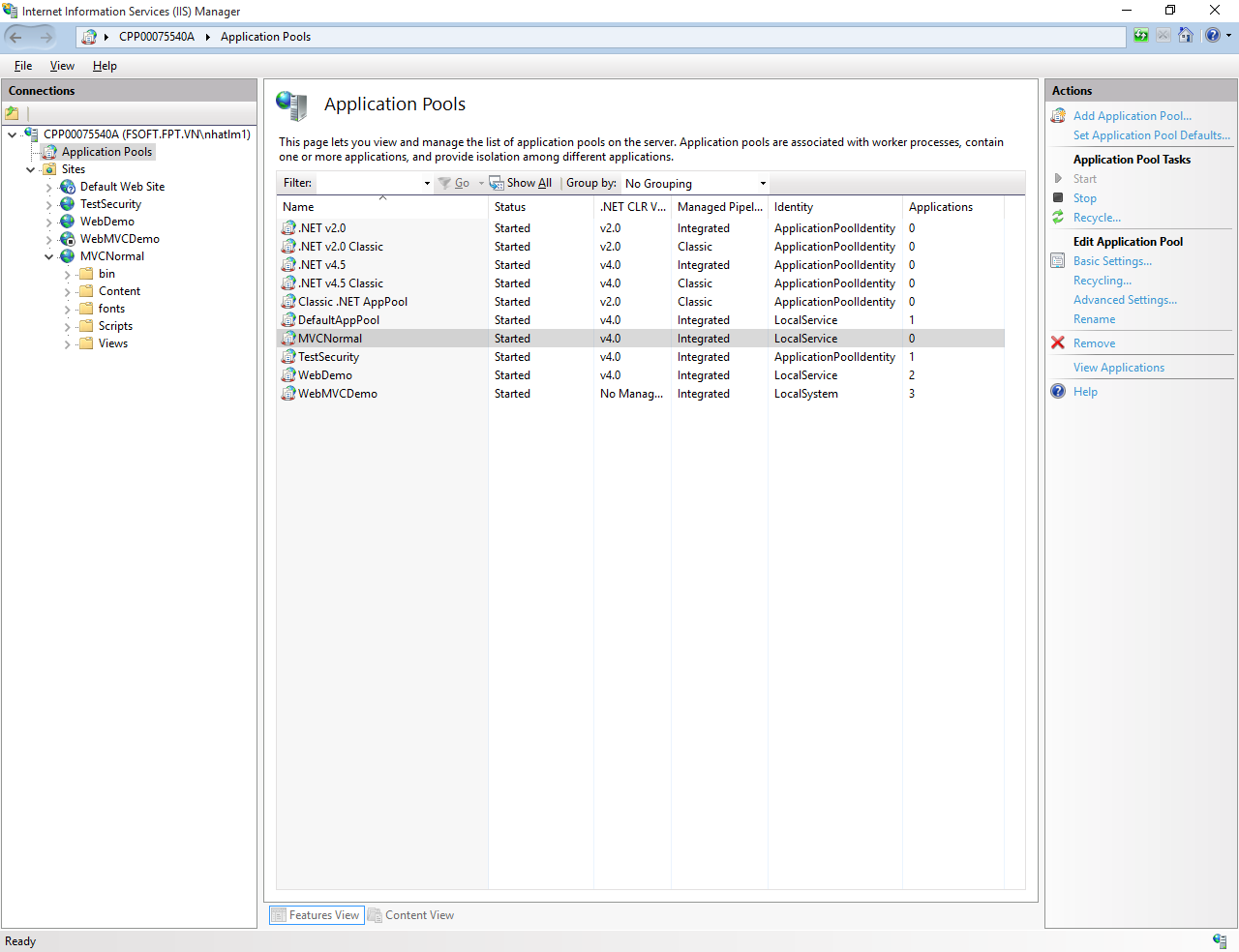


**Case 1: If identity is NetworkService, then sqlserver configurated as below:**

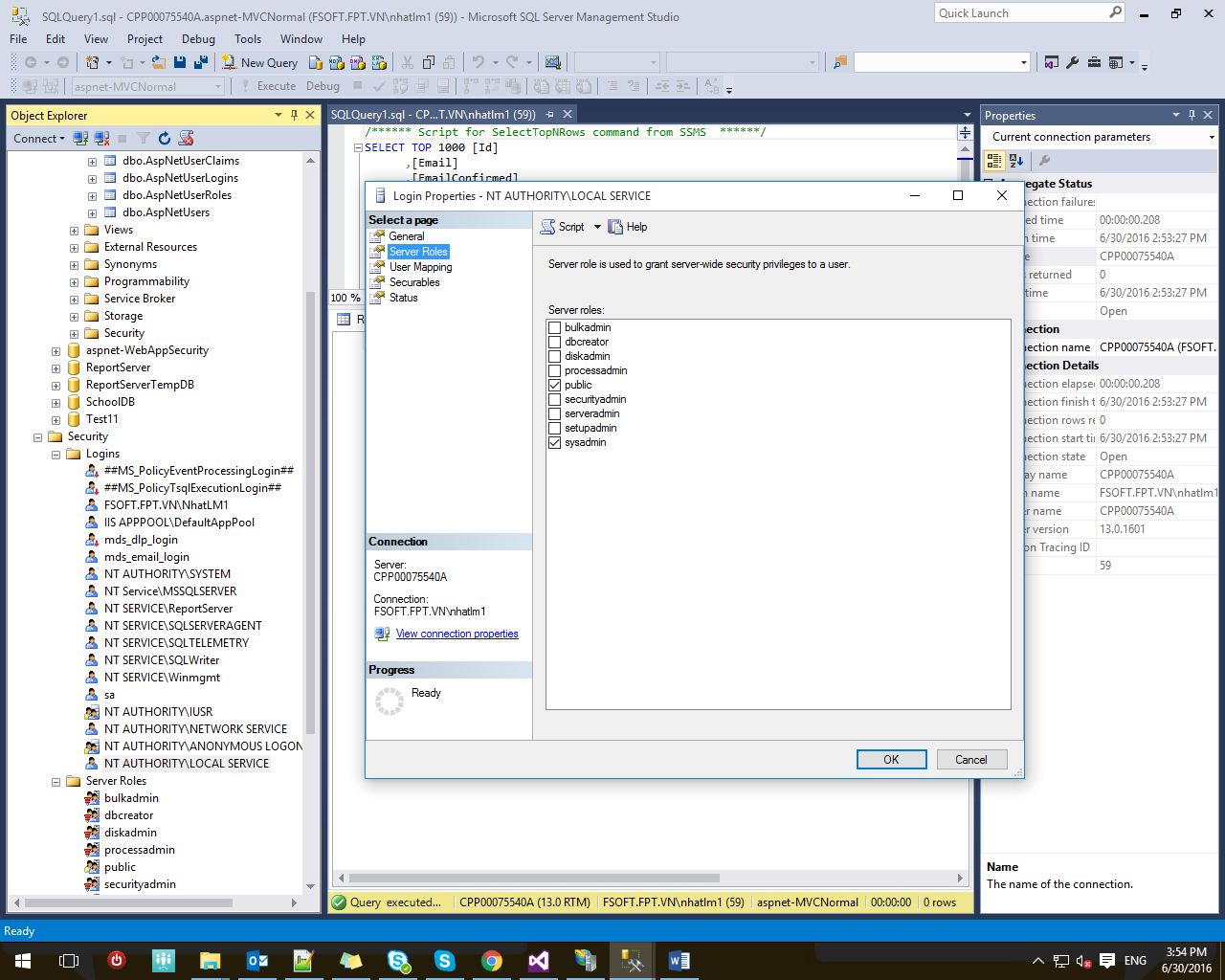


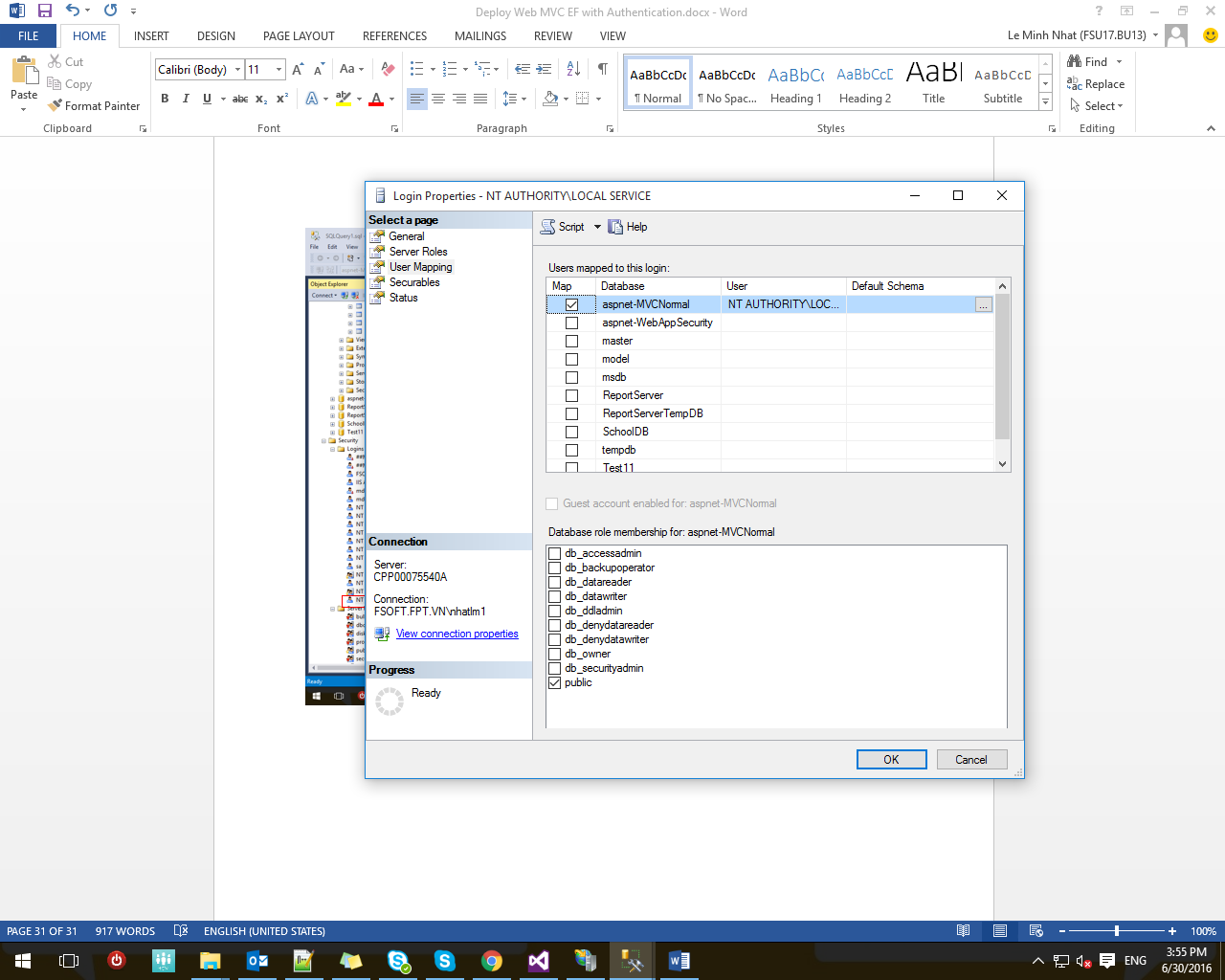


**Case 2: If identity is LocalService**

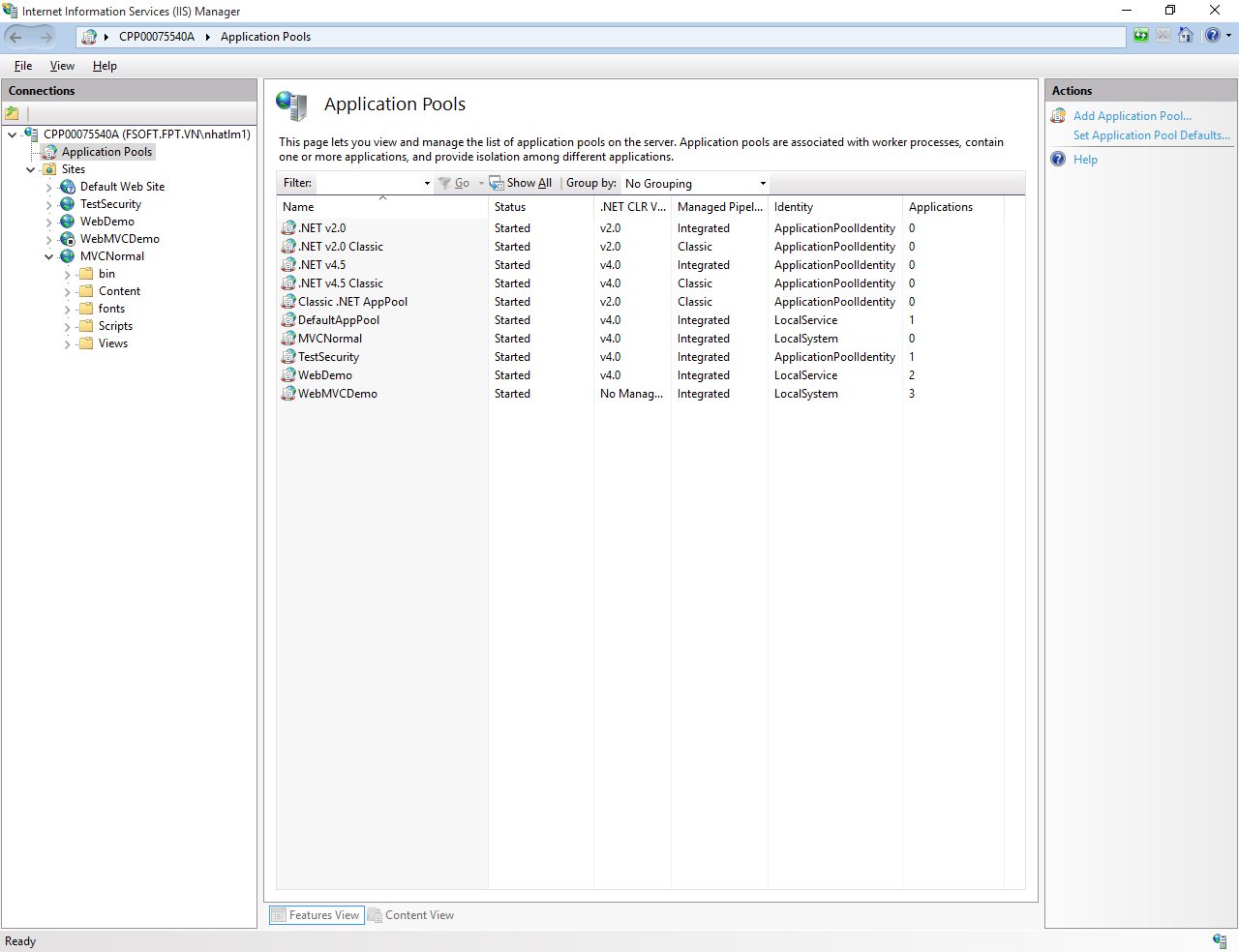


**SQL Server configurated as below:**





**Case 3: If identity is LocalSystem**



SQL server configurated as below:

