

2017

SVDNN-NUGET SERVER

Andre Masters

Automated Design Solutions

2/22/2017

Nuget Server Setup Instructions

Contents

Perquisites.....	3
Introduction to Nuget Server	3
Part I Create a Simple Web Server	3
Part II Create a Nuget Package	9
Part III Push a Nuget Package to the Local Nuget Server.....	12
Part IV Consume a Nuget Package in a Visual Studio Project.....	13

Perquisites

Completion of Tutorial #1

Technical Resource Requirements:

- Laptop Computer - Windows 10
- Visual Studio Community Edition 2015 or 2017

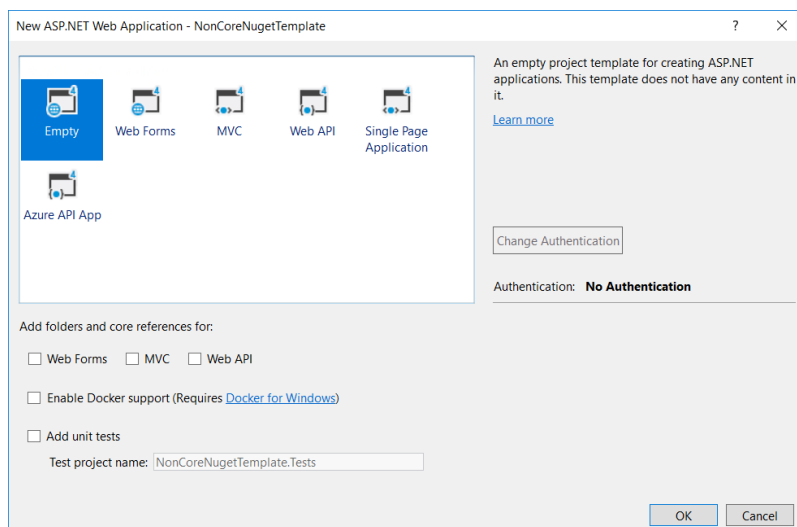
Introduction to Nuget Server

The purpose of having a Nuget server is to serve as an aid in the software development process. A Nuget server will allow for an organized and automated process for updating and storing library code to be reused in a later project.

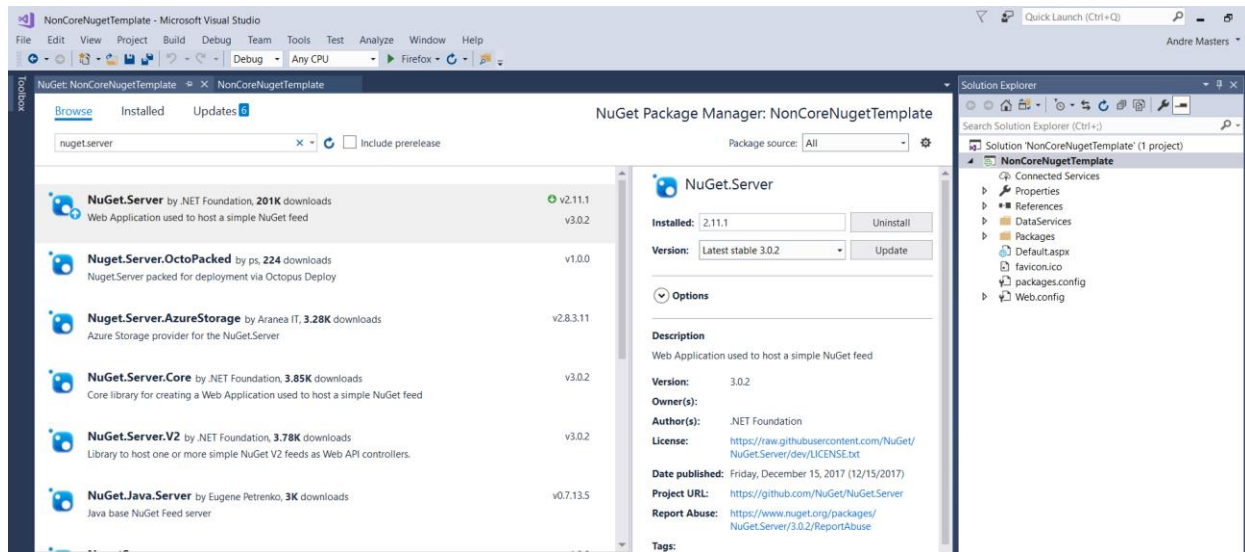
Part I Create a Simple Web Server

Create a web server by completing the following steps as illustrated in the sequence of steps below.

1.0 Create an empty template as shown in the following image.

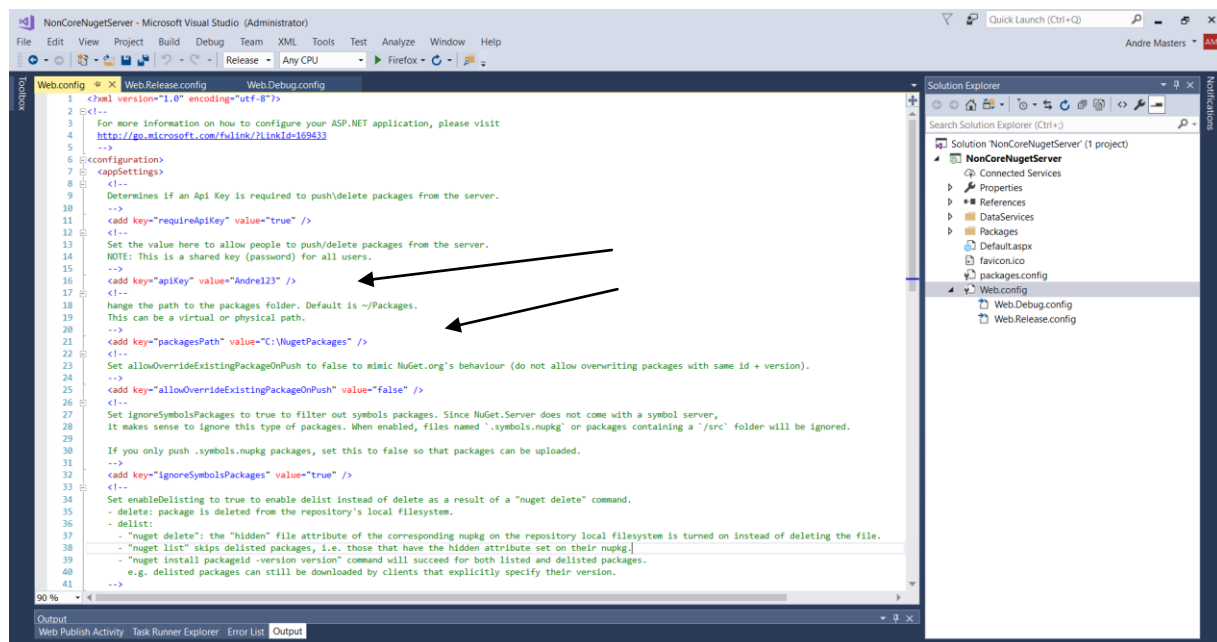


2.0 Create an empty server template as shown in the following image.

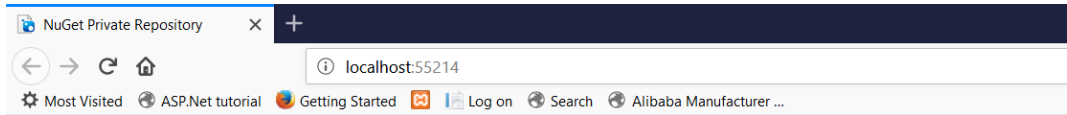


3.0 Edit Web Config file for the desired package path as shown below.

3.0 Enter the following value for apiKey: Andre123 as shown below.



3.0 Press F5 to verify the nuget server is running. The following should appear in the browser as shown below.



You are running NuGet.Server v2.11.1.0

Click [here](#) to view your packages.

Repository URLs

In the package manager settings, add the following URL to the list of Package Sources:

`http://localhost:55214/nuget`

To enable pushing packages to this feed using the nuget command line tool (nuget.exe). Set the api key appSetting in web.config.

`nuget push {package file} -s http://localhost:55214/nuget {apikey}`

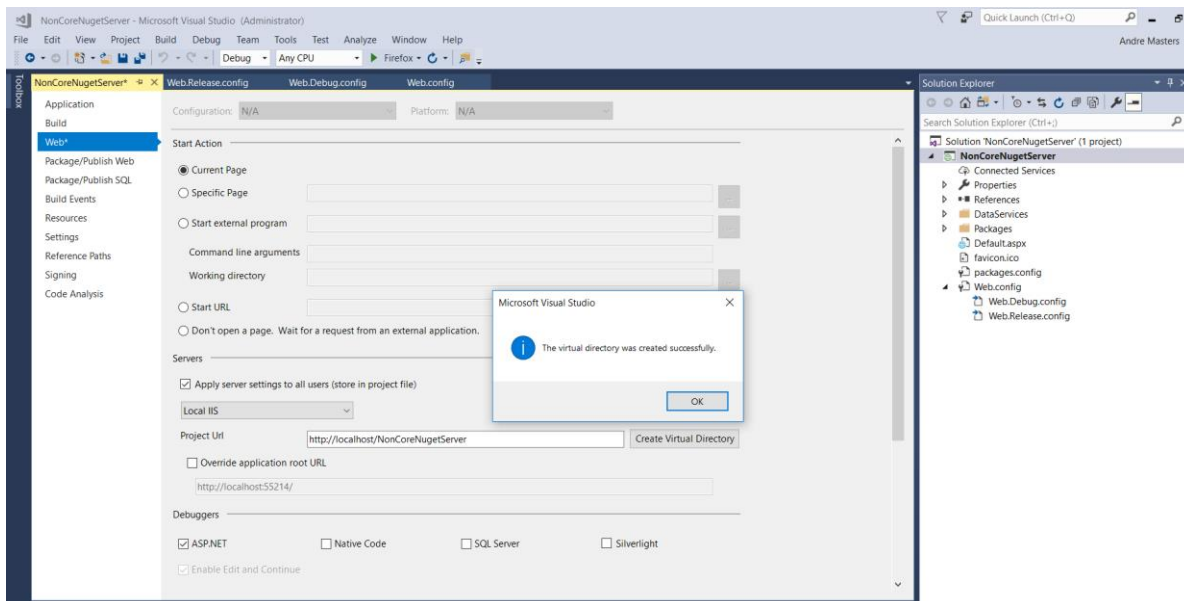
Adding packages

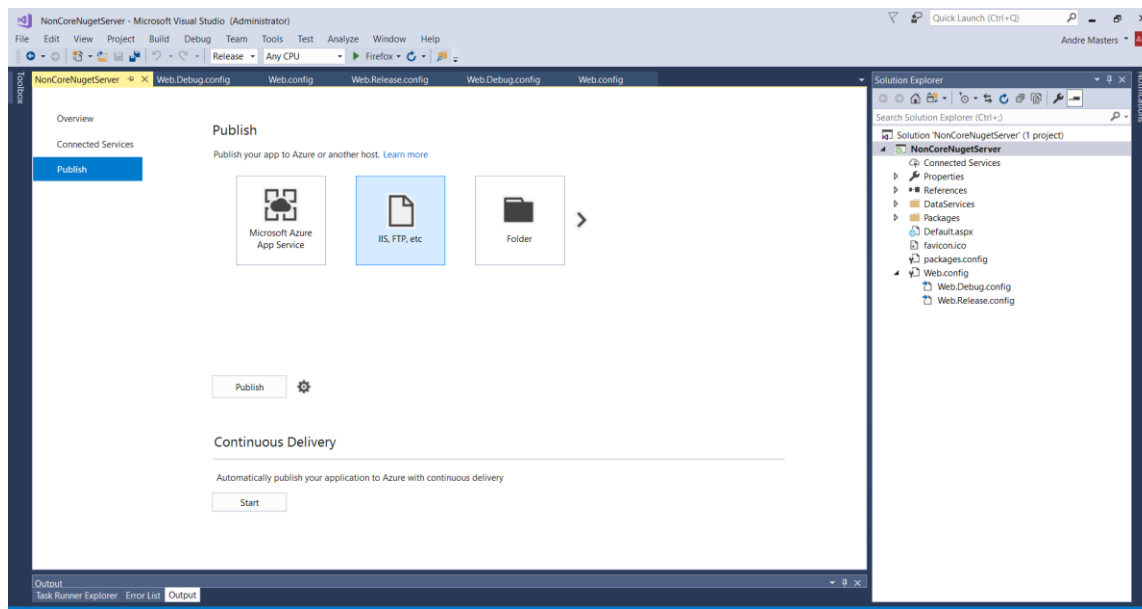
To add packages to the feed put package files (.nupkg files) in the folder `C:\Users\Admin\Desktop\Training\Nuget\Server`

Click [here](#) to clear the package cache.

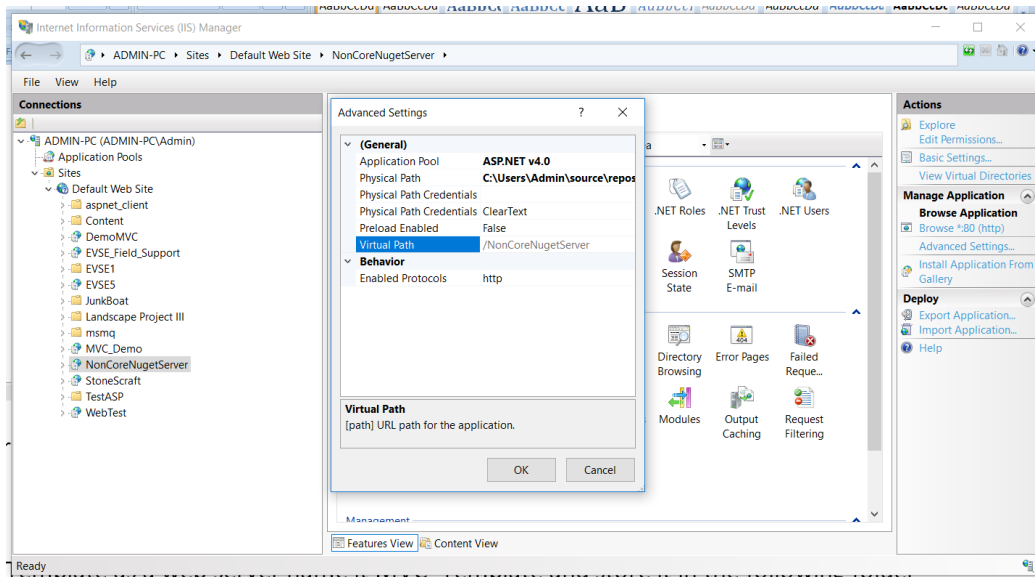
4.0 Set the project into release mode.

5.0 Configure the project properties for a local IIS web application as shown below.

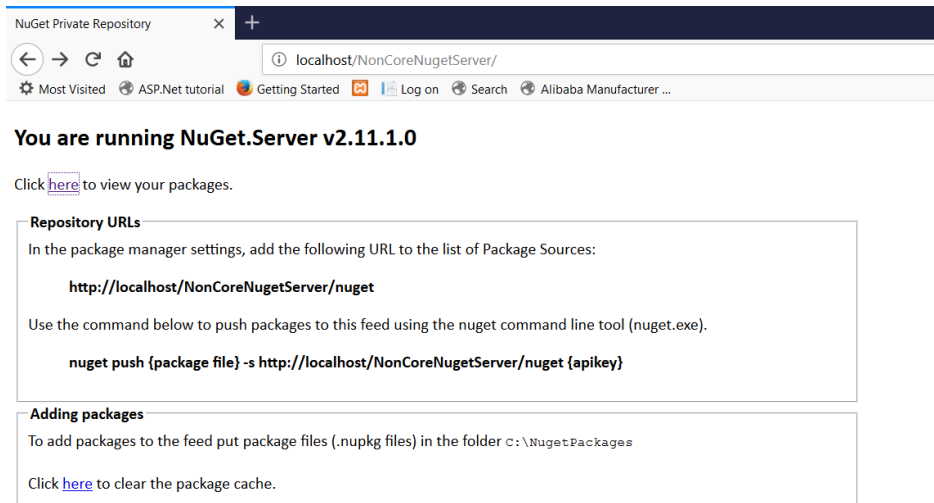




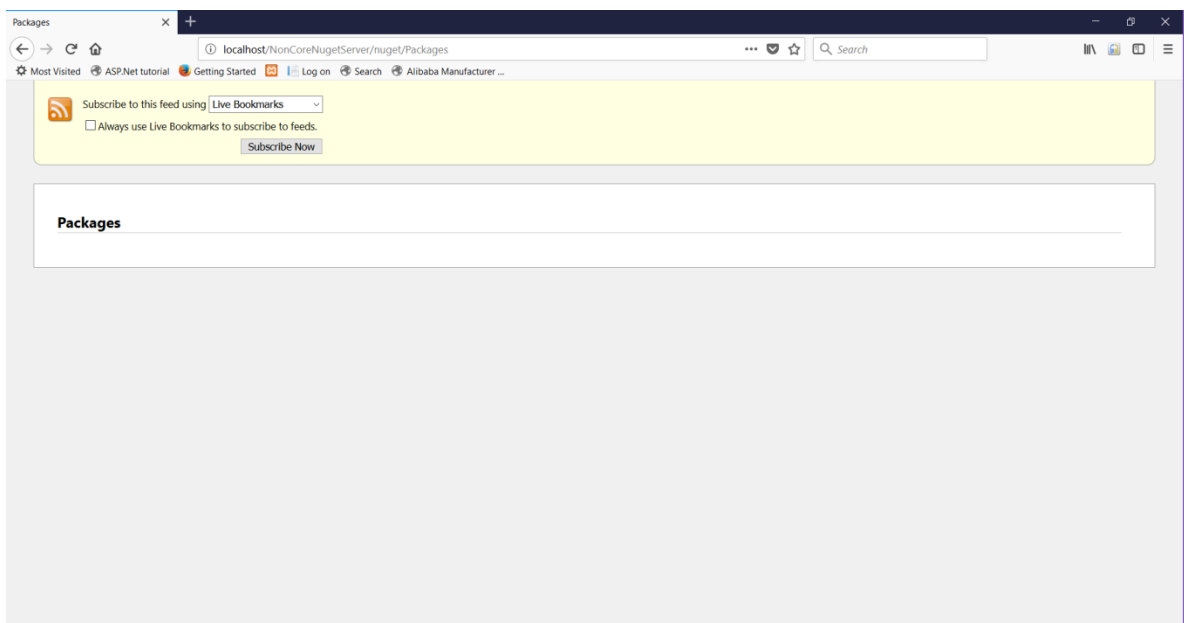
4.0 In the web application verify the following application pool matches matches the figure below.



4.0 In the web browser enter the following url and observe the following displays as shown below.



5.0 select click here link and observe the following screen appears with no packages installed.

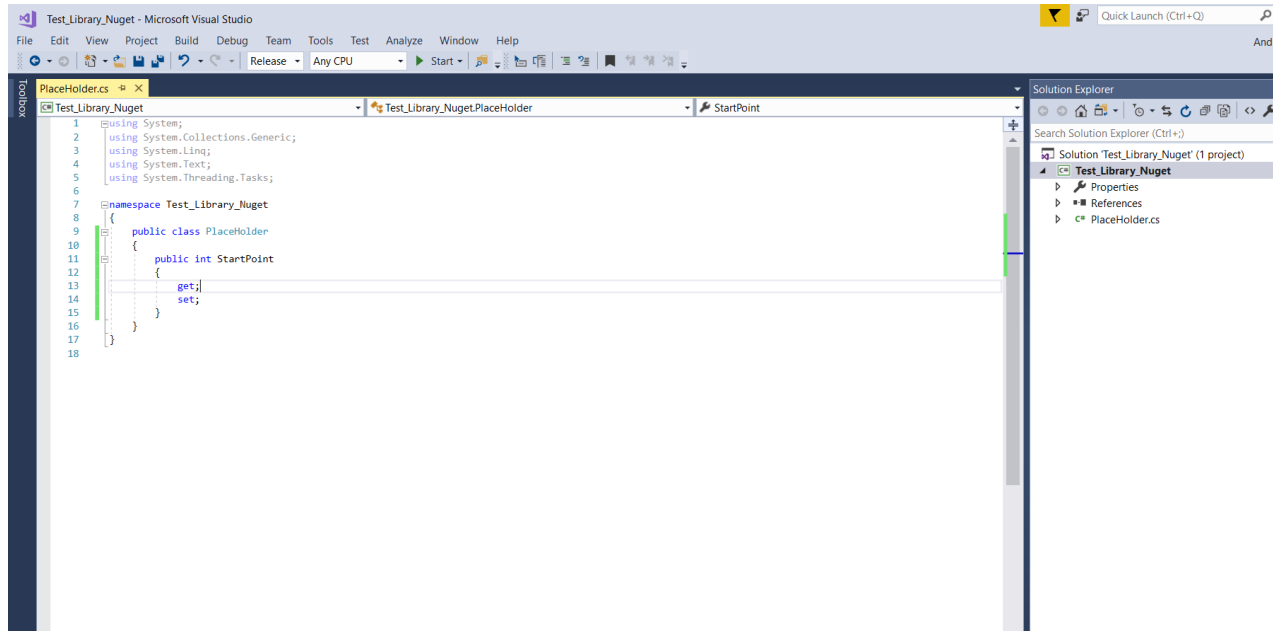


NOTE: THIS SECTION CANNOT BE BROWSED DIRECTLY FROM WITHIN IIS, IT MUST BE ACCESSED IN THE BROWSER URL.

THIS COMPLETES PART I SETTING UP A NUGET SERVER

Part II Create a Nuget Package

1.0 Create a Test Template as a web server name it MVC_Template and store it in the following folder structure: **C:_xCode\Nuget Libraries\Test_Library_Nuget**



2.0 Compile the project in Release Mode, and verify its success.

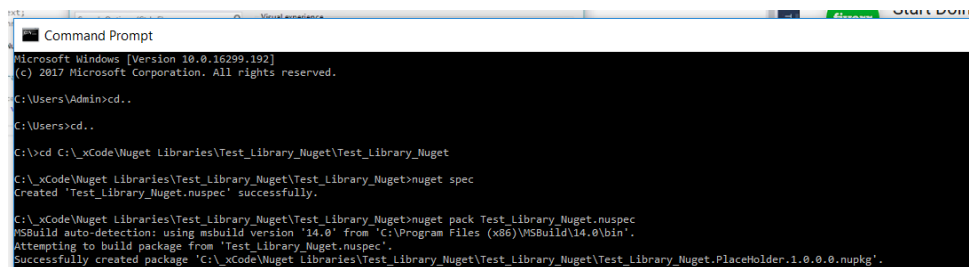
3.0 Open a command window and navigate to the project address.

C:_xCode\Nuget Libraries\Test_Library_Nuget\Test_Library_Nuget

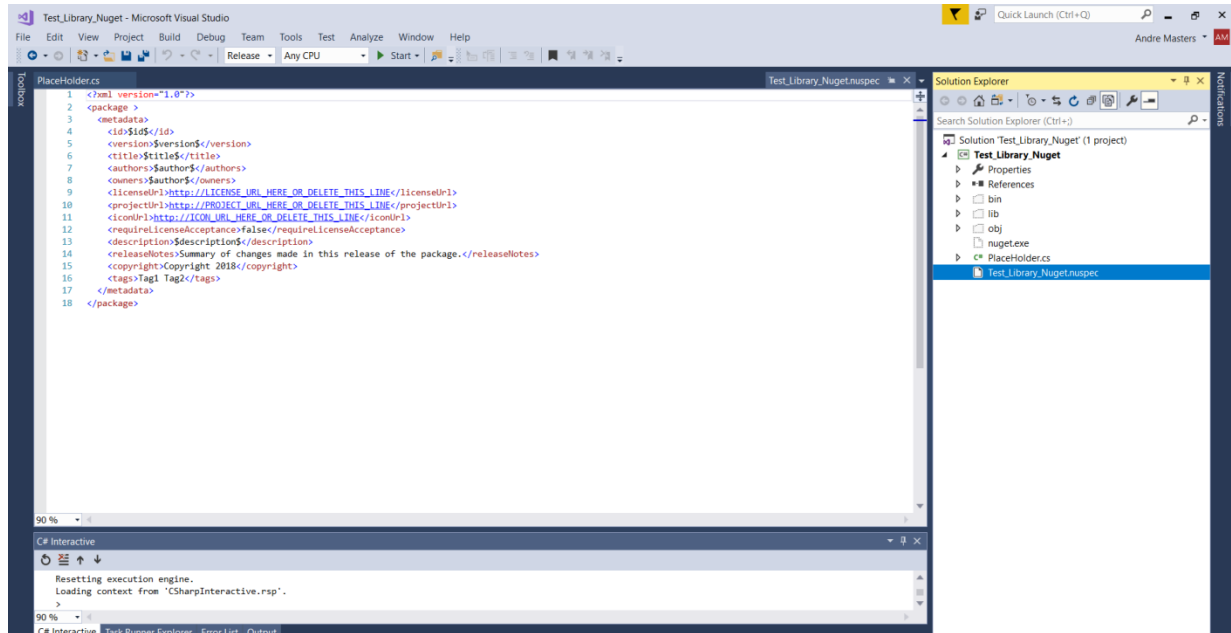
3.0 Observe the following files are present in the Test_Library_Nuget folder.

4.0 Open a command prompt and navigate to the project directory.

5.0 Enter the following Nuget command to create a nuspec file as shown below.

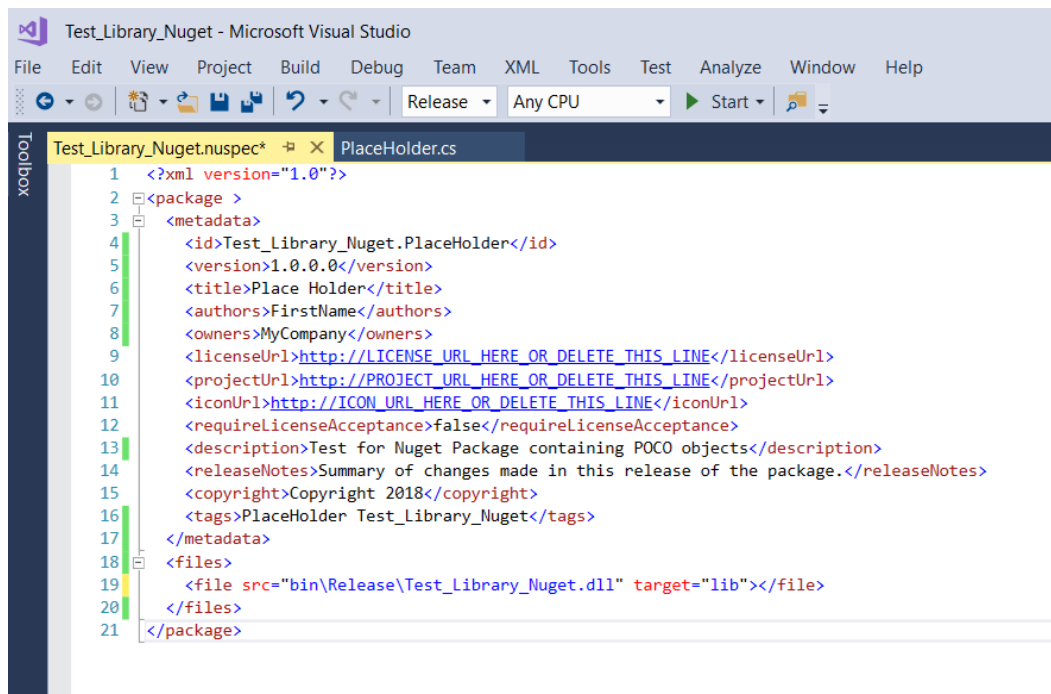


6.0 Include the Nuspec file into the project and open it for editing. Observe the following appears as shown below.



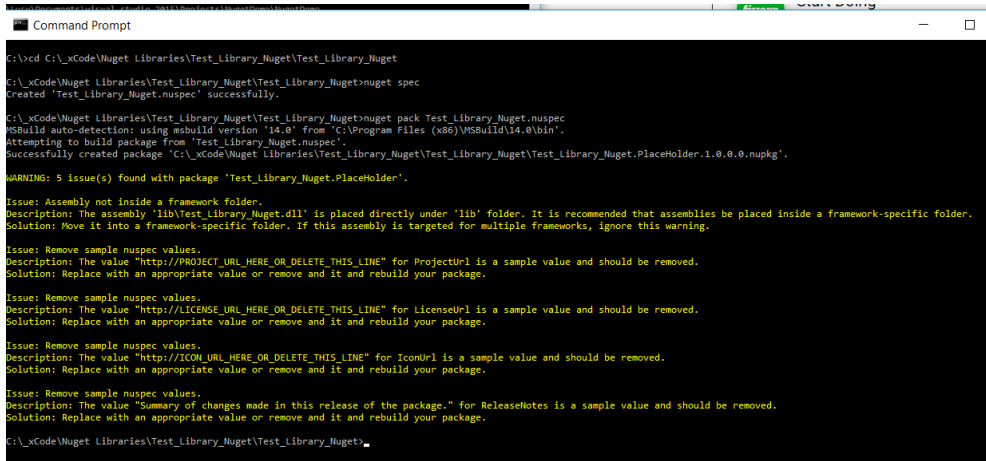
NOTE: To Include the Nuspec file select it RMB -> Select Include in Project option.

7.0 Make the following changes to the Nuspec file so that it matches what is shown in the image below.



8.0 Compile the project in release mode and verify it compiles successfully.

9.0 in the command prompt enter the pack command using the nuspec file requirements as shown in the image below.



```
Command Prompt

C:\>cd C:\_xCode\Nuget Libraries\Test_Library_Nuget\Test_Library_Nuget

C:\_xCode\Nuget Libraries\Test_Library_Nuget\Test_Library_Nuget>nuget spec
Created 'Test_Library_Nuget.nuspec' successfully.

C:\_xCode\Nuget Libraries\Test_Library_Nuget\Test_Library_Nuget>nuget pack Test_Library_Nuget.nuspec
MSBuild auto-detection: using msbuild version '14.0' from 'C:\Program Files (x86)\MSBuild\14.0\bin'.
Attempting to build package from 'Test_Library_Nuget.nuspec'.
Successfully created package 'C:\_xCode\Nuget Libraries\Test_Library_Nuget\Test_Library_Nuget\Placeholder.1.0.0.0.nupkg'.

WARNING: 5 issue(s) found with package 'Test_Library_Nuget.Placeholder'.

Issue: Assembly not inside a framework folder.
Description: The assembly 'lib\Test_Library_Nuget.dll' is placed directly under 'lib' folder. It is recommended that assemblies be placed inside a framework-specific folder.
Solution: Move it into a framework-specific folder. If this assembly is targeted for multiple frameworks, ignore this warning.

Issue: Remove sample nuspec values.
Description: The value "http://PROJECT_URL_HERE_OR_DELETE_THIS_LINE" for ProjectUrl is a sample value and should be removed.
Solution: Replace with an appropriate value or remove and it and rebuild your package.

Issue: Remove sample nuspec values.
Description: The value "http://LICENSE_URL_HERE_OR_DELETE_THIS_LINE" for LicenseUrl is a sample value and should be removed.
Solution: Replace with an appropriate value or remove and it and rebuild your package.

Issue: Remove sample nuspec values.
Description: The value "http://ICON_URL_HERE_OR_DELETE_THIS_LINE" for IconUrl is a sample value and should be removed.
Solution: Replace with an appropriate value or remove and it and rebuild your package.

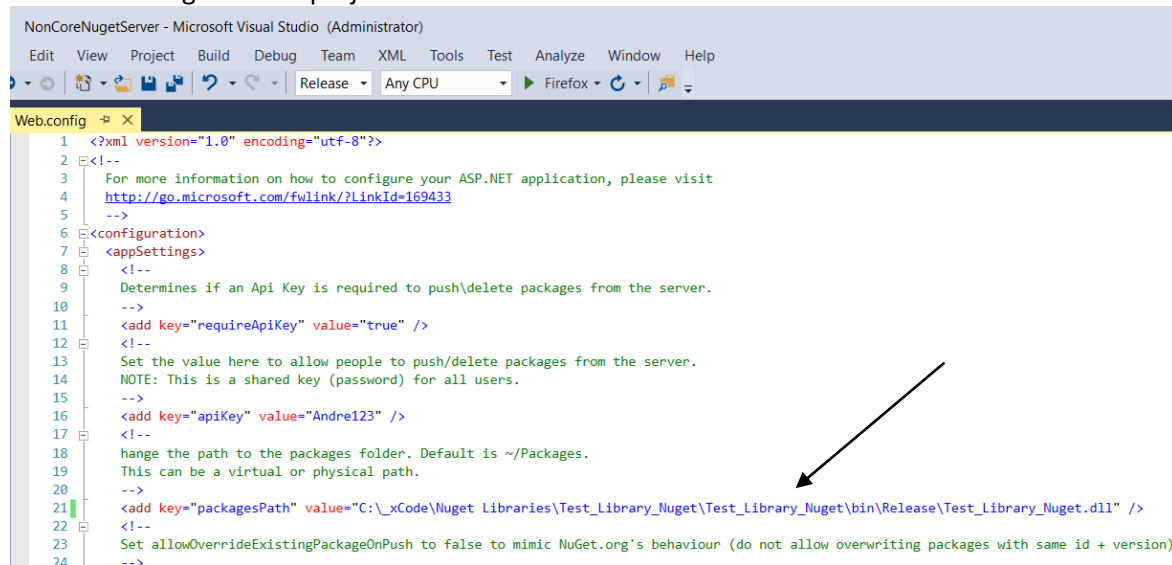
Issue: Remove sample nuspec values.
Description: The value "Summary of changes made in this release of the package." for ReleaseNotes is a sample value and should be removed.
Solution: Replace with an appropriate value or remove and it and rebuild your package.

C:\_xCode\Nuget Libraries\Test_Library_Nuget\Test_Library_Nuget>
```

THIS COMPLETES PART II CREATING A NUGET PACKAGE

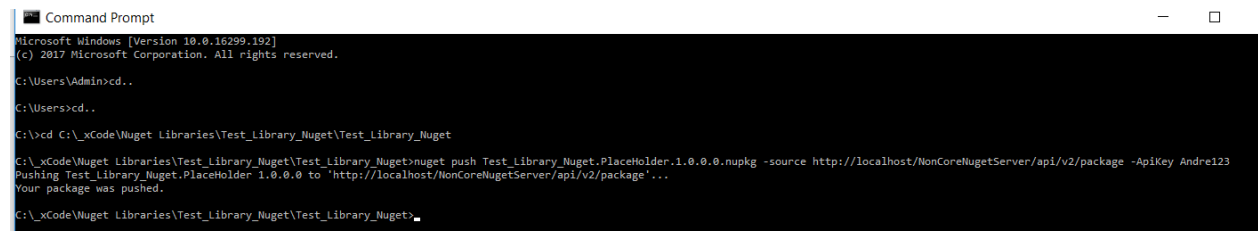
Part III Push a Nuget Package to the Local Nuget Server

10.0 To push a package to the Local Nuget Server, the first step is to update the Web Config file in the Nuget Server project.



```
<?xml version="1.0" encoding="utf-8"?>
<!--
For more information on how to configure your ASP.NET application, please visit
http://go.microsoft.com/fwlink/?LinkId=169433
-->
<configuration>
  <appSettings>
    <!--
    Determines if an Api Key is required to push/delete packages from the server.
    -->
    <add key="requireApiKey" value="true" />
    <!--
    Set the value here to allow people to push/delete packages from the server.
    NOTE: This is a shared key (password) for all users.
    -->
    <add key="apiKey" value="Andre123" />
    <!--
    Change the path to the packages folder. Default is ~/Packages.
    This can be a virtual or physical path.
    -->
    <add key="packagesPath" value="C:\_xCode\Nuget Libraries\Test_Library_Nuget\Test_Library_Nuget\bin\Release\Test_Library_Nuget.dll" />
    <!--
    Set allowOverrideExistingPackageOnPush to false to mimic NuGet.org's behaviour (do not allow overwriting packages with same id + version).
    -->
  </appSettings>
</configuration>
```

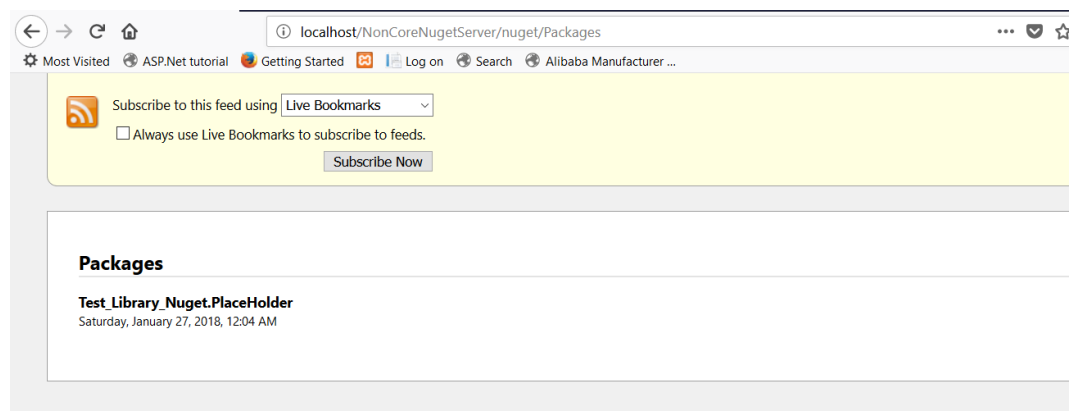
11.0 In the Command prompt push the package to the allowed location on the Nuget Server. The Nuget Server home page details the correct location for package migrations.



```
Microsoft Windows [Version 10.0.16299.192]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd..
C:\Users>cd..
C:\>cd C:\_xCode\Nuget Libraries\Test_Library_Nuget\Test_Library_Nuget
C:\_xCode\Nuget Libraries\Test_Library_Nuget>nuget push Test_Library_Nuget.PlaceHolder.1.0.0.0.nupkg -source http://localhost/NonCoreNugetServer/api/v2/package -ApiKey Andre123
Pushing Test_Library_Nuget.PlaceHolder 1.0.0.0 to 'http://localhost/NonCoreNugetServer/api/v2/package'...
Your package was pushed.
C:\_xCode\Nuget Libraries\Test_Library_Nuget\Test_Library_Nuget>
```

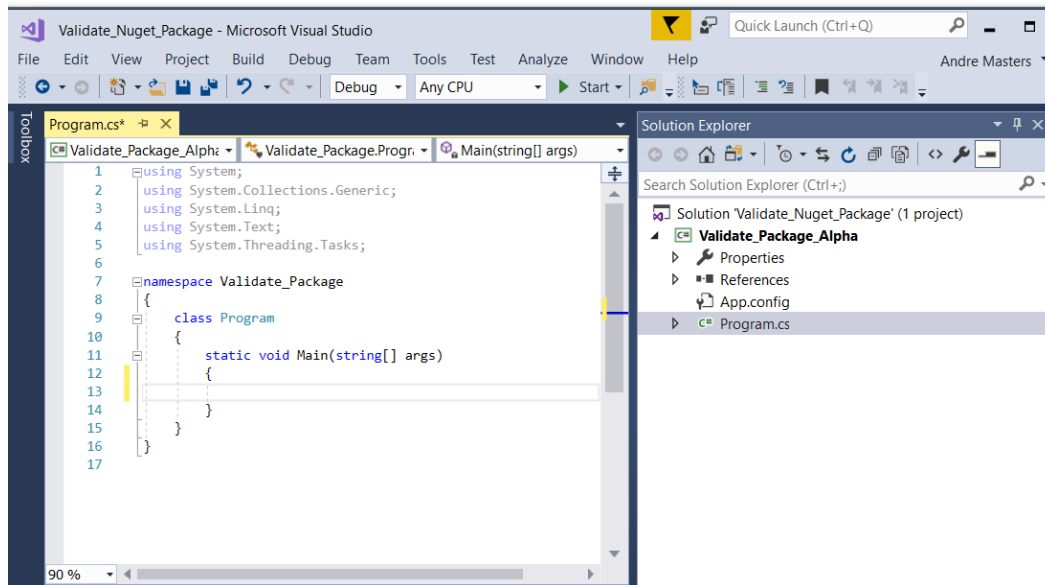
12.0 Verify the package exists by clicking on view packages on the main Nuget server page.



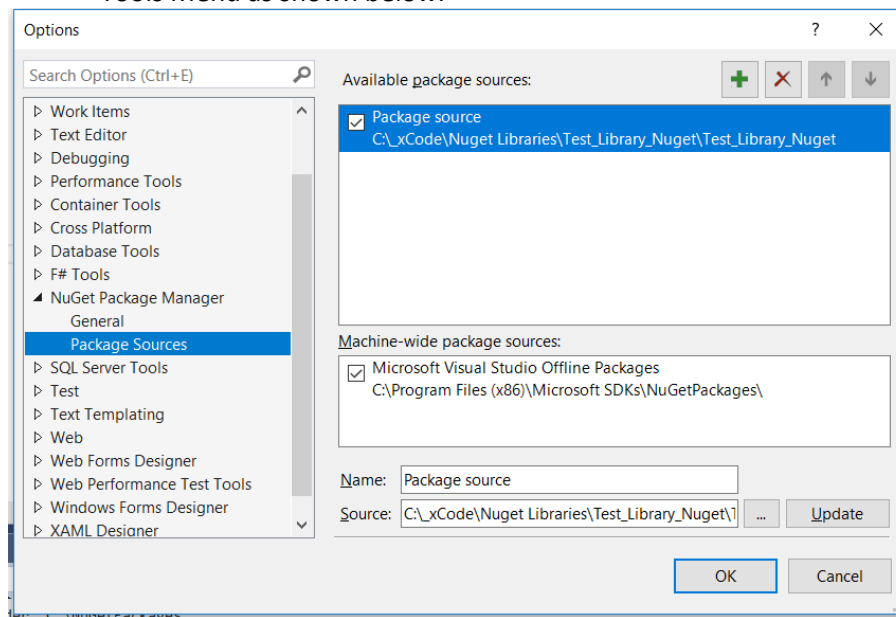
THIS COMPLETES PART III PUSHING A NUGET PACKAGE

Part IV Consume a Nuget Package in a Visual Studio Project

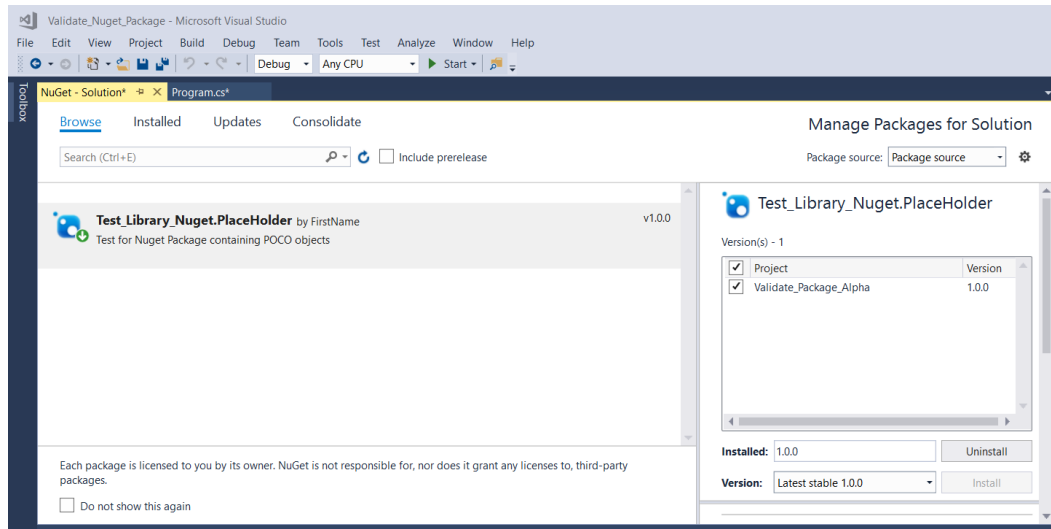
- 13.0 Create a new console application in Visual Studio. Call it Validate_Nuget_Package and store it in _xCode directory.



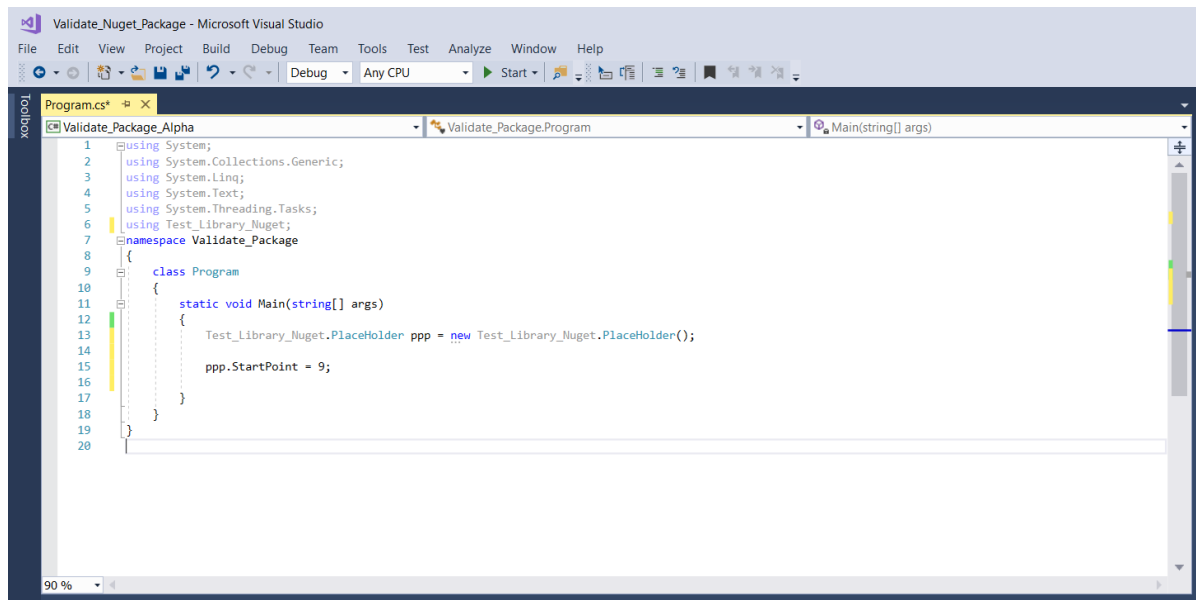
- 14.0 Update the Nuget Settings to include a Nuget Package, by Selecting Nuget Settings under the Tools Menu as shown below.



15.0 Navigate to the package source for the local Nuget packages as shown below.



16.0 Verify the reference was accepted by the project by entering the following code construction as shown below.



THIS COMPLETES PART IV PULLING DOWN A NUGET PACKAGE

