Steps to Customize the Code coverage Analysis in .Net

While performing the Code Coverage analysis for unit test in my ASP .net project, I noticed that code coverage was including the unit test assemblies as well. Principally, the unit test assemblies should not be part of overall code coverage calculation. In the following steps, I will demonstrate how to customize the test coverage settings that enables to pick and choose what files to be included OR excluded from *code coverage* analysis.

I am using VS 2013 for an MVC project and problem I had was that code coverage was calculating coverage for not only project assemblies but also the unit test project assemblies named as *\*.testing.dll.* You will see below how I am excluding the unit test assemblies.

# Create a test configuration (.runsettings) file

Step 1: Open your Visual Studio solution and *right click* on that -> *add new item* -> add a *XML file*

Step 2: Name your project as “<yourProjectName>CodeCoverage.runsettings”. Make sure the extension *.runsettings* is spelled correctly

Step 3: Paste the following in our new *.runsettings* file:

<?xml version="1.0" encoding="UTF-8"?>

<RunSettings>

<!-- Configurations that affect the Test Framework -->

<RunConfiguration>

<!-- Path relative to solution directory -->

<ResultsDirectory>.\TestResults</ResultsDirectory>

<!-- [x86] | x64

- You can also change it from menu Test, Test Settings, Default Processor Architecture -->

<TargetPlatform>x86</TargetPlatform>

<!-- Framework35 | [Framework40] | Framework45 -->

<TargetFrameworkVersion>Framework40</TargetFrameworkVersion>

</RunConfiguration>

<!-- Configurations for data collectors -->

<DataCollectionRunSettings>

<DataCollectors>

<DataCollector friendlyName="Code Coverage" uri="datacollector://Microsoft/CodeCoverage/2.0" assemblyQualifiedName="Microsoft.VisualStudio.Coverage.DynamicCoverageDataCollector, Microsoft.VisualStudio.TraceCollector, Version=11.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a">

<Configuration>

<CodeCoverage>

<ModulePaths>

<Include>

<ModulePath></ModulePath>

</Include>

<Exclude>

<ModulePath>.\*testing\*.dll</ModulePath>

</Exclude>

</ModulePaths>

</CodeCoverage>

</Configuration>

</DataCollector>

</DataCollectors>

</DataCollectionRunSettings>

<!-- Adapter Specific sections -->

<!-- MSTest adapter -->

<MSTest>

<MapInconclusiveToFailed>True</MapInconclusiveToFailed>

<CaptureTraceOutput>false</CaptureTraceOutput>

<DeleteDeploymentDirectoryAfterTestRunIsComplete>False</DeleteDeploymentDirectoryAfterTestRunIsComplete>

<DeploymentEnabled>False</DeploymentEnabled>

</MSTest>

</RunSettings>

In the highlighted text above (*<Exclude>* tag), you can edit the *ModulePath* for files that you want to exclude from being included.

Note: IncludePath – empty ModulePath means **Include All**

ExcludePath –empty ModulePath means **Exclude none**

You can edit other entries as you wish as well.

# Apply the Settings to Code Coverage

Go to menu *Test -> Test Settings -> Select Test Settings File*



In the file open dialog, select *.runsettings* file you just edited.

# Verify

Go to menu *Test -> Analyze Code Coverage -> All Test*

That shall run the Code coverage again for all the assemblies except the excluded assemblies.

References:

<https://technet.microsoft.com/en-us/subscriptions/downloads/jj635153(v=vs.110).aspx#example>

<https://technet.microsoft.com/en-us/subscriptions/downloads/jj635153(v=vs.110).aspx>

Cheers!