

- Download Visual Studio 2017 Community

Visual Studio

Modifying - Visual Studio Community 2017 (15.0.26228.9)

Workloads Individual components Language packs

.NET

- ☐ .NET Core runtime
- ☒ .NET Framework 3.5 development tools
- ☒ .NET Framework 4 targeting pack
- ☒ .NET Framework 4.5 targeting pack
- ☒ .NET Framework 4.5.1 targeting pack
- ☒ .NET Framework 4.5.2 targeting pack
- ☒ .NET Framework 4.6 targeting pack
- ☒ .NET Framework 4.6.1 SDK
- ☒ .NET Framework 4.6.1 targeting pack
- ☐ .NET Native
- ☒ .NET Portable Library targeting pack

Cloud, database, and server

- ☐ Azure Authoring Tools
- ☐ Azure Cloud Services core tools
- ☐ Azure Compute Emulator
- ☐ Azure Data Lake Tools
- ☐ Azure development prerequisites
- ☐ Azure libraries for .NET
- ☐ Azure Mobile Apps SDK
- ☐ Azure Resource Manager core tools
- ☐ Azure Storage AzCopy
- ☐ Azure Storage Emulator
- ☐ Cloud Explorer
- ☒ CLR data types for SQL Server
- ☐ Connectivity and publishing tools
- ☐ Container development tools
- ☒ Data sources and service references
- ☐ Data sources for SQL Server support
- ☐ IIS Express
- ☐ Redgate SQL Search
- ☐ Service Fabric Tools
- ☐ SQL ADAL runtime
- ☐ SQL Server Command Line Utilities
- ☐ SQL Server Data Tools
- ☐ SQL Server Express 2016 LocalDB
- ☐ SQL Server Native Client
- ☐ Web Deploy

Visual Studio

Modifying - Visual Studio Community 2017 (15.0.26228.9)

Workloads Individual components Language packs

Code tools

- ☐ Class Designer
- ☒ ClickOnce Publishing
- ☐ Dependency Validation
- ☐ Developer Analytics tools
- ☐ DGML editor
- ☐ Git for Windows
- ☐ GitHub extension for Visual Studio
- ☐ Help Viewer
- ☐ LINQ to SQL tools
- ☒ NuGet package manager
- ☐ PowerShell tools
- ☐ PreEmptive Protection - Dotfuscator
- ☒ Static analysis tools
- ☒ Text Template Transformation

Compilers, build tools, and runtimes

- ☒ C# and Visual Basic Roslyn compilers
- ☒ C++/CLI support
- ☐ Clang/C2 (experimental)
- ☐ IncrediBuild
- ☒ MSBuild
- ☐ Standard Library Modules
- ☒ VC++ 2015.3 v140 toolset (x86,x64)
- ☒ VC++ 2017 v141 toolset (x86,x64)
- ☐ Visual C++ compilers and libraries for ARM
- ☐ Visual C++ runtime for UWP
- ☒ Visual C++ tools for CMake
- ☒ Windows Universal CRT SDK
- ☐ Windows XP support for C++

Debugging and testing

- ☒ C++ profiling tools
- ☐ JavaScript diagnostics
- ☒ Just-In-Time debugger
- ☒ Profiling tools
- ☐ Testing tools core features

Development activities

- ☐ ASP.NET and web development tools
- ☒ C# and Visual Basic
- ☐ C++ Android development tools
- ☐ C++ iOS development tools
- ☐ F# language support
- ☐ JavaScript and TypeScript language support
- ☐ Mobile development with JavaScript core features

Visual Studio

Installing - Visual Studio Community 2017 (15.0.26228.9)

Workloads Individual components Language packs

- ☐ Node.js support
- ☐ Office Developer Tools for Visual Studio
- ☐ Visual C++ for Linux Development
- ☒ Visual Studio C++ core features
- ☐ Visual Studio Tools for Office (VSTO)
- ☐ Windows Communication Foundation
- ☐ Windows Workflow Foundation
- ☐ Xamarin
- ☐ Xamarin Workbooks

Emulators

- ☐ Google Android Emulator (API Level 23)
- ☐ Intel Hardware Accelerated Execution Manager (HAXM)
- ☐ Visual Studio Emulator for Android
- ☐ Windows 10 Mobile Emulator (Anniversary Edition)

Games and Graphics

- ☐ Cocos
- ☒ Graphics debugger and GPU profiler for DirectX
- ☐ Image and 3D model editors
- ☐ Unity editor
- ☐ Unreal Engine installer
- ☐ Visual Studio Tools for Unity

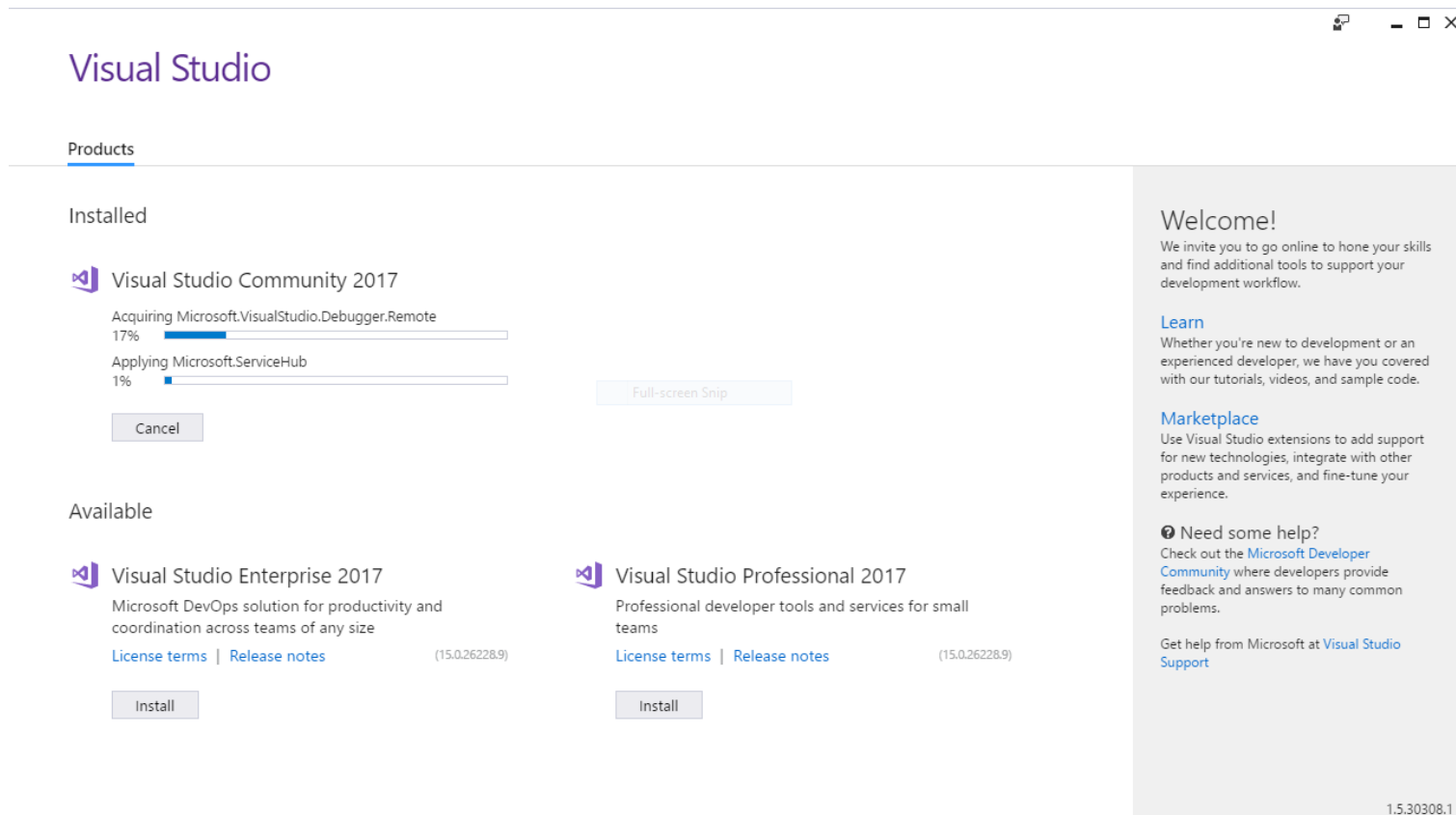
SDKs, libraries, and frameworks

- ☐ Android NDK (R11C)
- ☐ Android NDK (R11C) (32bit)
- ☐ Android NDK (R12B)
- ☐ Android NDK (R12B) (32bit)
- ☐ Android NDK (R13B)
- ☐ Android NDK (R13B) (32bit)
- ☐ Android SDK setup (API level 19 and 21)
- ☐ Android SDK setup (API level 22)
- ☐ Android SDK setup (API level 23)
- ☐ Apache Ant (1.9.3)
- ☐ Blend for Visual Studio SDK for .NET
- ☐ Cordova 6.3.1 toolset
- ☒ Entity Framework 6 tools
- ☒ Graphics Tools Windows 8.1 SDK
- ☐ Java SE Development Kit (8.0.920.14)
- ☐ MFC and ATL support (x86 and x64)
- ☒ Modeling SDK
- ☐ TypeScript 2.0 SDK
- ☐ TypeScript 2.1 SDK
- ☒ Visual C++ ATL support
- ☒ Visual Studio SDK

Location

C:\Program Files (x86)\Microsoft Visual Studio\2017\Community

- Make sure that the highlighted components are selected (required because our main development platform is VS 2015)



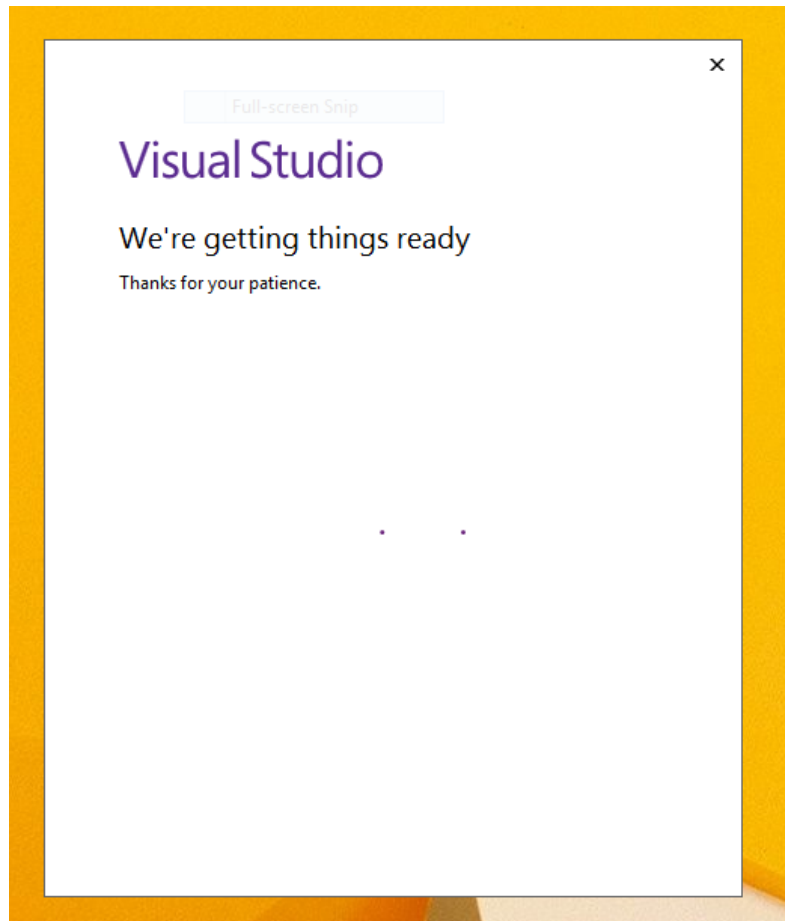
- Visual Studio 2017 is getting installed

Microsoft Build Tools 2015

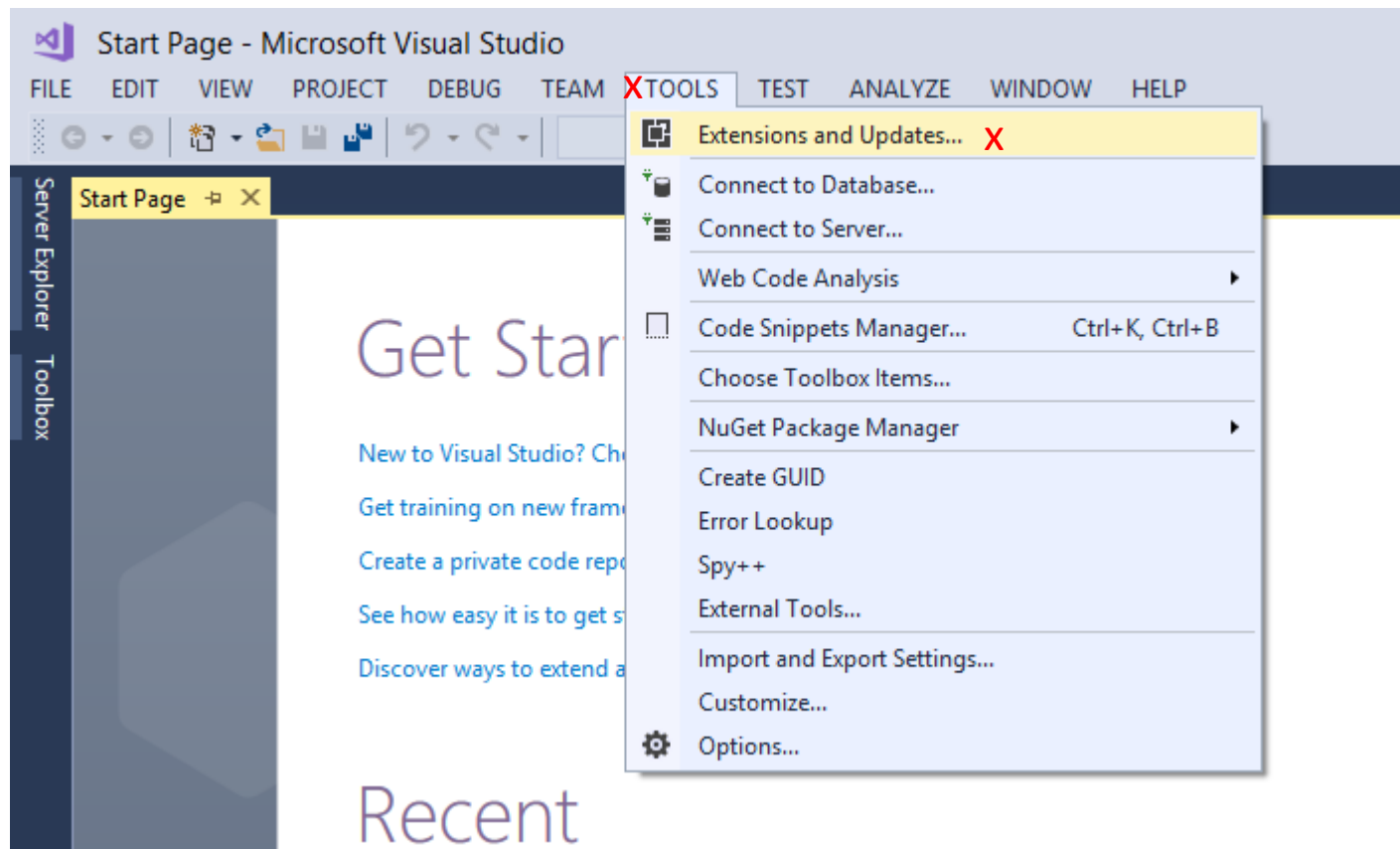
Select Language: 

[Download](#)

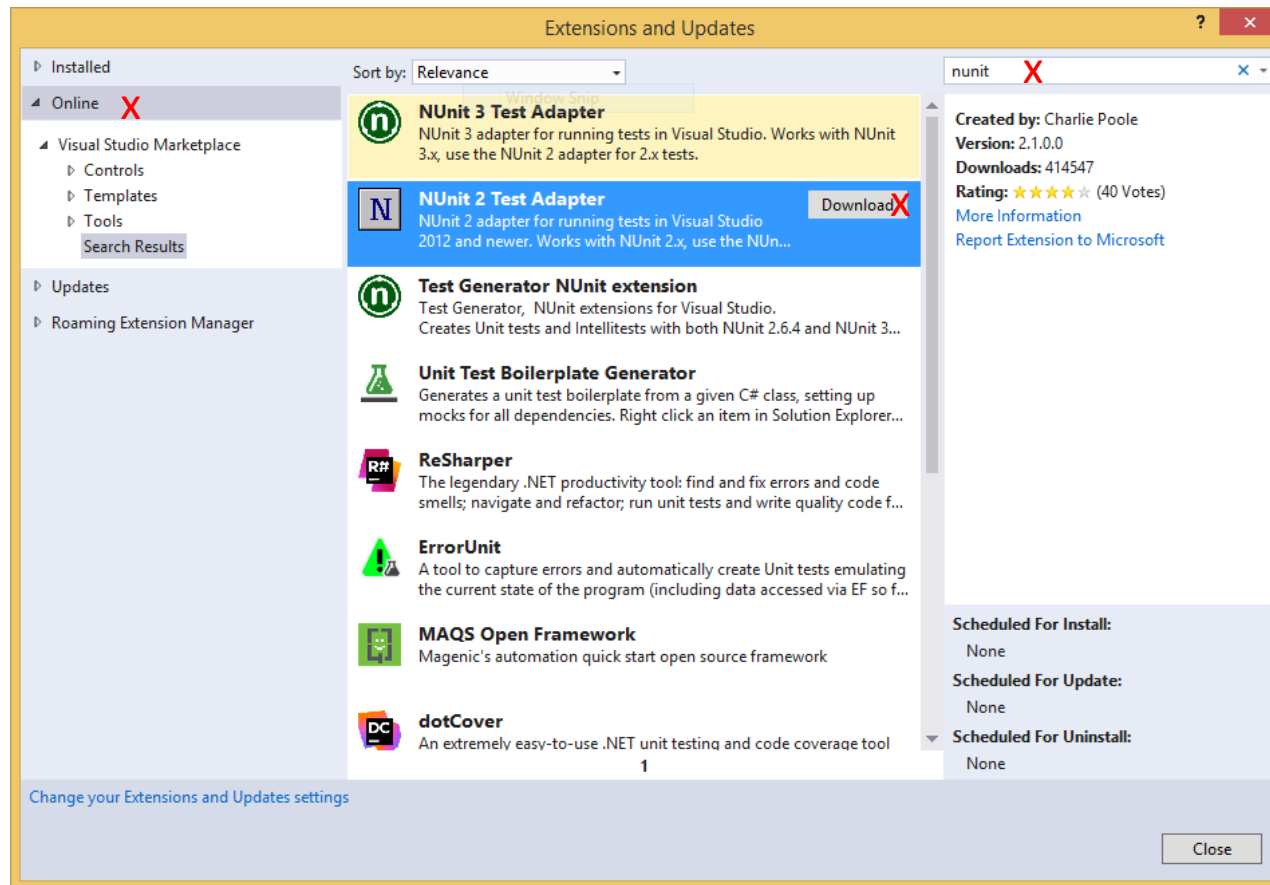
- Download Microsoft Build Tools 2015 from <https://www.microsoft.com/en-us/download/details.aspx?id=48159> (required because our main development platform is VS 2015)



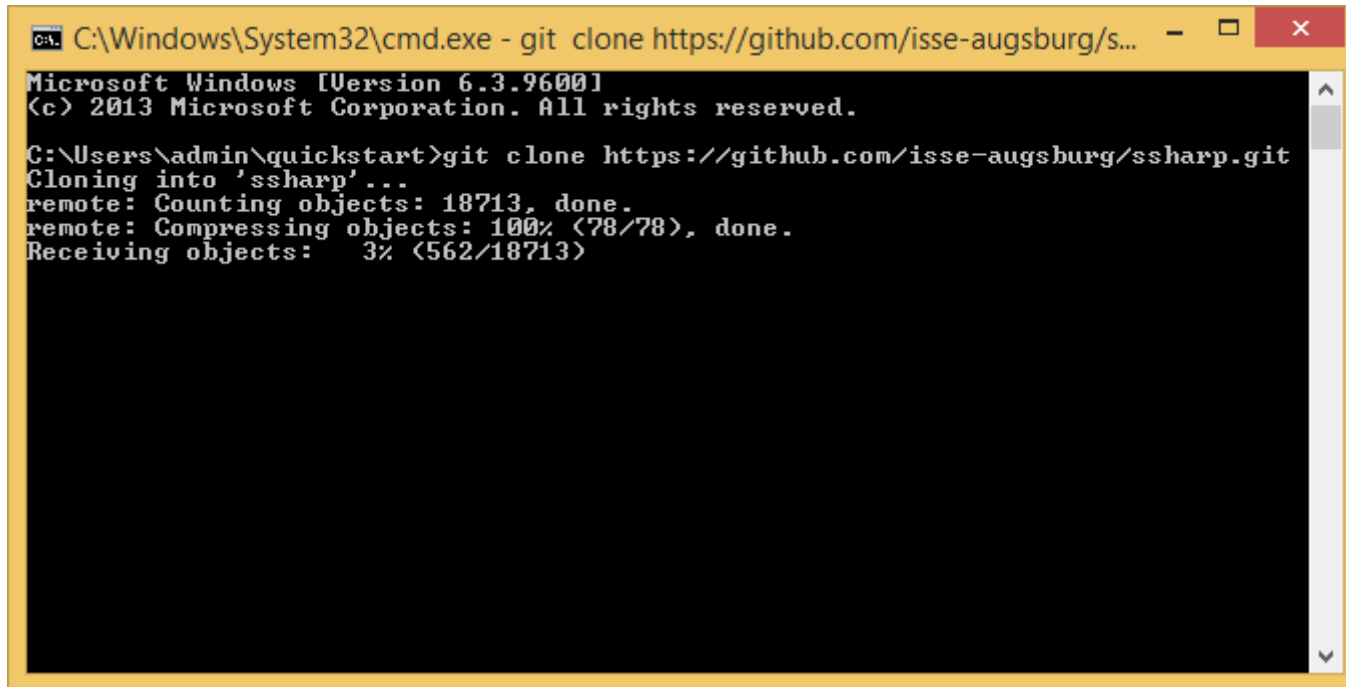
- Start Visual Studio



- Open Tools -> Extensions and Updates



- Select „Online“
- search for „nunit“ on the right
- Download NUnit 2 Test Adapter

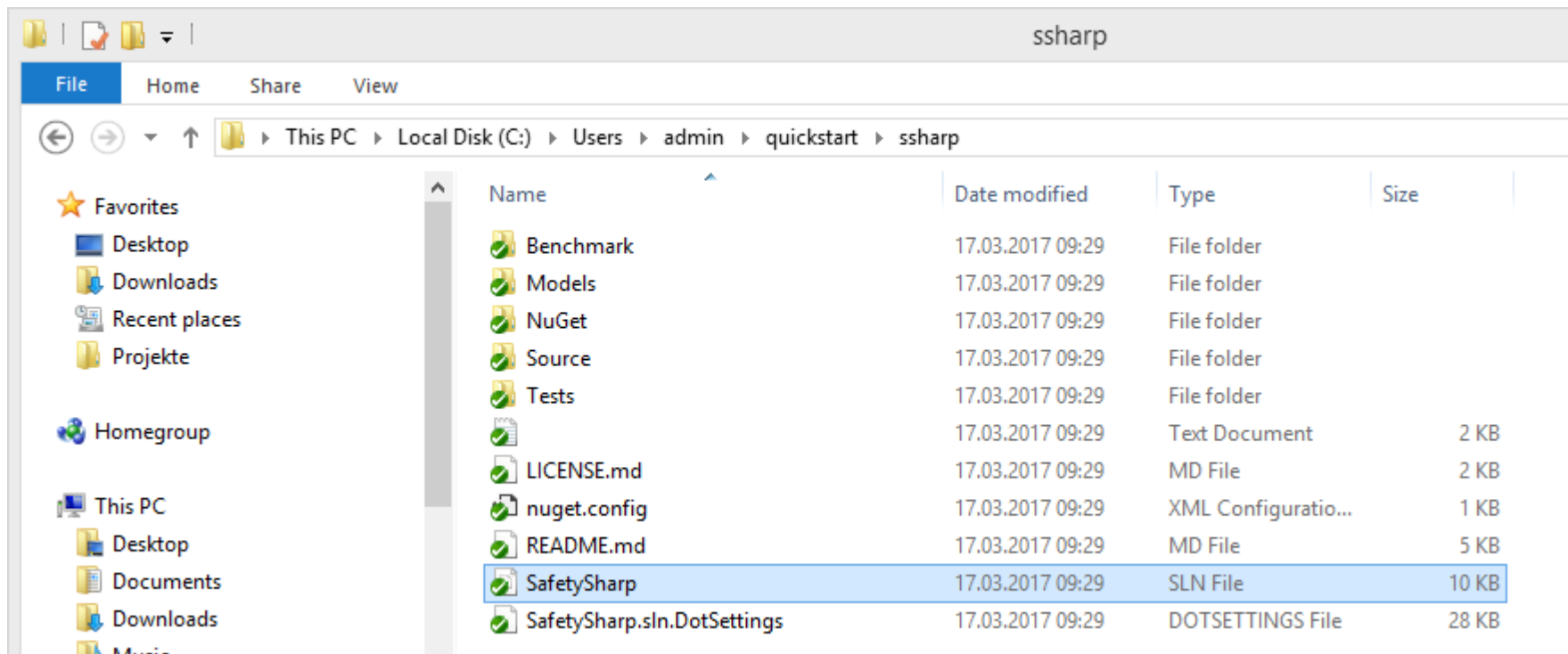


A screenshot of a Windows command prompt window. The title bar shows the path `C:\Windows\System32\cmd.exe` and the command `git clone https://github.com/isse-augsburg/s...`. The window has standard Windows window controls (minimize, maximize, close). The command prompt text is as follows:

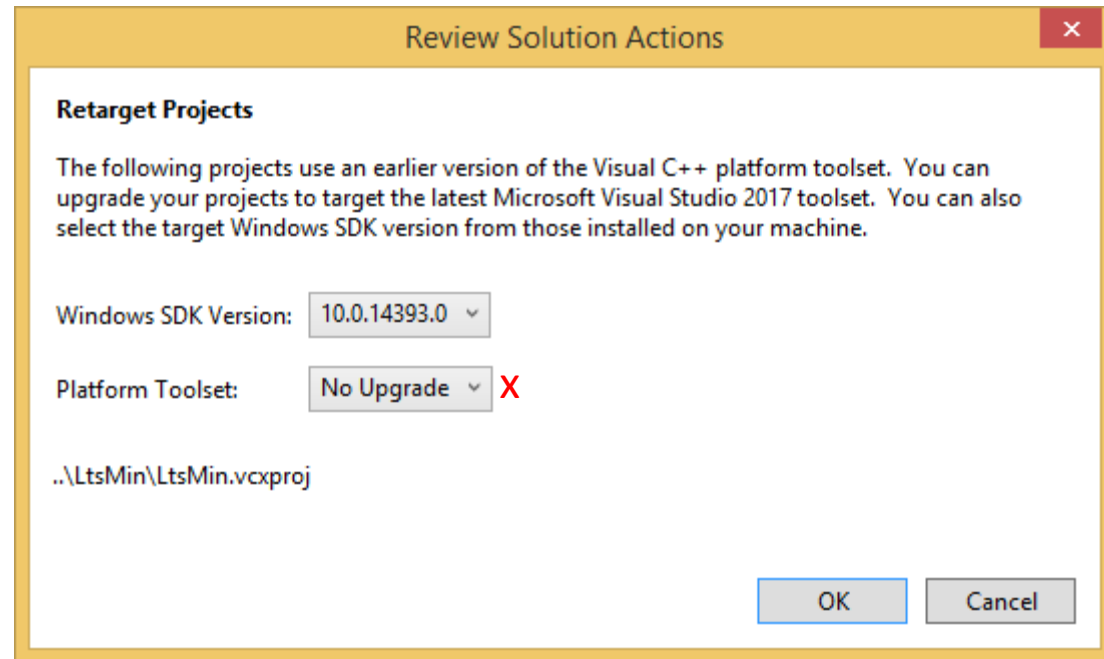
```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\admin\quickstart>git clone https://github.com/isse-augsburg/sssharp.git
Cloning into 'sssharp'...
remote: Counting objects: 18713, done.
remote: Compressing objects: 100% (78/78), done.
Receiving objects: 3% (562/18713)
```

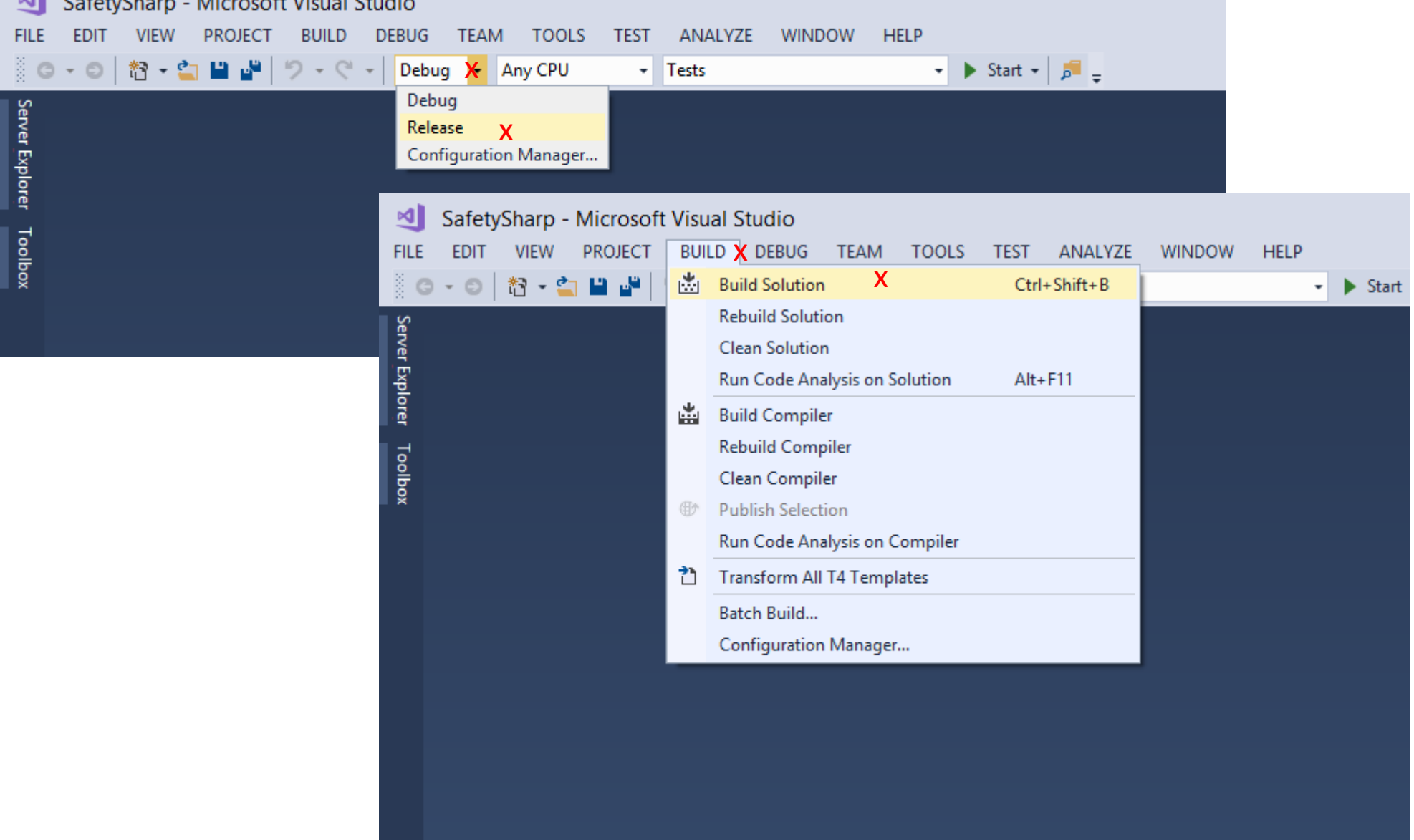
- Download our git repository from <https://github.com/isse-augsburg/sssharp.git>
- You can use your favorite git tool for that



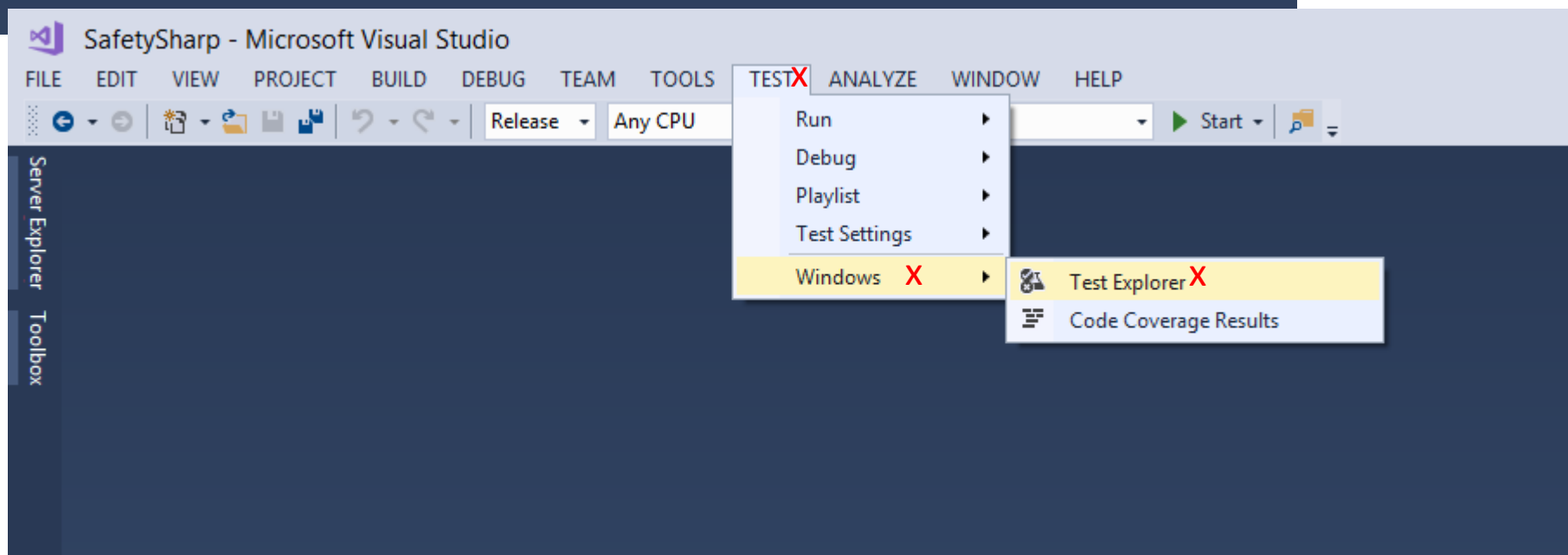
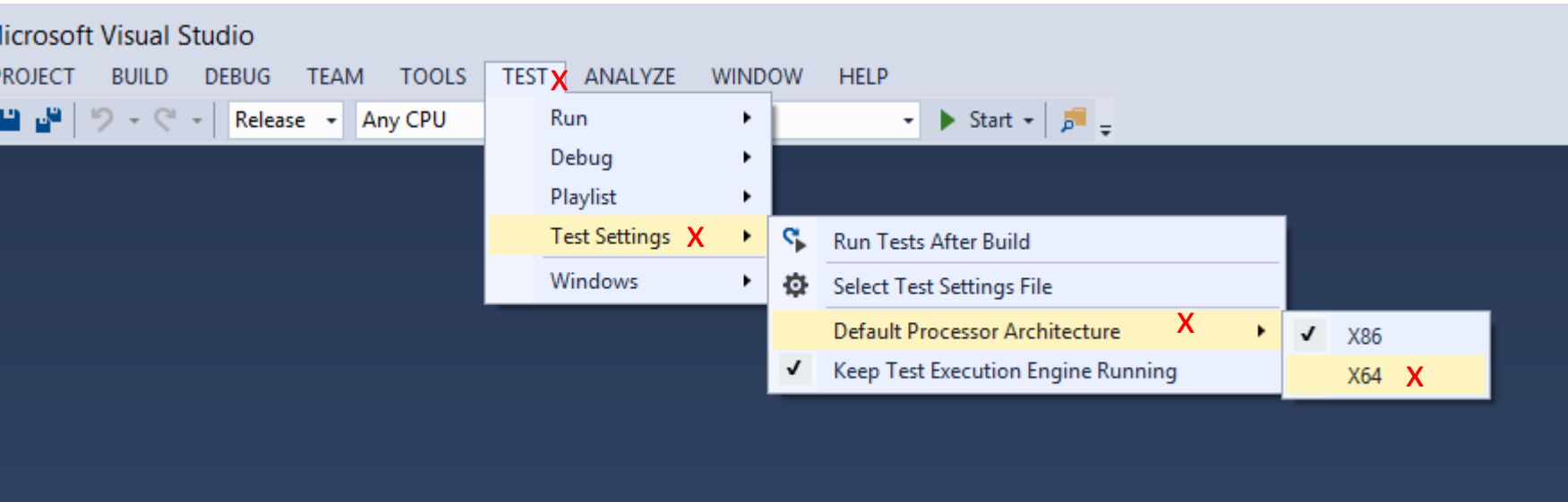
- Open the SafetySharp.sln solution file



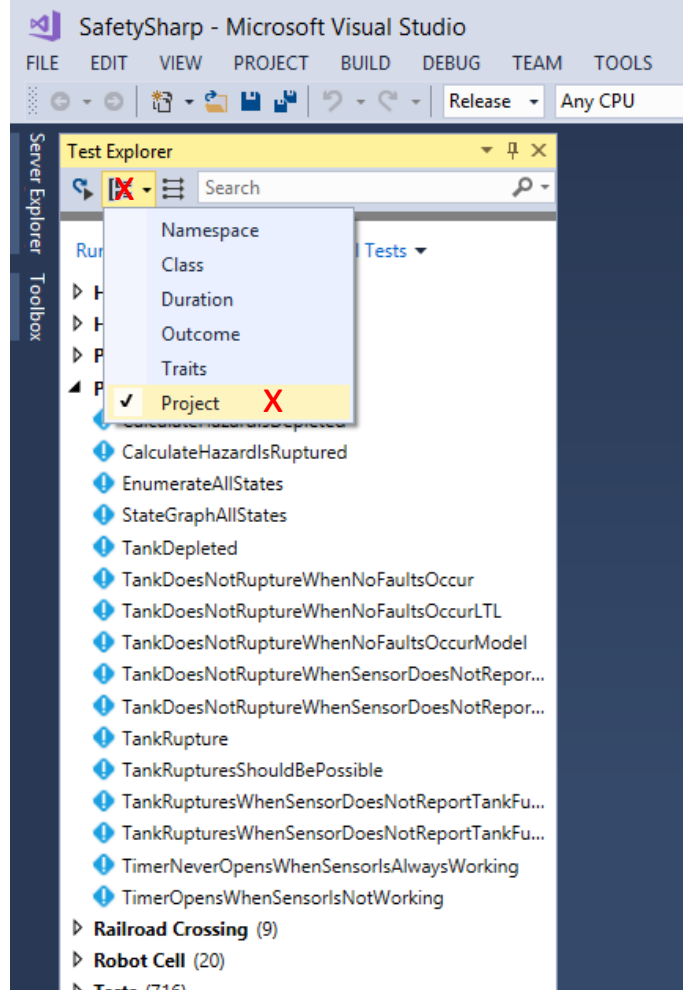
- Select Platform Toolset „No Upgrade“
(required because our main development platform is VS 2015)



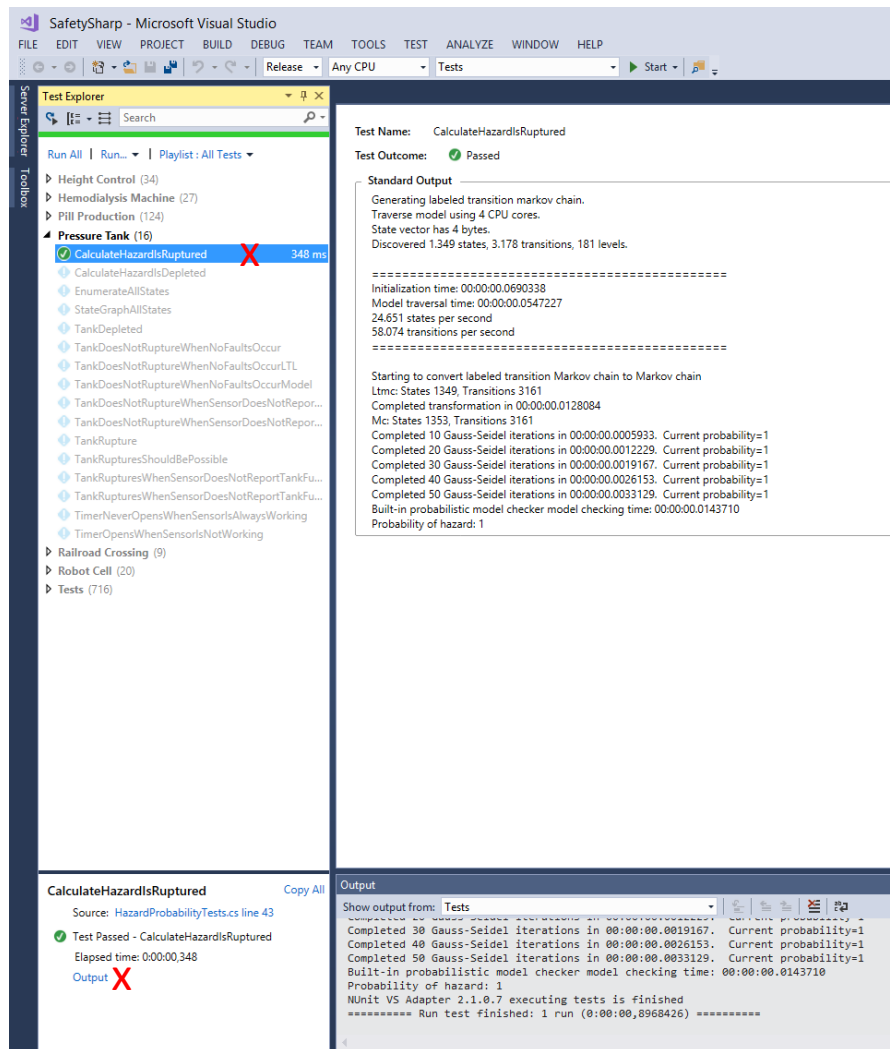
- Select „Release“ and run „Build->Build Solution“ to create an optimized build



- Select X64 as processor architecture for the tests
- Open the Test Explorer



- Sort the test cases by project
- You may need to restart Visual Studio to see all projects in the test explorer



- Right click on „CalculateProbability...,“ and select „run...“ to conduct a quantitative analysis
- The hazard probability is shown in the main windows after clicking „Output“

Pressure Tank (16)

✖ TankDoesNotRuptureWhenNoFaultsOccur... 66 ms

✔ CalculateHazardIsDepleted 331 ms

✔ CalculateHazardIsRuptured 92 ms

✔ EnumerateAllStates 100 ms

✔ StateGraphAllStates 183 ms

✔ TankDepleted 338 ms

✔ TankDoesNotRuptureWhenNoFaultsOccur 41 ms

✔ TankDoesNotRuptureWhenNoFaultsOccur... 60 ms

✔ TankDoesNotRuptureWhenSensorDoesNot... 33 ms

✔ TankDoesNotRuptureWhenSensorDoesNot... 63 ms

✔ TankRupture ✖ 337 ms

✔ TankRupturesShouldBePossible 75 ms

✔ TankRupturesWhenSensorDoesNotReportT... 78 ms

✔ TankRupturesWhenSensorDoesNotReportT... 34 ms

✔ TimerNeverOpensWhenSensorsAlwaysWo... 61 ms

✔ TimerOpensWhenSensorsIsNotWorking 70 ms

▶ Railroad Crossing (9)

▶ Robot Cell (20)

▶ Tests (716)

Of the 4 faults contained in the model,
0 faults are suppressed,
0 faults are forced, and
4 faults are nondeterministically activated.

Checking the empty fault set...

Checking 4 sets of cardinality 1...

Checking 6 sets of cardinality 2...

critical: { SuppressIsFull, SuppressTimeout }

Checking 2 sets of cardinality 3...

=====

Deductive Cause Consequence Analysis: Results

=====

Elapsed Time: 00:00:00.2297876

Fault Count: 4

Faults: SuppressIsEmpty, SuppressIsFull, SuppressPumping, SuppressTimeout

Checked Fault Sets: 13 (81% of all fault sets)

Minimal Critical Sets: 1

Average Minimal Critical Set Cardinality: 2,0

(1) { SuppressIsFull, SuppressTimeout }

=====

Fault Activation Order Analysis: Results

=====

Elapsed Time: 00:00:00.0623714

Order Relationship Count: 1

{ SuppressIsFull, SuppressTimeout }

(1) SuppressIsFull < SuppressTimeout

TankRupture

Source: SafetyAnalysisTests.cs line 45

✔ Test Passed - TankRupture

Elapsed time: 0:00:00.337

Output ✖

Output

Show output from: Tests

Elapsed Time: 0:00:00.0623714

Order Relationship Count: 1

{ SuppressIsFull, SuppressTimeout }

(1) SuppressIsFull < SuppressTimeout

NUnit VS Adapter 2.1.0.7 executing tests is finished

===== Run test finished: 16 run (0:00:02,5326672) =====

- Right click on „CalculateProbability...,“ and select „run...“ to conduct a DCCA analysis
- The minimal cut sets are shown in the main windows after clicking „Output“