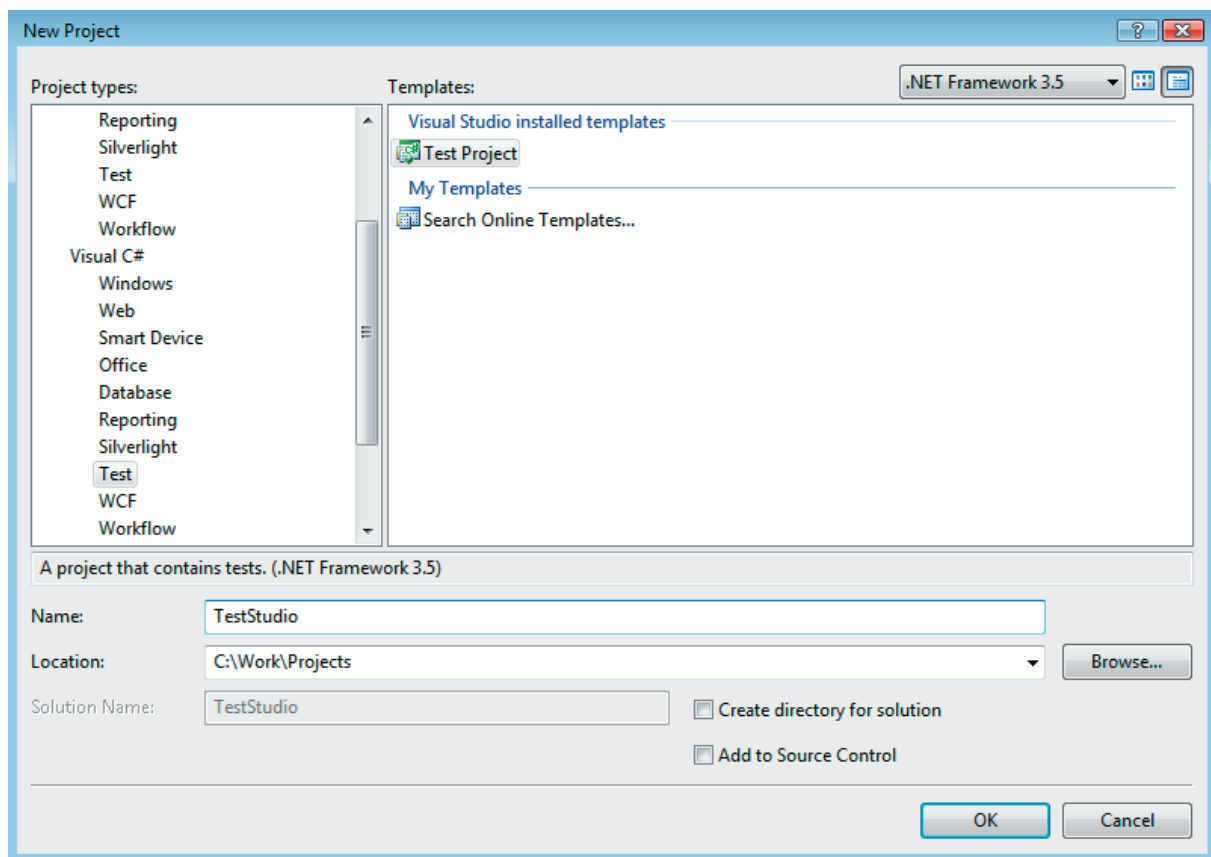


Installing the WebUI TestStudio: What You Should Look For

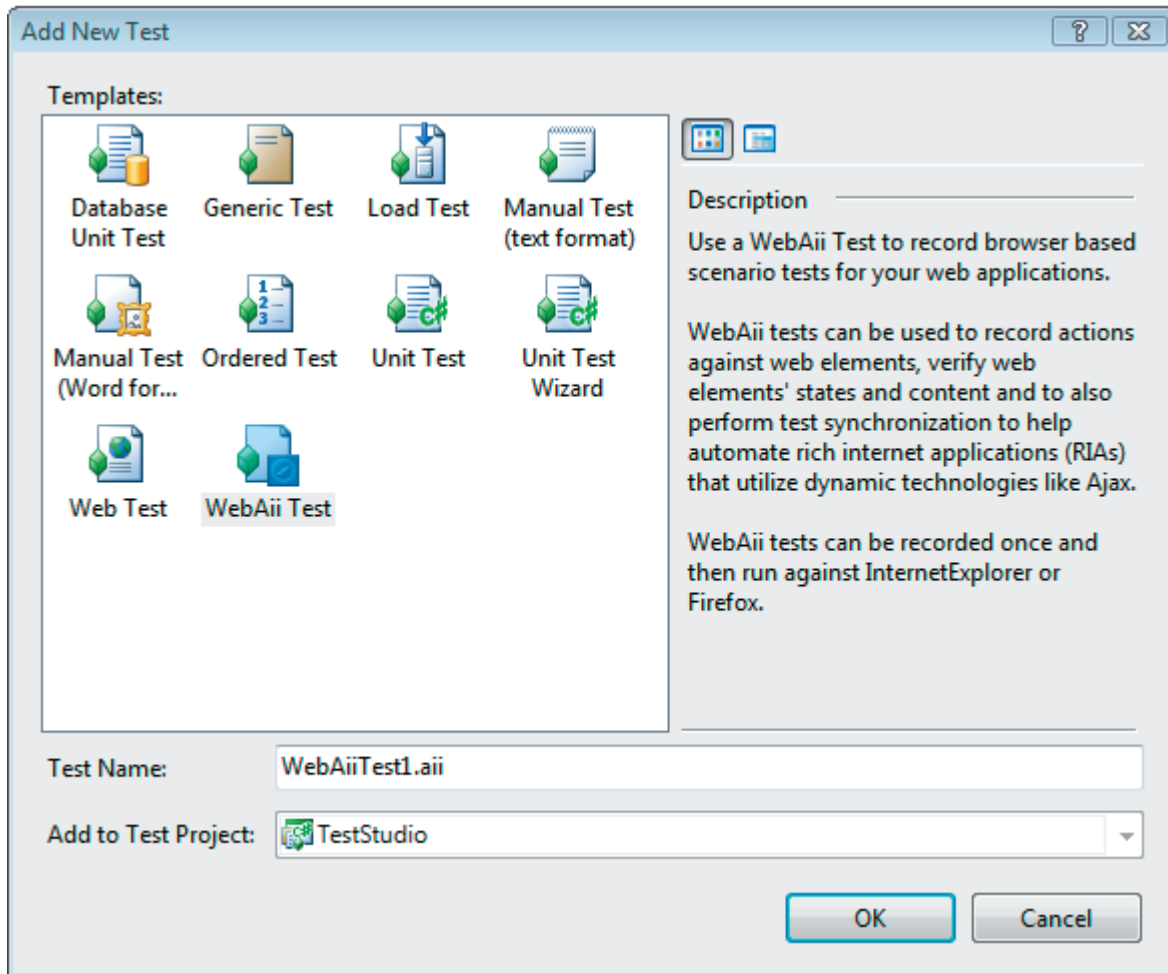
First, you need to have the Visual Studio 2008 SP1 installed in order to proceed further.

The installation of the WebUI TestStudio is a pretty straight-forward process. One can quickly verify everything is up and running properly following the next steps:

1. As soon as the installation is completed create a new VS Test Project (either C# or VB):

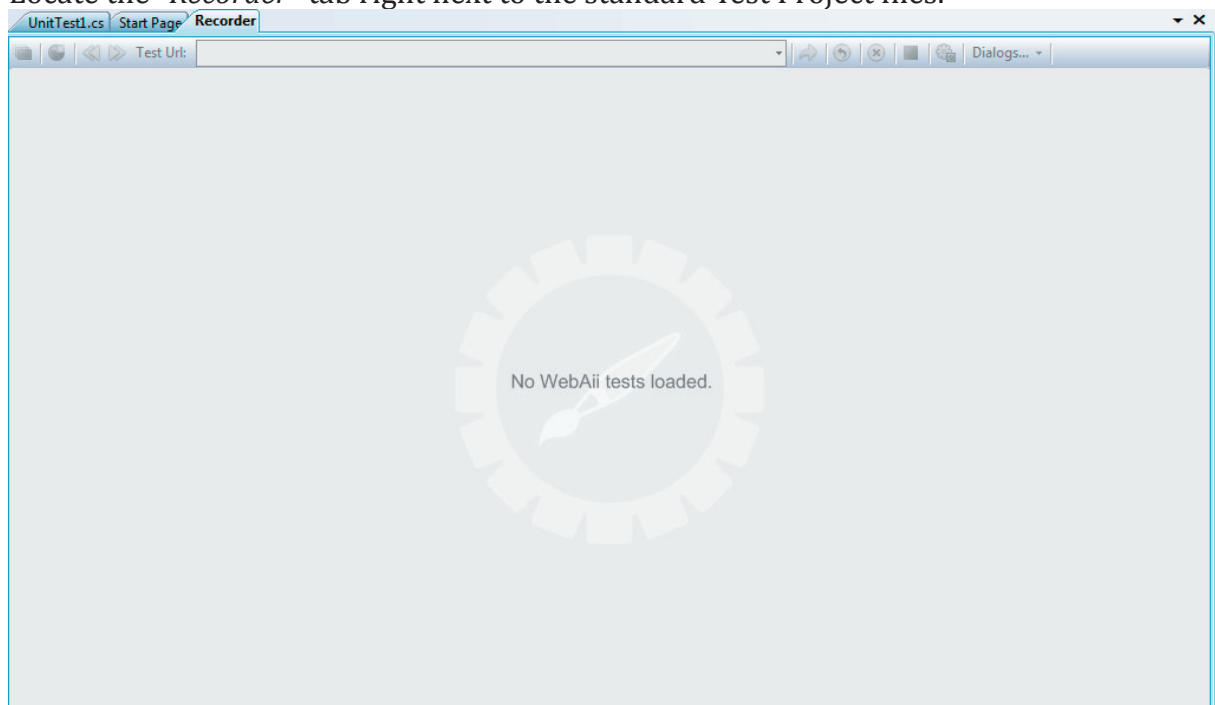


2. Locate the *WebAii Test* Template from the "Add New Test" dialog:



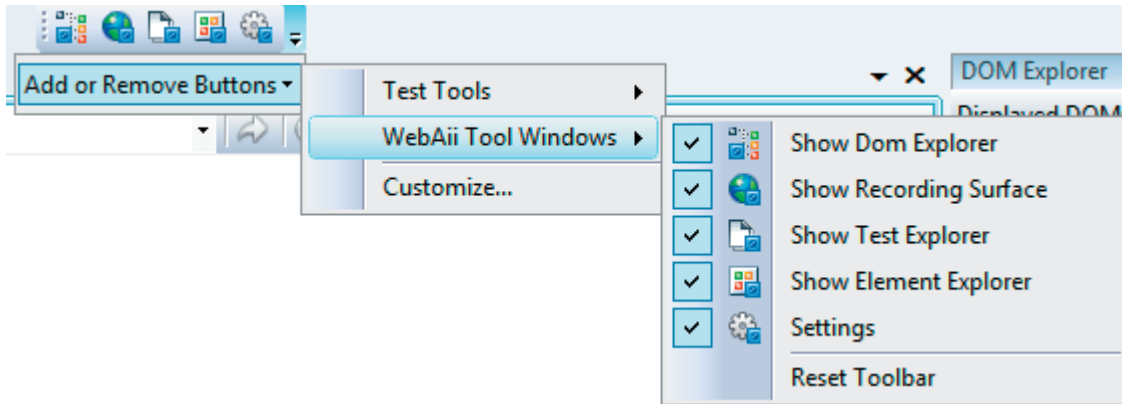
Having this template available means the underlying WebAii framework is installed successfully.

3. Locate the "Recorder" tab right next to the standard Test Project files.

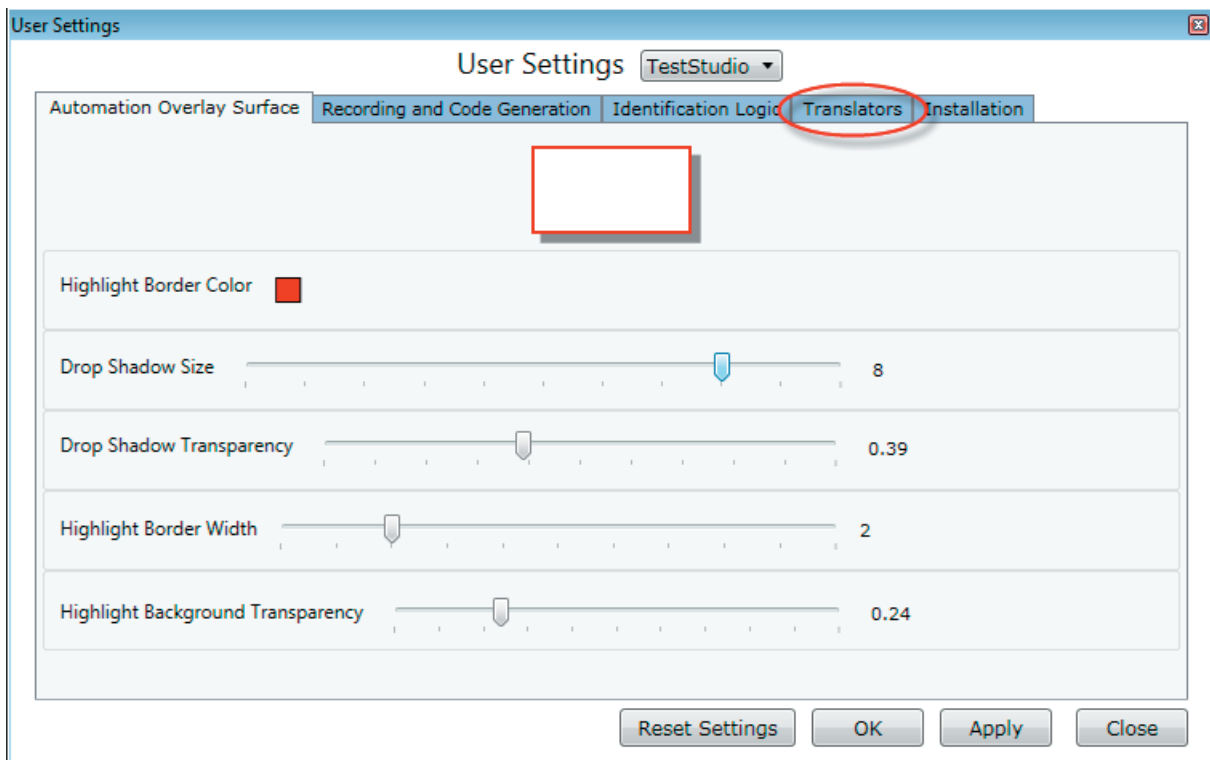


This assures you have the Automation Design Canvas recording tool available as well.

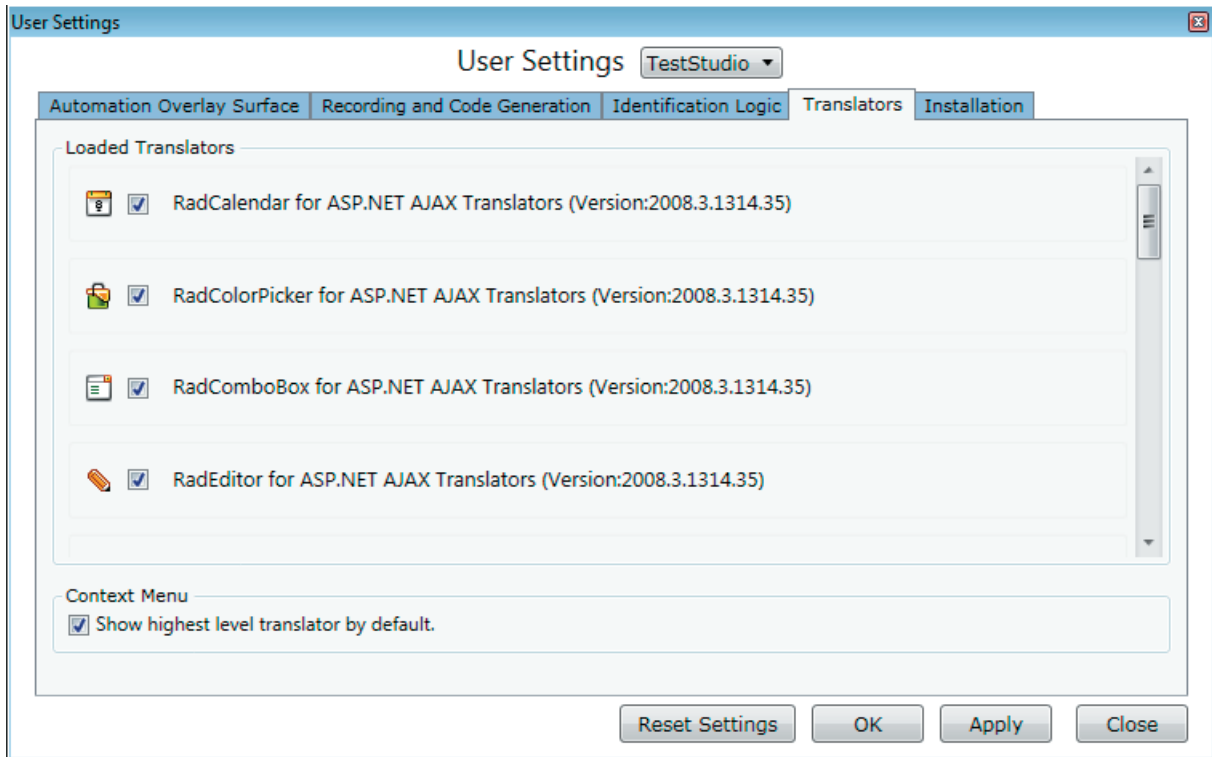
4. Now locate the WebAii ToolBar



and click on the “User Settings” (the last) button. Choose the “Translators” [1] tab.

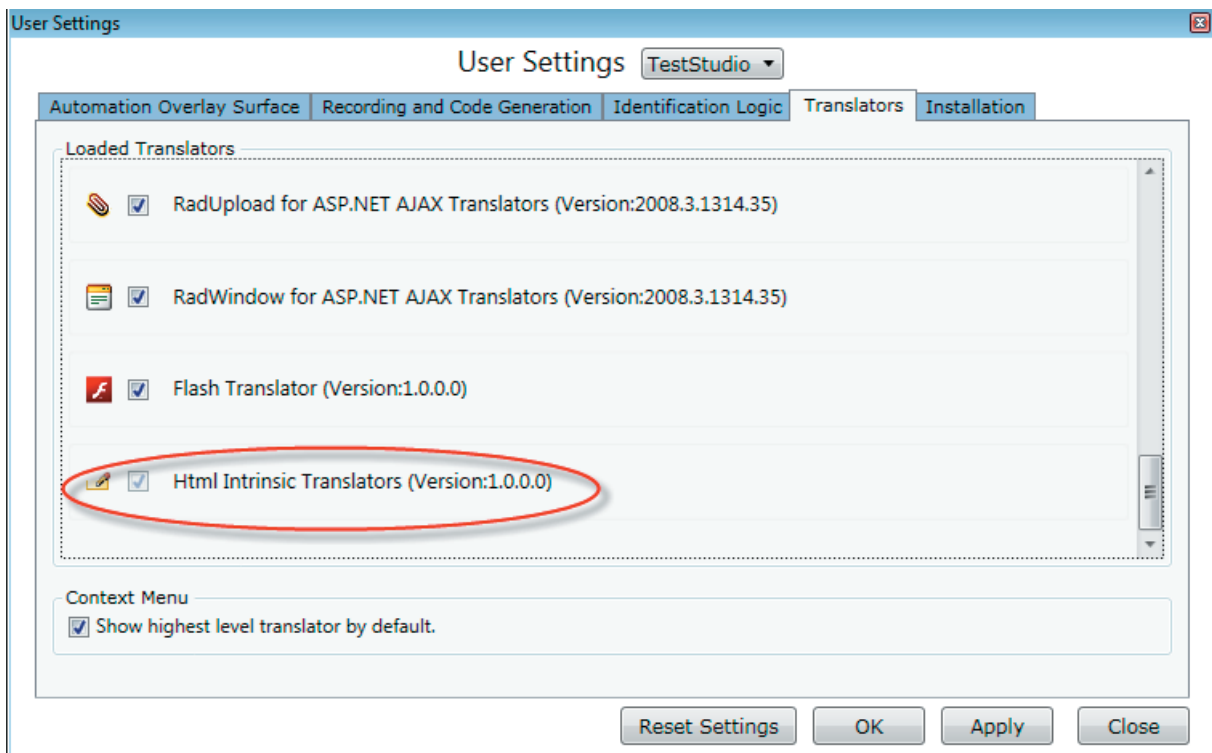


It should display a list of all the available translators the Design Canvas can work with. Make sure the RadControls translators you need are listed:



This means all the WebUI TestStudio components are installed.

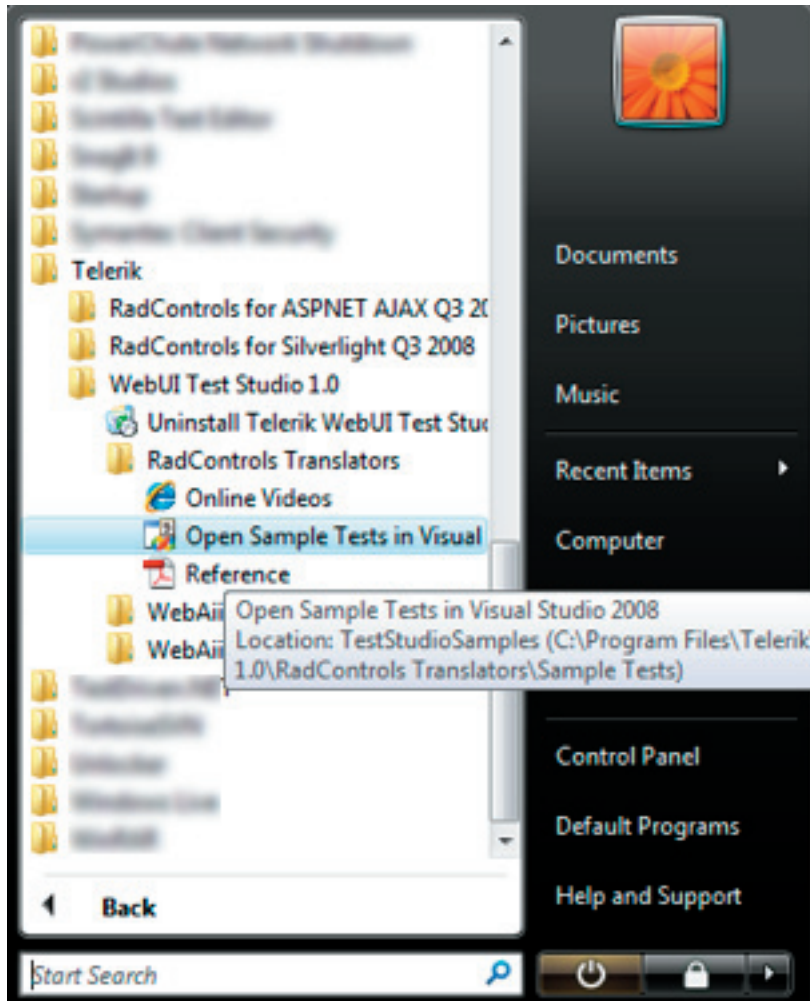
You can optionally disable those translators you may not need. Now scroll down to the bottom of the list:



These *Html Intrinsic Translators* operate with all the base elements in ASP.NET Web Pages. It's recommended to always keep them enabled.

Now load a demo with the RadControls on the page, turn on the element "*Hover over highlighting*" (the first button at the top left corner) and start recording.

There is a sample test project built with the RadControls translators. One can load it via the shortcut from the start menu:



Now browse the folders and open a chosen test in the designer. Make sure the *test explorer* is on so you can see the recorded test steps. Feel free to play with the ready tests (configure, run) or add your own.

[1] '*Translator*' is the term used for the main unit in WebUI TestStudio responsible for the specific element's location, processing the actions and exposing the necessary verifications.