
WinForms to Wisej Migration Whitepaper

Step-by-step guide for migrating WinForms applications to Wisej

This whitepaper describes the steps for migrating an existing WinForms application to Wisej – the first Web Integrated Server Environment for creating Real Time Web Applications. For creating a new Wisej application from scratch, check this tutorial: <http://wisej.com/docs/html/HelloWorldTutorial.htm>. If you are interested in an automatic conversion of your existing applications (WinForms, VB6, or even Visual WebGUI) please contact us at <http://www.fecher.net/contact>

1. Prerequisite: Install the latest version of Wisej (you can get a trial version here: <http://wisej.com/>). You can order licenses from fecher for special prices. Please contact us here: <https://www.fecher.net/contact/>
2. Change the project type from a WinForms executable to a Wisej web project. There are two ways to do this:
 - a. Create a new Wisej project and copy over all the files from the original WinForms application; if you choose this approach then you can skip steps 3 and 4 as the assemblies and the configuration files will be added automatically;
 - b. Change the settings of the existing project. You can do this by editing the .csproj file in a text editor or in Visual Studio (unload the projects before) and make the following changes:
 - Add the project type GUIDs under the ProjectGuid node:

```
<ProjectTypeGuids>{349c5851-65df-11da-9384-00065b846f21};{fae04ec0-301f-11d3-bf4b-00c04f79efbc}</ProjectTypeGuids>
```
 - Change the OutputType from
 - From: `<OutputType>WinExe</OutputType>`
 - To: `<OutputType>Library</OutputType>`
 - Add the import nodes for the web applications under the main Project node:

```
<Import Project="$(VSToolsPath)\WebApplications\Microsoft.WebApplication.targets" Condition="'$(VSToolsPath)' != ''" />
<Import Project="$(MSBuildExtensionsPath32)\Microsoft\VisualStudio\v14.0\WebApplications\Microsoft.WebApplication.targets" Condition="false" />
```

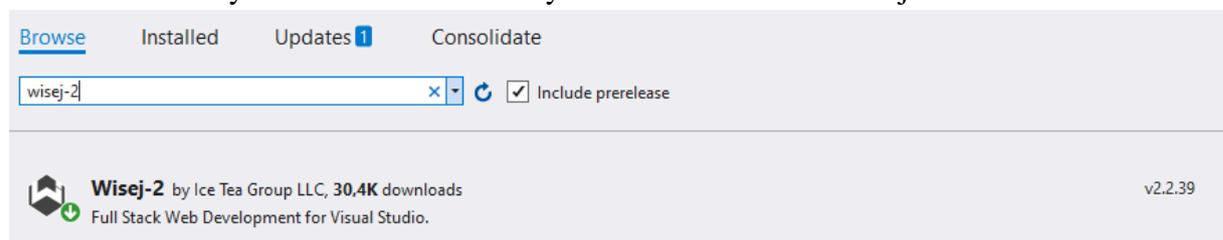
```

1  <?xml version="1.0" encoding="utf-8"?>
2  <Project ToolsVersion="15.0" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
3  <Import Project="$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.Common.props" Condition="Exists('$(MSBuildExtensionsPath)\$(MSBuildT
4  <Import Project="$(VSToolsPath)\WebApplications\Microsoft.WebApplication.targets" Condition="'$(VSToolsPath)' != ''" />
5  <Import Project="$(MSBuildExtensionsPath32)\Microsoft\VisualStudio\v14.0\WebApplications\Microsoft.WebApplication.targets" Condition="false" />
6  <PropertyGroup>
7  <Configuration Condition="'$(Configuration)' == ''">Debug</Configuration>
8  <Platform Condition="'$(Platform)' == ''">AnyCPU</Platform>
9  <ProjectGuid>{126A87C0-E2A7-44F6-B324-3D8FF13E10A4}</ProjectGuid>
10 <ProjectTypeGuids>{349c5851-65df-11da-9384-00065b846f21};{fae04ec0-301f-11d3-bf4b-00c04f79efbc}</ProjectTypeGuids>
11 <OutputType>Library</OutputType>

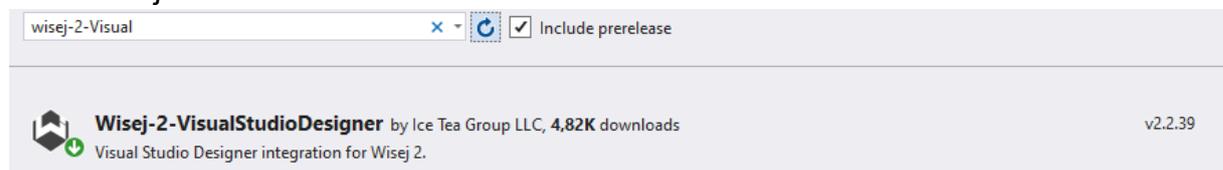
```

- Open the WinForms solution (or reload the projects in Visual Studio) and add the references to the Wisej assemblies: Wisej.Framework and Wisej.Framework.Design.
Note: if these assemblies don't appear in the list check the target .NET framework version; it has to be at least .NET 4.5.

You can manually add the references or by NuGet: browse for “Wisej-2”



and “Wisej-2-Visual”

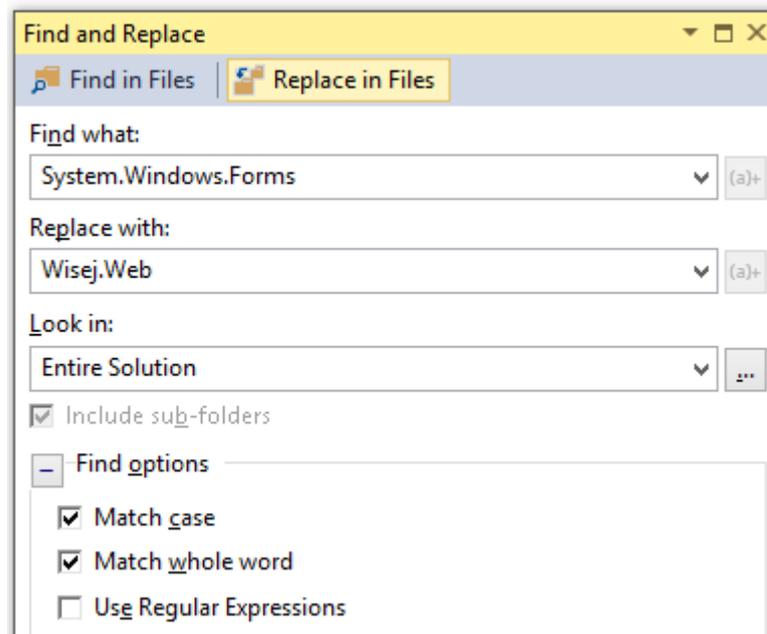


Quick Tip: check briefly the other NuGet packages starting with “Wisej-2”. They can help to improve your Wisej solution.

- Add the configuration files to the project. You will need the following files:
 - Default.html
 - Default.json
 - Web.config

All these files are generated by default when creating a new Wisej project, so you can simply copy them over from a new project. You can find a detailed explanation of these here: <http://wisej.com/docs/html/Configuration.htm>

- Replace all occurrences of System.Windows.Forms with Wisej.Web



6. Build the application and resolve the compiler errors: in most of the cases you will get compiler errors due to some missing properties and/or methods that are obsolete in Wisej or must be replaced by a different property. Most of them you can simply comment out.
7. Change the Main method; in a typical WinForms application you probably had something like this in the Main method (see Program.cs):

```
13 public static void Main()
14 {
15     Application.EnableVisualStyles();
16     Application.SetCompatibleTextRenderingDefault(false);
17     Application.Run(new Login());
18 }
```

The first two lines can be commented out as they are not needed in a web application. The third one must be changed to show instead the Login dialog:

```
13 public static void Main()
14 {
15     //Application.EnableVisualStyles();
16     //Application.SetCompatibleTextRenderingDefault(false);
17     //Application.Run(new Login());
18     Login Login = new Login();
19     Login.Show();
20 }
```

8. Change the startup method in the Default.json file to the Main method from Program.cs:

```
{
  "url": "Default.html",
  //"startup": "WebApplication1.Program.Main, WebApplication1"
  "startup": "John.SocialClub.Desktop.Program.Main, John.SocialClub.Desktop"
}
```

9. That's it! You should be able to run now the Wisej application.