Advanced Web Design

The Clock Project

First Draft

February 19, 2001

Copyright © 2001 by Fred T. Hofstetter. All rights reserved.

The Clock Project

The Clock Project is a quick introduction to building your first Asp.Net page. This page will tell the user what time it is.

Get the environment running by choosing Start \rightarrow Programs \rightarrow Microsoft Visual Studio.NET \rightarrow Microsoft Visual Studio.NET. Note: to make this quicker to launch in the future, you can create a shortcut on your desktop by right-clicking this menu choice, holding the right button down, and dragging this item onto your desktop. From now on, to start Visual Studio.NET, just double-click the shortcut. To make it even quicker to start, you can drag the shortcut onto your Quick Launch toolbar. The shortcut appears as follows:



Once you have Visual Studio.NET running, your screen will appear as follows:



You will now start your first project. Click the link to <u>Create New Project</u>, or pull down the File menu and chose New \rightarrow Project. The New Project dialog appears as follows:

New Project					
Project Types:		<u>T</u> emplates:			8-8- 8-8- 8-8-
✓ Visual Basic Projects ✓ Visual C# Projects ✓ Visual C++ Projects ✓ Visual FoxPro Projects ✓ Setup and Deployment Projects ● Other Projects ✓ Visual Studio Solutions		Windows Application Windows Control Library	Console Application	Class Library	
Create a C# Web Form.					
Name:	ckProject				
Location: http://localhost			•	Browse	
Project will be created at http://localhost/ClockProject.					
▼ More		ОК	Cancel	Help	

Notice how the dialog has two panes. In the Project Types pane, click Visual C# Projects. This indicates that you want to create your project using the C# language. In the Templates pane, click the Web Application icon. In the Name field, make the name of this application be ClockProject. Notice how the message near the bottom of the dialog tells you where the project will be created. In this particular example, the location is http://localhost/Clock. On your screen, the name of your computer will appear in place of localhost.

Click OK. If you get a dialog asking you if you want to create the project with a file share path, click OK.

Wait a few seconds while Visual Studio creates your new project. Notice how your project has a page called WebForm1.aspx. ASPX is the filename extension for ASP.NET pages. At the bottom of the screen, you will see tabs that say Design and HTML. If you click the HTML tab, you will see the HTML code that has been entered onto your page so far. If you click the Design tab, you will see your design, which is blank so far. You will also see a message telling you that you are working in linear layout mode, but you can change to absolute (x and y) positioning if you want to. For now, leave all the settings alone, except for one. The filename Webform1 is not very descriptive of what this file is going to do. Because this file is going to tell the user what time it is, let's change its name to clock.aspx. To do that, look into the Solution Explorer window, which appears along the right side of the screen. The Solution Explorer shows all of the files that have been created in your project so far. To change the name of the

WebForm1.aspx file, click it, and when the cursor appears inside it, change its name to Clock.aspx. If you get a message telling you that Windows cannot rename the file while it is opened, close it by clicking the X in the upper-right corner of the Webform1.aspx window, then rename the file, and then, after you rename it, double-click Clock.aspx in the Solution Explorer to reopen the Clock.aspx file.



Now to create our clock. Click the HTML tab to go into the HTML window. Notice how there are some codes already there, which the wizard created for you automatically. Do not modify any of this code. Position your cursor inside the <body> of the page, and type the following code. The first line will be familiar to you, as it is the standard HTML code for a heading. The second line is a C# statement. Notice how the second line begins with <% and ends with %>. Whatever you type in between <% and %> is considered to be code. Because the language you specified when you created this project is C#, your C# code goes between those marks. The lines to type are:

<h1>The Clock Strikes</h1> The time is <% = DateTime.Now.ToLongTimeString() %> Here is how your page appears after you enter these lines. The newly entered lines are highlighted to show the lines you type. All of the other lines were created automatically by the wizard:



If you click the Design tab, you will see how your heading appears in a bold heading style above the text that says what time it is. The actual time is not given, however. That is because you need to compile and run this page in order to make it active. To do that,

click the Build button, which appears as \square on your Build toolbar. If the Build toolbar is not visible, pull down the View menu and choose Toolbars \rightarrow Build. After your project builds, you will see an Output window telling you whether any errors occurred. If all goes well, the last line of this window will say:

Build: 1 succeeded, 0 failed, 0 skipped

To run your project, click the Start button, which appears as \triangleright on the Standard toolbar. Your browser will launch and run the page, which will tell you the time, as illustrated here:



This little example that you have just completed is only the beginning of what you will become empowered to do with C# and the Visual Studio.NET environment.

As a final touch, let's add a button that the user can click to update the time. To do that, go to the Design view. Mouse over the Toolbox tab at the left of the screen. The toolbox opens as illustrated below:



Notice how one of the tools is a button. Click and drag the button from the toolbox onto your page's design. The button will appear on your page, with the generic name Button. Because this is going to be your Update button, which the user will click to update the time display, let's change its name to Update. In order to do that, click the button once to select it. Its properties are displayed in the Properties window. If you do not see the properties, right-click the button and, when the menu pops out, choose Properties. One of the button's properties is its Text. Right now, the text says "button." Change it to Update, then click in the Design window. Notice how the change takes effect in your design: now you have an Update button.

Click the build button to build your project, and then click the Run button to run it. As illustrated below, your Clock page now has an Update button:

🚰 http://localhost/ClockProject/Clock.aspx - Microsoft Internet Explorer 📃 🗖 🗙
Eile Edit View Favorites Tools Help
📙 🖶 Back 🔻 🤿 🖌 🙆 🖓 🖓 Search 🕋 Favorites 🔇 History 🛛 🛃 🎝 🥙
Address 🔄 http://localhost/ClockProject/Clock.aspx 🔽 🔗 Go 🗍 Links
The Clock Strikes The time is 11:03:31 Update
🖉 Done 🛛 🖉 Local intranet

Click the Update button. Notice how it causes the time to update. Is this some kind of magic? How can it be that the time updates automatically? What did you do to cause this? Well, behind the scenes, the wizard created a handler for your button, and by default, the button handler causes the page to update. If you would like to see the code that the wizard created for you, go to Design view and double-click your Update button. This takes you to the C# code that the wizard created automatically when you put the Update button into your design. Notice how the filename of the code is Clock.cs. The CS filename extension stands for C-Sharp. This is the "code behind the page" that runs on your server when a user access the Clock.aspx page. Do not alter this code right now, just peruse it and think of the power of having this code created automatically for you based on what you do with your design. It will not always be so easy, in more complicated designs you will need to write more of the code yourself, but the wizard always will be working along with you to minimize the work you will need to do.

Note: While you are working with VS.NET, if your build tool is not active, you probably still are debugging. Pull down the Debug menu and choose the option to Stop Debugging. Soon, your build tools will become active again.