V Skills

Skills for a secure future

Certified ASP.NET Programmer

VS-1025

V-Skills Certifications

A Government of India

&

Government of NCT Delhi Initiative

V-Skills

Skills for a secure future

1. INTRODUCING TO ASP.NET & .NET FRAMEWORK 3.5

1.1 Microsoft's .NET 3.5 Framework

The .NET Framework is the base of what geeks call the stack. You can think of the stack as a multilayered cake where layers depend on the layer below for support. The .NET Framework (technically, a compiled portion called the Common Language Runtime, or CLR) sits at the bottom, and its code talks to the underlying operating system, such as Windows Server 2008 and Windows Vista. ASP.NET 3.5 depends on the .NET 3.5 Framework.

You hear geeks refer to classes or class libraries that make up the .NET Framework. They use dot-filled names like System.Web, System.Data and System.Xml. This dot stuff is just a way to organize and categorize thousands of chunks of prewritten code that programmers can tap into via programming languages, such as C#, C++, and Visual Basic.

Microsoft provides tons of reference documentation on everything that's in the .NET Framework. If you still don't find what you need, you can peek into its source code to see how Microsoft makes it all work.

1.2 ASP.NET 3.5

ASP.NET, the next version of ASP, is a programming framework that is used to create enterprise-class Web applications. The enterprise-class Web applications are accessible on a global basis, leading to efficient information management. However, the advantages that ASP.NET offers make it more than just the next version of ASP. ASP.NET is integrated with Visual Studio .NET, which provides a GUI designer, a rich toolbox, and a fully integrated debugger. ASP.NET includes a large number of prebuilt controls, such as text boxes, buttons, images, and data grids, which you can assemble, configure, and manipulate with code to create HTML pages that correctly appear in all popular browsers. This allows the development of applications in a What You See is What You Get (WYSIWYG) manner. Therefore, creating ASP.NET applications is much simpler.

Unlike the ASP runtime, ASP.NET uses the Common Language Runtime (CLR) provided by the .NET Framework. The CLR is the .NET runtime, which manages the execution of code. The CLR allows the objects, which are created in different languages, to interact with each other and hence removes the language barrier. CLR thus makes Web application development more efficient.

In addition to simplifying the designing of Web applications, the .NET CLR offers many advantages. Some of these advantages are listed as follows

- ✓ **Improved performance:** The ASP.NET code is a compiled CLR code instead of an interpreted code. The CLR provides just-in-time compilation, native optimization, and caching. Here, it is important to note that compilation is a two-stage process in the .NET Framework. First, the code is compiled into the Microsoft Intermediate Language (MSIL). Then, at the execution time, the MSIL is compiled into native code. Only the portions of the code that are actually needed will be compiled into native code. This is called Just In Time compilation. These features lead to an overall improved performance of ASP.NET applications.

- ✓ **Flexibility:** The entire .NET class library can be accessed by ASP.NET applications. You can use the language that best applies to the type of functionality you want to implement, because ASP.NET is language independent.

- ✓ **Configuration settings:** The application-level configuration settings are stored in an Extensible Markup Language (XML) format. The XML format is a hierarchical text format, which is easy to read and write. This format makes it easy to apply new settings to applications without the aid of any local administration tools.

- ✓ **Security:** ASP.NET applications are secure and use a set of default authorization and authentication schemes. However, you can modify these schemes according to the security needs of an application.

1.3 ASP.NET Features

ASP.NET introduces two major features: Web Forms and Web Services.

Client vs server script

Developers not familiar with Web development can spend a great deal of time, for example, figuring out how to validate the e-mail address on a form. You can validate the information on a form by using a client-side script or a server-side script. Deciding which kind of script to use is complicated by the fact that each approach has its benefits and drawbacks, some of which aren't apparent unless you've done substantial design work. If you validate the form on the client by using client-side JScript code, you need to take into consideration the browser that your users may use to access the form. Not all browsers expose exactly the same representation of the document to programmatic interfaces. If you validate the form on the server, you need to be aware of the load that users might place on the server. The server has to validate the data and send the result back to the client.

Web Forms

Web Forms simplify Web development to the point that it becomes as easy as dragging and dropping controls onto a designer to design interactive Web applications that span from client to server. ASP.NET Web Forms lets you build dynamic websites using a familiar drag-and-drop, event-driven model. A design surface and hundreds of controls and components let you rapidly build sophisticated, powerful UI-driven sites with data access.

Web Services

A Web service is an application that exposes a programmatic interface through standard access methods. Web Services are designed to be used by other applications and components and are not intended to be useful directly to human end users. Web Services make it easy to build applications that integrate features from remote sources. For example, you can write a Web Service that provides weather information for subscribers of your service instead of having subscribers link to a page or parse through a file they download from your site. Clients can simply call a method on your Web Service as if they are calling a method on a component installed on their system — and have the weather information available in an easy-to-use format that they can integrate into their own applications or Web sites with no trouble.

ASP.NET Event Model

One of the key features of ASP.NET is that it uses an event-based programming model. When an event is raised, the handler for that specific event is executed.

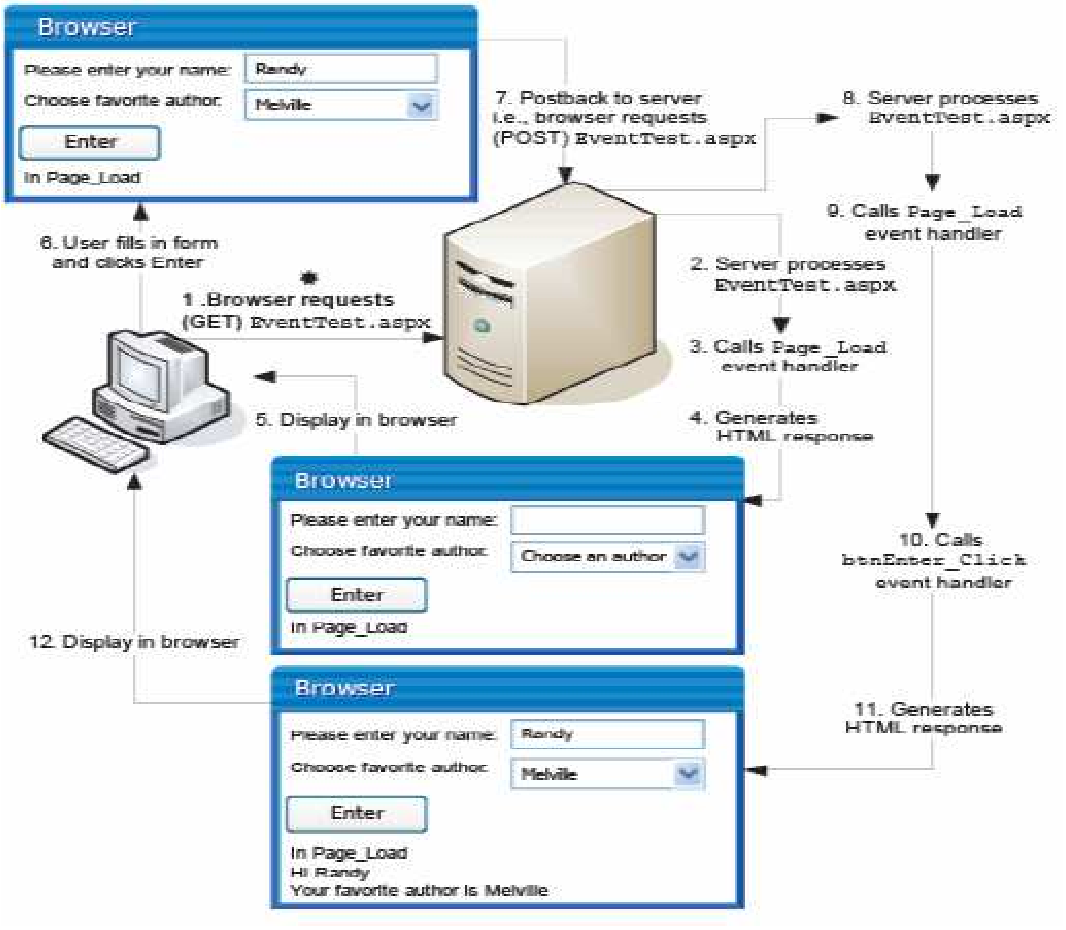
Event Handler

An event handler is a method that determines what actions are performed when an event occurs, such as when the user clicks a button or selects an item from a list. The event system in ASP.NET operates in a different manner than in a Windows application or from the event system in browser-based Javascript.

Postback

Postback is the process by which the browser posts information back to itself. That is, posts information back to the server by requesting the same page. Postback in ASP.NET only occurs within web forms (i.e., within a form element with runat=server), and only server controls post back information to the server.

Each cycle in which information is displayed then posted back to the server, and then redisplayed again, is sometimes also called a round trip.


Browser
Please enter your name: Randy
Choose favorite author: Melville
Enter
In Page_Load
7. Postback to server i.e., browser requests (POST) EventTest.aspx
8. Server processes EventTest.aspx
9. Calls Page_Load event handler
6. User fills in form and clicks Enter
2. Server processes EventTest.aspx
1 .Browser requests (GET) EventTest.aspx
3. Calls Page_Load event handler
5. Display in browser
4. Generates HTML response
Browser
Please enter your name:
Choose favorite author: Choose an author
Enter
In Page_Load
10. Calls btnEnter_Click event handler
12 Display in browser
Browser
11. Generates HTML response
Please enter your name: Randy
Choose favorite author: Melville
Enter
In Page_Load
Hi Randy
Your favorite author is Melville

Postback

Event Types in ASP.NET

There are two types of events in ASP .Net

Page events - Always triggered and always in a certain specific order (see Page Lifecycle)

Control events - Associated with particular controls and only triggered in certain circumstances.

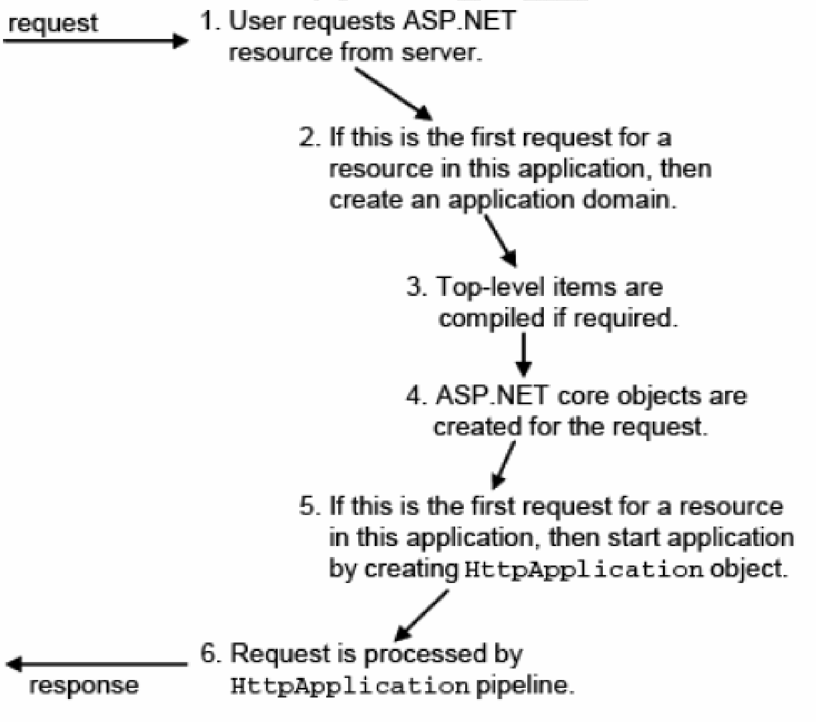
Page Life Cycle in ASP.NET

Page and control events occur in a certain order which we can call the page life cycle. Five general stages

- ✓ Page initialization
- ✓ Loading
- ✓ Postback control event handling
- ✓ Rendering
- ✓ Unloading

ASP.NET Application Lifecycle

The page life cycle is just one of several processing steps which occur as part of the larger ASP.NET life cycle.


request
1. User requests ASP.NET resource from server.
2. If this is the first request for a resource in this application, then create an application domain.
3. Top-level items are compiled if required.
4. ASP.NET core objects are created for the request.
5. If this is the first request for a resource in this application, then start application by creating HttpApplication object.
6. Request is processed by HttpApplication pipeline.
response

ASP.NET Application Lifecycle

1.4 JavaScript and Client-Side Code

Modern browsers understand an internal programming language called JavaScript. When the browser encounters JavaScript code inside an HTML page, it runs the program's instructions. The browser (the client) doesn't need a connection to the server to run JavaScript code —it's completely independent. Client-side script uses the processing power of the computer on which the browser is running. That's a tremendous advantage because it takes the pressure off the Web server and distributes tasks to individuals.

Client-side scripting becomes complicated — and extremely powerful —when combined with logic on the server. Imagine this scenario: The Web server sends a stream of HTML that contains JavaScript instructions. Those instructions include JavaScript code that checks whether the anonymous user has typed a number from 1 to 10 in a text box. The browser sees the script and executes it locally. Until the user has typed a number from 1 to 10, the Web server isn't involved. When the browser sends the number back to the Web server, the return action is known as a postback.

1.5 ASP.NET AJAX

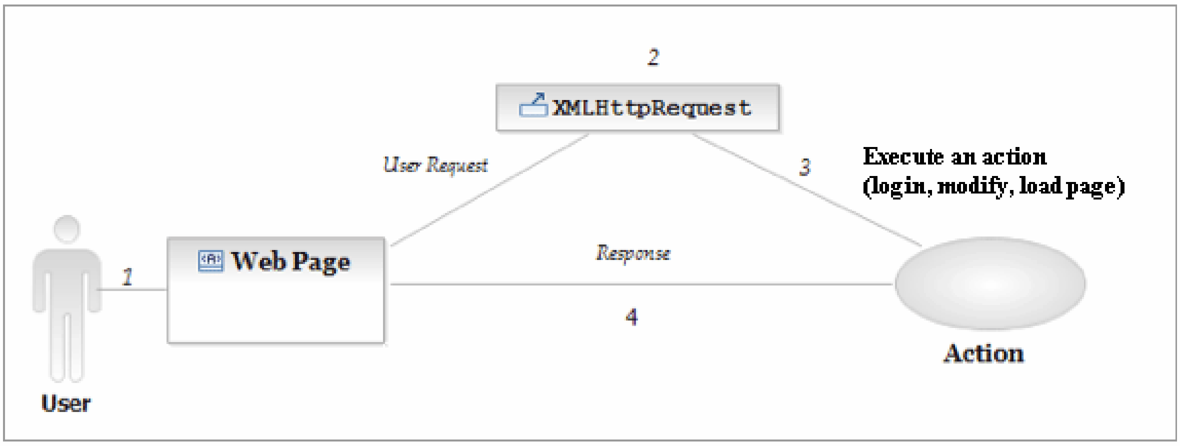
AJAX

In traditional JavaScript coding, if you want to get any information from a database or a file on the server, or send user information to a server, you will have to make an HTML form and GET or POST data to the server. The user will have to click the "Submit" button to send/get the information, wait for the server to respond, and then a new page will load with the results.

Because the server returns a new page each time the user submits input, traditional web applications can run slowly and tend to be less user-friendly. With AJAX, your JavaScript communicates directly with the server, through the JavaScript XMLHttpRequest object.

With an HTTP request, a web page can make a request to, and get a response from a web server, without reloading the page. The user will stay on the same page, and he or she will not notice that scripts request pages, or send data to a server in the background.

This picture is a simplified introduction about how Ajax works



AJAX

ASP .NET Asynchronous JavaScript and XML (AJAX) is a technology that reduces unnecessary and wasteful full page refreshes by limited the transfer of data to and from the Web server.

On an AJAX-enabled page, you can type your credit card number in a text box, click the Submit button, and get a response such as “Credit Card Accepted” without disrupting the images, menus, and text elsewhere on the page. The browser sends only the required data to the server. When the message comes back, AJAX uses JavaScript code and Dynamic HTML to write into the designated part of the page.

1.6 Dynamic HTML

While not exclusively a Microsoft technology, Dynamic HTML (DHTML) plays an important role in making Web pages responsive, interactive, and more like a regular Windows program. When the browser analyzes the HTML code for a page, it creates an in memory document. This document has a hierarchical structure where child elements nest inside their parent containers. For example, table rows are nested inside tables that are nested within the document’s body.

The word dynamic in DHTML refers to the ability to change the characteristics of an element by using JavaScript. You’ve seen this ability many times without necessarily paying attention. For example, you’re seeing DHTML at work when you hover the mouse over an image, and the image changes. Likewise, DHTML is at work when you click a plus sign to expand a paragraph of text. Chances are, JavaScript is instructing the text (or its container) to become visible — even though the original code sent from the server set the text as hidden. The ability of JavaScript and ASP.NET AJAX to manipulate and rewrite almost any part of a Web page (the text included) is what makes most dynamic effects possible.

1.7 Extensible Markup Language (XML)

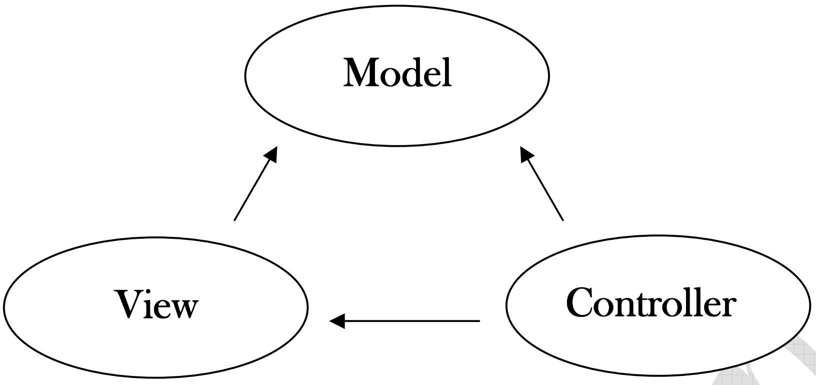
Although Microsoft had a hand in the specifications for Extensible Markup Language (XML), the standards come from the World Wide Web Consortium (W3C). Microsoft uses XML extensively in its Web technologies as a way of passing data around. These data exchanges include browser-to-server, server-to-browser, server-to-server, and from one program to another. XML is also a big part of AJAX. XML data has three big advantages

- ✓ It’s generated as plain text so that it passes easily through firewalls.
- ✓ Humans can read it and make at least some sense of it.
- ✓ You can create, parse, and manipulate XML on any platform, not just on Microsoft’s operating systems.

1.7 MVC

ASP.NET MVC is a part of the ASP.NET Web application framework. It is one of the two different programming models you can use to create ASP.NET Web applications, the other being ASP.NET Web Forms.

An MVC Application is designed and implemented using the following three attributes:



Model: The model contains the core information for an application. This includes the data and validation rules as well as data access and aggregation logic.

View: The view encapsulates the presentation of the application, and in ASP.NET this is typically the HTML markup.

Controller: The controller contains the control-flow logic. It interacts with the Model and Views to control the flow of information and execution of the application.

This separation of entity allows you to have quickness and flexibility in building and maintaining your application. For example, by separating the views, you can iterate on the appearance of your application without touching on any of the core business logic. You can also separate work by role. For example designers can work on the views, while developers work on the model.

ASP.NET MVC brings the power of this development paradigm to ASP.NET development, allowing you to use your .NET development skills to build MVC applications.

It gives you

- ✓ Complete control over your HTML Markup
- ✓ Enables rich AJAX and jQuery integration
- ✓ Allows you to create SEO-friendly URLs for your site
- ✓ Makes Test Driven Development (TDD) easy

1.8 ADO.NET

ADO.NET is Microsoft's technology for working with data and databases of all types. When a Web application talks to a database such as Microsoft SQL Server, it's probably using ADO.NET. The introduction of LINQ has hidden much of ADO.NET from view in Visual Web Developer.

1.9 SQL Server

SQL Server 2005 and 2008 are key products in Microsoft's Web technology strategy. The phrase "It's all about the data" applies to most serious Web applications. Whether you're tracking user preferences, generating complex reports, or storing customer orders, you need a fast and reliable data engine and relational database.

Microsoft provides SQL Server Express for free, making it a great choice for beginners. The skills and data you acquire by using SQL Express are directly transferable to the latest versions of SQL Server from standard to enterprise. You use SQL Server (mostly the Express version) throughout the book.

1.10 Internet Information Services

Internet Information Services (IIS) is Microsoft's premier Web server product that comes free with the latest versions of Windows. As a platform, IIS delivers Web pages and Web services as requested by a browser or other application.

You can run IIS on your developer workstation, over your company's intranet, or expose it to the vast public on the Internet. However, unless you're running a large business on the Internet, you probably use IIS through an independent hosting company. These hosters are specialists who rent space on their servers, sell bandwidth, maintain connections to the Internet, and schedule backups.

During the development stage in Visual Web Developer, you may not use IIS at all. VWD includes a light Web server that does almost everything you need on your local development machine. When you're satisfied with the pages and code, you transfer the site to an IIS machine from within VWD.

Certifications

- **Information Technology**
  - Certified BizTalk Professional
  - Certified Software Architects
  - Certified Open Source CMS (Drupal) Professional
  - Certified Software Testing Professional
- **Accounting, Banking and Finance**
  - Certified Business Accountant
  - Certified Foreign Exchange Professional
  - Certified Treasury Markets Professional
- **Office Skills**
  - Certified Office Administrator
- **Foreign Trade**
  - Certified Import Export Professional
- **Logistics and Supply Chain Management**
  - Certified Logistics and Supply Chain Management Professional

& many more job related certifications

Contact us at :
V-Skills
011-322 22 125 or **info@vskills.in**
Intelligent Communication Systems India Limited
DSIIDC Administrative Building,
Okhla Industrial Estate-III, Delhi-110020
www.vskills.in