

INEW 1000 Course Syllabus

Name of Course: Developing Microsoft ASP.NET Web Applications

2. Number of Clock Hours: 40

3. Course Description:

This course will teach Microsoft C# programmers and beginning Web developers the fundamentals of Web application site implementation by using Microsoft ASP.NET and Microsoft C#. This course focuses on using the Microsoft Visual Studio .NET environment and the Microsoft .NET platform to create an ASP.NET Web application that delivers dynamic content to a Web site. MOC 2310. Prerequisites: The ability to create HTML or DHTML and programming experience using C#

4. Course Objectives

At the end of the course, students will be able to:

- Explain the Microsoft .NET Framework and ASP.NET
- Create a component in C#.
- Create an ASP.NET Web application project by using Visual Studio .NET.
- Add server controls to an ASP.NET Web Form.
- Create and populate ASP.NET Web Forms.
- Add functionality to server controls that are on an ASP.NET Web Form.
- Use the Trace and Debug objects that are provided with Visual Studio .NET.
- Use validation controls to validate user input.
- Create a user control.
- Access data by using the built-in data access tools that are available in Visual Studio .NET.
- Use Microsoft ADO.NET to access data in an ASP.NET Web application.
- Accomplish complex data access tasks from an ASP.NET Web application.
- Access Extensible Markup Language (XML) data and read it into a DataSet.
- Call an XML Web service from an ASP.NET Web application and incorporate the returned data into a Web application.
- Store application and session data by using a variety of methods.
- Configure and deploy an ASP.NET Web application.
- Help protect an ASP.NET Web application by using a variety of technologies.

5. Rationale:

Upon completion of this course, students will have a better understanding of Microsoft C# programming, the most popular operating system in use today.

6. Required Materials:

Microsoft Official Curriculum, MOC 2310, included.

7. Evaluation

Those who participate in class discussions, complete course lab work, and miss no more than three class meetings will be awarded 4.0 continuing education units.

8. Course Outline

Module 1: Overview of the Microsoft .NET Framework

Take a closer look: [Download Sample Module 1](#) (Portable Document Format, 1.02 MB).

This module introduces the conceptual framework of the .NET Framework and ASP.NET.

Lessons

- Introduction to the .NET Framework
- Overview of ASP.NET
- Overview of the Lab Application
- Resources

There is no lab for this module

After completing this module, students will be able to:

- Explain the advantages of using the .NET Framework.
- Understand the key functionality and purpose of using ASP.NET when developing Web applications.
- Understand the basic functionality of the Web application that you will build in the labs throughout the course.

Module 2: Using Microsoft Visual Studio .NET

Take a closer look: [Download Sample Module 2](#) (Portable Document Format, 1.13 MB).

This module explains how to create new projects, and how to use the primary features that are available in Visual Studio .NET.

Lessons

- Overview of Visual Studio .NET
- Creating an ASP.NET Web Application Project

Lab 2: Using Microsoft Visual Studio .NET

- Creating an ASP.NET Web Application Project Using Visual Studio .NET

After completing this module, students will be able to:

- Navigate the Visual Studio .NET IDE.
- Create, build, and view an ASP.NET Web application.

Module 3: Using Microsoft .NET-Based Languages

This module will introduce the various languages that support .NET. This module will focus on Visual Basic .NET and C#. Students will use Visual Studio .NET to create a class project and write code in either Visual Basic .NET or C#.

Lessons

- Overview of the .NET-Based Languages
- Comparison of the .NET-Based Languages
- Creating a Component Using Visual Studio .NET

Lab 3: Building a Microsoft Visual Studio .NET Component

- Create a new project in Visual Studio .NET for a Visual Basic class

After completing this module, students will be able to:

- Identify the languages that support ASP.NET.
- Choose an appropriate development language for their needs.
- Create a component by using Visual Studio .NET.

Module 4: Creating a Microsoft ASP.NET Web Form

This module explains how to create and display an ASP.NET Web Form.

Lessons

- Creating Web Forms
- Using Server Controls

Lab 4: Creating a Microsoft ASP.NET Web Form

- Creating the default.aspx Web Form
- Creating the life.aspx Web Form

After completing this module, students will be able to:

- Add a Web Form to an ASP.NET Web Application project.
- Use the Visual Studio .NET toolbox to add server controls to a Web Form.

Module 5: Adding Code to a Microsoft ASP.NET Web Form

This module explains how to add event procedures to an ASP.NET Web application and add server controls on an ASP.NET Web Form. Examples will be show in Visual Studio .NET.

Lessons

- Using Code-Behind Pages
- Adding Event Procedures to Web Server Controls
- Using Page Events

Lab 5: Adding Functionality to a Web Application

- Creating a **Page_Load** Event Procedure
- Creating a **Click** Event Procedure

After completing this module, students will be able to:

- Use code-behind pages in an ASP.NET Web application.
- Create event procedures for Web server controls.
- Use **Page** events in an ASP.NET Web application.

Module 6: Tracing in Microsoft ASP.NET Web Applications

This module explains how to use the Trace feature and the Debug object in Visual Studio .NET. Students will learn about the two tracing techniques in ASP.NET: page-level tracing and application-level tracing. Students will also learn how use the debugger to create breakpoints, set watch variables, and step between pages and components in a Web application.

Lessons

- Understanding Tracing
- Remote Debugging

Lab 6: Tracing in Microsoft ASP.NET Web Applications

- Using Trace Statements
- Tracing into a Component

After completing this module, students will be able to:

- Use the **Trace** object to view runtime information about an ASP.NET Web application.
- Debug Web applications remotely.

Module 7: Validating User Input

This module explains how to use the client-side and server-side validation controls to screen data.

Lessons

- Overview of User Input Validation
- Using Validation Controls

- Page Validation

Lab 7: Validating User Input

- Using RequiredFieldValidator Controls
- Using the ValidationSummary Control
- Using the CompareValidator Control
- Using the RegularExpressionValidator Control

After completing this module, students will be able to:

- Identify when input validation is appropriate in Web Forms.
- Use input validation controls to verify user input on a Web Form.
- Verify that all validation controls on a page are valid.

Module 8: Creating User Controls

This module explains user controls and how to create them.

Lessons

- Adding User Controls to an ASP.NET Web Form
- Creating User Controls

Lab 8: Creating User Controls

- Creating a User Control
- Using the User Control

After completing this module, students will be able to:

- Add a user control to an ASP.NET Web Form.
- Create a user control.

Module 9: Accessing Relational Data Using Microsoft Visual Studio .NET

This module explains a conceptual overview of the objects in ADO.NET.

Lessons

- Overview of ADO.NET
- Creating a Connection to the Database
- Displaying a DataSet in a List-Bound Control

Lab 9: Accessing Data Using Microsoft Visual Studio .NET

- Connecting to a Database
- Paging and Selection in a DataGrid Control

After completing this module, students will be able to:

- Describe ADO.NET.
- Create a connection to a database by using ADO.NET.
- Display data in a Web Form by using a list-bound control.

Module 10: Accessing Data with Microsoft ADO.NET

This module explains how to manually add data access tools to a Web application.

Lessons

- Introduction to Using ADO.NET
- Connecting to a Database
- Accessing Data with DataSets
- Using Multiple Tables
- Accessing Data with DataReaders

Lab 10: Accessing Data with Microsoft ADO.NET

- Using a SqlDataReader
- Viewing Data from the Database

After completing this module, students will be able to:

Describe the ADO.NET object model that is used for accessing data.

- Create security-enhanced connections to a Microsoft SQL Server database by using the **SqlConnection** and **SqlDataAdapter** objects.
- Use **DataSet** objects to support the local data storage and manipulation requirements of Web Forms.
- Store multiple tables of data in a **DataSet** object, and then display that data in **DataGrid** controls.
- Programmatically read data from a SQL Server database by using a **SqlDataReader** object.

Module 11: Calling Stored Procedures with Microsoft ADO.NET

This module covers the more advanced and complicated features of ADO.NET.

Lessons

- Overview of Stored Procedures
- Calling Stored Procedures

Lab 11: Calling Stored Procedures with Microsoft ADO.NET

- Calling Stored Procedures with Microsoft ADO.NET

After completing this module, students will be able to:

- Explain what a stored procedure is and the reasons for using stored procedures when accessing a database.
- Call stored procedures.

Module 12: Reading and Writing XML Data

This module explains the methods that can be used for reading data from XML files.

Lessons

- Overview of XML Architecture in ASP.NET
- XML and the DataSet Object
- Working with XML Data
- Using the XML Web Server Control

Lab 12: Reading XML Data

- Reading a List from an XML File
- Reading, Transforming, and Displaying XML
- Nested Data

After completing this module, students will be able to:

- Describe XML architecture in ASP.NET.
- Read and write XML data into a **DataSet** object.
- Identify how to store, retrieve, and transform XML data by using **XMLDataDocument** and **XsltTransform** objects.
- Use the XML Web server control to display, load, and save XML data.

Module 13: Consuming and Creating XML Web Services

This module explains the steps that are necessary to access a Web service from an ASP.NET page and then incorporate that data into the Web application.

Lessons

- Overview of Using XML Web Services
- Calling an XML Web Service by HTTP
- Using a Proxy to Call an XML Web Service
- Creating an XML Web Service

Lab 13: Creating a XML Web Service

- Create an XML Web service
- Create an XML Web service method
- Consume an XML Web service method

After completing this module, students will be able to:

- Describe the purpose and process behind calling an XML Web service from a Web Form.
- Call an XML Web service directly from a browser by using Hypertext Transfer Protocol (HTTP).
- Create a Web reference proxy for an XML Web service Web method and call the method from a Web Form.
- Use the templates in Visual Studio .NET to create an XML Web service.

Module 14: Managing State

This module explains the several methods that are available for storing application and session data, for both short- and long-term storage.

Lessons

- State management
- Application and Session Variables
- Cookies and Cookieless Sessions

Lab 14: Storing Application and Session Data

- Using Session Variables
- Using Cookies
- Using Application Variables
- Storing Session Variables in a Database

After completing this module, students will be able to:

- Describe state management and its different types of options that are available to manage state in an ASP.NET Web application.
- Use application and session variables to manage state in ASP.NET Web applications.
- Use cookie and cookieless sessions to manage state in ASP.NET Web applications

Module 15: Configuring, Optimizing, and Deploying a Microsoft ASP.NET Web Application

This module explains how to configure and deploy an ASP.NET Web application.

Lessons

- Using the Cache Object
- Using ASP.NET Output Caching
- Configuring an ASP.NET Web Application
- Deploying an ASP.NET Web Application

Lab 15: Configuring, Optimizing, and Deploying a Microsoft ASP.NET Application

- Using the Cache object
- Using the Page Output Cache
- Partial Page Caching
- Using Dynamic Properties
- Deploying Your Site

After completing this module, students will be able to:

- Use the Cache object to store information.
- Use ASP.NET output caching to store Web pages and Web page fragments.
- Configure an ASP.NET Web application by using the Machine.config and Web.config files.
- Deploy an ASP.NET Web application.

Module 16: Helping to Protect a Microsoft ASP.NET Web Application

This module explains how to help protect a Web application by using a variety of technologies.

Lessons

- Web Application Security Overview
- Working with Windows-Based Authentication
- Working with Forms-Based Authentication
- Overview of Microsoft Passport Authentication

Lab 16: Securing a Microsoft ASP.NET Web Application

- Securing Your Web Site Using Windows-Based Authentication
- Securing Your Web Site Using Forms-Based Authentication
- Registering New Users
- Permitting Users to Sign Out

After completing this module, students will be able to:

- Describe the ASP.NET and Internet Information Services (IIS) authentication methods.
- Use Microsoft Windows-based authentication to help protect ASP.NET Web applications.
- Use Forms-based authentication to help protect ASP.NET Web applications.
- Use Microsoft Passport to help protect ASP.NET Web applications.

Module 17: Review

This module reinforces the concepts that the students have learned throughout the course. Students will have an opportunity to implement knowledge gained by using an interactive game.

Lessons

- Review of Material Covered
- Introduction to the Game

Lab 17: Review Game

- Part 1
- Part 2
- Part 3

After completing this module, students will be able to:

- Use their new knowledge to complete the tasks that are presented in the interactive game.