

Course 20486B:

Developing ASP.NET MVC 4 Web Applications

Course Details

Course Outline

Module 1: Exploring ASP.NET MVC4

The goal of this module is to outline to the students the components of the Microsoft Web Technologies stack, which can be used to host a completed web application. Students will also learn about ASP.NET 4.5 and be introduced to the web forms, web pages, and MVC programming models. Finally they will see an overview of ASP.NET MVC 4, including new features and configuration.

Lessons

- Overview of Microsoft Web Technologies
- Overview of ASP.NET 4.5
- Introduction to ASP.NET MVC 4

Lab : Exploring ASP.NET MVC4

- Exploring a Photo Sharing Application
- Exploring a Web Pages Application
- Exploring a Web Forms Application
- Exploring an MVC Application

After completing this module, students will be able to:

- describe the Microsoft Web Technologies stack and select an appropriate technology to use to develop any given application.

Module 2: Designing ASP.NET MVC 4 Web Applications

The goal of this module is to introduce students to the typical design process that architects must complete when they plan an MVC 4 application. At this stage in the design process, MVC 4 has been selected as the most appropriate programming model, but the details of the application, such as the overall architecture, Controllers, Views, Models, and routes to create, have not been fixed. How to plan such details is shown during this module.

Lessons

- Planning in the Project Design Phase

- Designing Models, Controllers, and Views

Lab : Designing ASP.NET MVC 4 Web Applications

- Planning Models
- Planning Controllers
- Planning Views
- Architecting an MVC Application

After completing this module, students will be able to:

- design the architecture and implementation of a web application that will meet a set of functional requirements, user interface requirements, and address business models.

Module 3: Developing ASP.NET MVC 4 Models

The goal of this module is to enable the students to create Models within an MVC application that implement the business logic necessary to satisfy business requirements. The module also describes how to implement a connection to a database, or alternative data store, using the Entity Framework and LINQ.

Lessons

- Creating MVC Models
- Working with Data

Lab : Developing ASP.NET MVC 4 Models

- Creating an MVC Project and Adding a Model
- Creating a New SQL Azure Database in Visual Studio
- Adding Properties and Methods to MVC Models
- Using Display and Edit Annotations in MVC Models

After completing this module, students will be able to:

- create MVC Models and write code that implements business logic within Model methods, properties, and events.

Module 4: Developing ASP.NET MVC 4 Controllers

The goal of this module is to enable students to add Controllers to MVC applications and to implement actions that respond to user input and other events. The students will learn how Controllers relate to Models and how to implement Controller actions that define the View used to display or edit data. This module also covers how to write action filters that run code before or after multiple actions in the Controller. The students will learn about situations when action filters are useful.

Lessons

- Writing Controllers and Actions
- Writing Action Filters

Lab : Developing ASP.NET MVC 4 Controllers

- Adding an MVC Controller and Writing the Actions
- Writing the Action Filters in a Controller
- Using the Photo Controller

After completing this module, students will be able to:

- add Controllers to an MVC Application to manage user interaction, update models, and select and return Views.

Module 5: Developing ASP.NET MVC 4 Views

The goal of this module is to describe the role of Views in an MVC web application and enable users to create and code them. The syntax of a Razor View is of critical importance for students to understand because it defines both the layout and the functionality of the data display. HTML Helpers will also be discussed in detail and common Helpers, such as `Html.ActionLink()` and `Html.EditorFor()`, will be described. Reusing code by defining Partial Views and Razor Helpers will be discussed as well.

Lessons

- Creating Views with Razor Syntax
- Using HTML Helpers
- Reusing Code in Views

Lab : Developing ASP.NET MVC 4 Views

- Adding a View for Photo Display
- Adding a View for New Photos
- Creating and Using a Partial View
- Adding a Home View and Testing the Views

After completing this module, students will be able to:

- create Views in an MVC application that display and edit data and interact with Models and Controllers.

Module 6: Testing and Debugging ASP.NET MVC 4 Web Applications

The goal of this module is to enable students to increase the resilience and quality of an application by locating and correcting code errors, bugs, and other unexpected results. MVC applications are well suited to unit testing techniques and these techniques ensure a high quality of code by systematically testing the functionality of each small component. In addition the debugging

tools and exception handling available in Visual Studio will be explained.

Lessons

- Unit Testing MVC Components
- Implementing an Exception Handling Strategy

Lab : Testing and Debugging ASP.NET MVC 4 Web Applications

- Performing Unit Tests
- Configuring Exception Handling

After completing this module, students will be able to:

- run unit tests and debugging tools against a web application in Visual Studio 2012 and configure an application for troubleshooting.

Module 7: Structuring ASP.NET MVC 4 Web Applications

The goal of this module is to enable students to structure a web application in such a way that users can rapidly locate the information they need. Two aspects of the design are emphasized: the URLs presented in the browser address bar should be understandable and can be controlled by adding routes to the ASP.NET Routing Engine, and the navigation controls, such as menus and breadcrumb trails, should present the most relevant links to frequently read pages. Search Engine Optimization is important throughout this module.

Lessons

- Analyzing Information Architecture
- Configuring Routes
- Creating a Navigation Structure

Lab : Structuring ASP.NET MVC 4 Web Applications

- Using the Routing Engine
- Building Navigation Controls

After completing this module, students will be able to:

- develop a web application that uses the ASP.NET routing engine to present friendly URLs and a logical navigation hierarchy to users.

Module 8: Applying Styles to ASP.NET MVC 4 Web Applications

The goal of this module is to explore how students can impose a consistent look and feel to an MVC application and share other common components, such as headers and footers, between all Views. Besides describing CSS styles and template views, the module will discuss how to migrate a look and feel created by a web designer into an MVC application. Techniques for adapting the display of a site for small screens and mobile devices will also be introduced.

Lessons

- Using Template Views
- Applying CSS to an MVC Application
- Creating an Adaptive User Interface

Lab : Applying Styles to ASP.NET MVC 4 Web Applications

- Using Template Views
- Applying a Consistent Look and Feel to an MVC Application
- Adapting Web Pages for Different Browsers

After completing this module, students will be able to:

- implement a consistent look and feel, including corporate branding, across an entire MVC web application.

Module 9: Building Responsive Pages in ASP.NET MVC 4 Web Applications

The goal of this module is to describe to the students how partial page updates and caching can optimize the responsiveness of a web application. Students will see how to make use of AJAX helpers and partial views to update small portions of a page instead of refreshing the entire page. The module also covers the different caches developers can use to store rendered pages and discusses how to configure caching for maximum performance.

Lessons

- Using AJAX and Partial Page Updates
- Implementing a Caching Strategy

Lab : Building Responsive Pages in ASP.NET MVC 4 Web Applications

- Using Partial Page Updates
- Configuring the ASP.NET Caches

After completing this module, students will be able to:

- use partial page updates and caching to reduce the network bandwidth used by an application and accelerate responses to user requests.

Module 10: Using JavaScript and jQuery for Responsive MVC 4 Web Applications

The goal of this module is to teach the students techniques that run code on the browser. This approach can increase the responsiveness of the application because a rendered page can respond to a user action without reloading the entire page from the server. Students will learn about the jQuery script library and how to use it to call web services and update user interface components.

Lessons

- Rendering and Executing JavaScript Code

- Using jQuery and jQueryUI

Lab : Using JavaScript and jQuery for Responsive MVC 4 Web Applications

- Using jQuery to Respond to Users
- Using jQueryUI to Build a User Interface

After completing this module, students will be able to:

- write JavaScript code that runs on the client-side and utilizes the jQuery script library to optimize the responsiveness of an MVC web application.

Module 11: Controlling Access to ASP.NET MVC 4 Web Applications

The goal of this module to ensure good security in terms of strong authentication and authorization for access. The lessons describe how to enable anonymous users to create their own user account and gain privileged access to content.

Lessons

- Implementing Authentication and Authorization
- Assigning Roles and Membership

Lab : Controlling Access to ASP.NET MVC 4 Web Applications

- Configuring Authentication and Membership Providers
- Building the Logon and Register Views
- Authorizing Access to Resources
- Building a Password Reset View

After completing this module, students will be able to:

- implement a complete membership system in an MVC 4 web application.

Module 12: Building a Resilient ASP.NET MVC 4 Web Application

The goal of this module is to enable the students to build applications that are stable and reliable. Such applications are not vulnerable to common hacking techniques such as cross-site scripting and also store state information such as the contents of a shopping cart and user preferences. This state information is preserved when servers or browsers restart, connections are lost, and other connectivity issues occur.

Lessons

- Developing Secure Sites
- State Management

Lab : Building a Resilient ASP.NET MVC 4 Web Application

- Storing User Preferences

- Using User Preferences in the Photo Gallery

After completing this module, students will be able to:

- build an MVC application that resists malicious attacks and persists information about users and preferences.

Module 13: Using Windows Azure Web Services in ASP.NET MVC 4 Web Applications

The goal of this module is to introduce Windows Azure to the students and explain why a developer would write a Windows Azure service instead of code in a web application. Students will also see how to write such a service and call it from a web application or from other applications, such as a mobile device app.

Lessons

- Introducing Windows Azure
- Designing and Writing Windows Azure Services
- Consuming Windows Azure Services in a Web Application

Lab : Using Windows Azure Web Services in ASP.NET MVC 4 Web Applications

- Creating and Coding a Windows Azure Service
- Consuming Data from a Windows Azure Service

After completing this module, students will be able to:

- describe how to write a Windows Azure web service and call it from an MVC application.

Module 14: Implementing Web APIs in ASP.NET MVC 4 Web Applications

The goal of the module is to introduce the concept of a Web API to students and to describe how to make an application's core functionality more broadly available for integration into other web and mobile applications. Students will learn about the new Web API feature of MVC 4 and see how to build a RESTful Web API and call it from other applications.

Lessons

- Developing a Web API
- Calling a Web API from Mobile and Web Applications

Lab : Implementing Web APIs in ASP.NET MVC 4 Web Applications

- Developing a Web API in MVC 4
- Adding Routes and Controllers to Handle REST Requests
- Call RESTful services from Client-Side Code

After completing this module, students will be able to:

- describe what a Web API is and why developers might add a Web API to an application.

Module 15: Handling Requests in ASP.NET MVC 4 Web Applications

The goal of this module is to describe how to write components that intercept requests from browsers before they are received by MVC Controllers. These components include HTTP Modules, HTTP Handlers, and the Web Sockets protocol. The module describes scenarios in which developers use such components and shows how to add them to an MVC application.

Lessons

- Using HTTP Modules and HTTP Handlers
- Using Web Sockets

Lab : Handling Requests in ASP.NET MVC 4 Web Applications

- Writing a Web Handler that Uses Web Sockets
- Building a Chat Room in The Photo Sharing Application

After completing this module, students will be able to:

- modify the way browser requests are handled by an MVC application.

Module 16: Deploying ASP.NET MVC 4 Web Applications

The goal for this module is to enable students to deploy a completed MVC application to a web server or Windows Azure. The module begins by describing testing, staging, and production deployments and the web server environments required for each. It also describes the advantages and disadvantages of using Windows Azure to host the application. Students also see all the available deployment options in Visual Studio.

Lessons

- Deploying a Web Application
- Deploying an MVC 4 Application

Lab : Deploying ASP.NET MVC 4 Web Applications

- Deploying an Application to Windows Azure
- Test the Completed Application

After completing this module, students will be able to:

- describe how to package and deploy an ASP.NET MVC 4 web application from a development computer to a web server for staging or production.
-