

10 Ways to Build Web Services in .NET ASP.NET MVC4

Chad McCallum
@ChadEmm



Module Outline

- **Introduction to ASP.NET MVC4**
- **Review ASP.NET MVC4 Configuration**
- **Create a ApiController and method**
- **Use an Accept header to request a different data format**
- **Create a Get method with a route parameter**
- **Create a Post and Put method with a request body parameter**
- **Create a Delete method**
- **Automatically generate API Documentation**
- **Use the EasyHTTP library to request data from the web service**
- **Use the EasyHTTP library to send data to the web service**
- **Review**

ASP.NET MVC4

- **Model – View – Controller**

- Model: Data and its operations
- Controller: Handles the interaction between the client and the server
- View: A representation of data (HTML, JSON, XML, etc)

- **Route**

- Maps between request and appropriate controller and action
- Uses Uri Templates
 - `/ {controller} / {action} / {param}`
 - `/api/ {controller}`

ASP.NET MVC4

■ **Convention over Configuration**

- Relies on naming conventions and other standards to avoid configuration steps
- Controllers: /Controllers folder, extends Controller base class
- Views: /{Controller}/{View}.cshtml
- Routes: /{Controller}/{Method}
 - In the case of ApiController, the HTTP verb corresponds to the method executed

Review

- **Introduction to ASP.NET MVC4**
- **Review ASP.NET MVC4 Configuration**
- **Create a ApiController and method**
- **Use an Accept header to request a different data format**
- **Create a Get method with a route parameter**
- **Create a Post and Put method with a request body parameter**
- **Create a Delete method**
- **Automatically generate API Documentation**
- **Use the EasyHTTP library to request data from the web service**
- **Use the EasyHTTP library to send data to the web service**

ASP.NET MVC4 – Convention over Configuration

- **MVC4 relies on convention to replace configuration files**
 - Uses reflection to automatically include code that extends certain interfaces or base classes (ApiController, IValueController, etc.)
 - Routes define how URIs and HTTP verbs are mapped to ApiController methods
 - Views require a certain folder and naming structure to match Controllers

ASP.NET MVC4 – Value Providers and Content Negotiation

- **MVC4 comes with a few baked-in value providers**
 - Route, Form (request body), File, Json, Query String
- **Value Providers automatically deserialize data from a request and pass it into a Controller's method if present**
 - For Get(int id), the ID could come from a route parameter, the Query String, JSON, or Form Data
- **MVC4 provides automatic content negotiation**
 - Uses the Accept or Content-Type header to format data if the requested type is supported

ASP.NET MVC4 - EasyHTTP

- **The EasyHTTP library is an open source library that provides an easy interface to create HTTP requests**
 - Create and send requests using the Get, Post, Put, and Delete methods
 - Data is automatically serialized and deserialized by the JsonFX library (installed with NuGet package)
 - Available under a Modified BSD license