Summary of ASP.NET Web API

Overview

- ASP.NET Web API - Easy way to create web services responding to HTTP verbs - GET, POST, PUT & DELETE.
- Using Web API, Build back-end web services that a client-specific web application can call.

SOAP-based Web Services vs HTTP Web Services

- SOAP Stands for " Simple Object Access Protocol "
- Specification for exchanging structured data i.e. XML using web services built on top of the HTTP Protocol
- Relies on HTTP, follows the request and response model represented by XML documents called messages.
- HTTP is a flexible protocol that works very well on the internet where most we communicate.
- HTTP messages can be cached and travel through firewalls.
- They're lightweight which can be processed by processors on mobile devices.
- They can be encrypted and best of all nearly every programming environment in the world.
- SOAP-based web services required a tool kit and more processing power whereas HTTP is everywhere.

Introduction to WEB API

- Web API framework, is part of ASP.NET MVC which enables to build Representational State Transfer (REST) –enabled web services.
- REST enabled APIs help external systems use the business logic implemented in your application to increase the reusability of the application logic.
- Web API facilitates two way communication between client system and the server
  - Reading data values
  - Updating data values
- Web API enable to obtain business information without complicated XML request such as SOAP
- Web API uses URL in requests and obtains results in the JSON format

Features of Web API

- Strong Support for URL Routing to produce clean URLs similar to MVC
- Content Negotiation based on Accept headers for request and response serialization
- Support for a host of supported output formats including JSON, XML, ATOM
- Strong default support for REST semantics
- Very extensible, based on MVC like extensibility model of Formatters and Filters
- Self-hostable in non-Web applications
- Testable using testing concepts similar to MVC

HTTP Verbs

- GET - This is used to retrieve resource(s).
- POST - This is used to add new resource.
- PUT - This is used to update resource.
- DELETE - This is used to delete resource.

Consuming ASP.NET Web API

- Can be consumed from any .NET Application using System.Net.Http.HttpClient class
- Use XMLHttpRequest object to consume an Web API from Java script and JQuery

Web API Routing

- Routing helps to map HTTP request to the Web API controllers and actions by using HTTP verbs and the request URL
- By default , routing rule in Web API is similarly to the routing rule in ASP.NET MVC

- We can make use of the naming convention to map request to actions or we can control the behavior of mapping by using annotations on action methods.
- Web API 2 support two types of Routing - Convention based Routing & Attribute Routing

Action Filters

- ActionName - Represents an attribute that is used for the name of an action.
- NonAction - Represents an attribute that is used to indicate that a controller method is not an action.

Web API Parameter Binding

- Parameter Binding provides a mechanism to get values from the URI and from the message body
- It allows to bind values to parameters when a controller are called
- The rules for binding the parameters depend upon the following types
  - Simple Types eg:- int ,bool, double,Datetime ,etc
  - Complex Type eg :- instance of a class
- Following attributes helps in Parameter Binding
  - FromUriAttribute - Specifies that an action parameter comes from URI of the incoming Http Request
  - FromBodyAttribute - Specifies that an action parameter comes only from the entity body of the incoming Http Request

Content Negotiation

- The process of selecting the best representation for a given response when there are multiple choices.
- The primary mechanism for content negotiation in HTTP are these request headers -
  - Accept, Accept-Charset, Accept-Encoding, Accept-Language, etc.

How Content Negotiation Works

- First, the pipeline gets the IContentNegotiator service from the HttpConfiguration object. It also gets the list of media formatters from the HttpConfiguration.Formatters collection.
- Next, the pipeline calls IContent Negotiatior .Negotiate, passing in: a) The type of object to serialize. b)The collection of media formatters. C) The HTTP request
- The Negotiate method returns two pieces of information: a) which formatter to use. b) The media type for the response.
- If no formatter is found, the Negotiate method returns null, and the client receives HTTP error 406 (Not Acceptable).

Securing Web API

- Security is a very important concern for every application.
- In case of a Web Application it is very important and crucial concern for developer.
- Implementing Security is a very complex concept.
- In Asp.Net Web API security can be implement using Authentication and Authorization
  - Authentication helps to Authenticate a user against the application
  - Authorization help to check whether Authenticated user has permission to perform specific task.
- Types of Authentication in Web API
  - Forms Authentication
  - Basic Authentication
- Authorize Attribute
  - Authorize attribute is one of the of authentication filter which help us to implement authentication and authorization in Asp.Net Web API.
  - It is available in System.Web.Http namespace
  - Using this attribute we set an action of controller to be called only when a user has authenticated himself or herself.
  - We can also include user from a role or specific users to access the action method.