

Giới thiệu ASP.NET Web API

Nguyễn Thanh Tùng
CTO - MISA

Nội dung

1. Giới thiệu ASP.NET Web API

1. Tính năng
2. Web API và WCF

2. Web API Controller

1. Routes
2. Demo

3. Web API Clients



Giới thiệu ASP.NET Web API

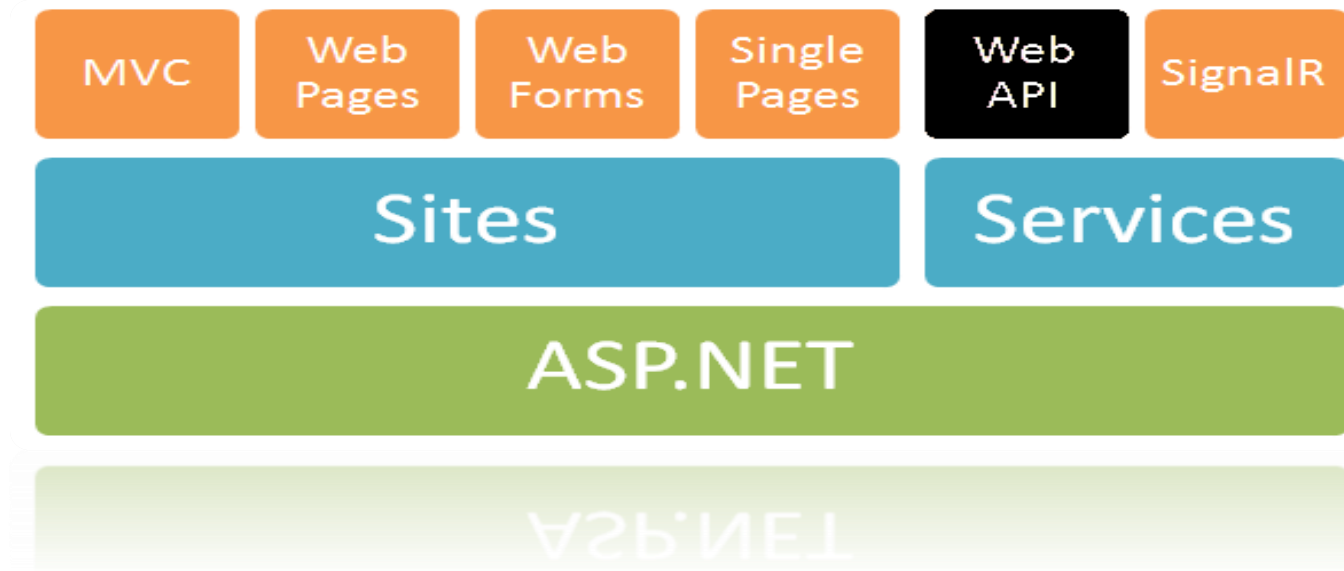


ASP.NET

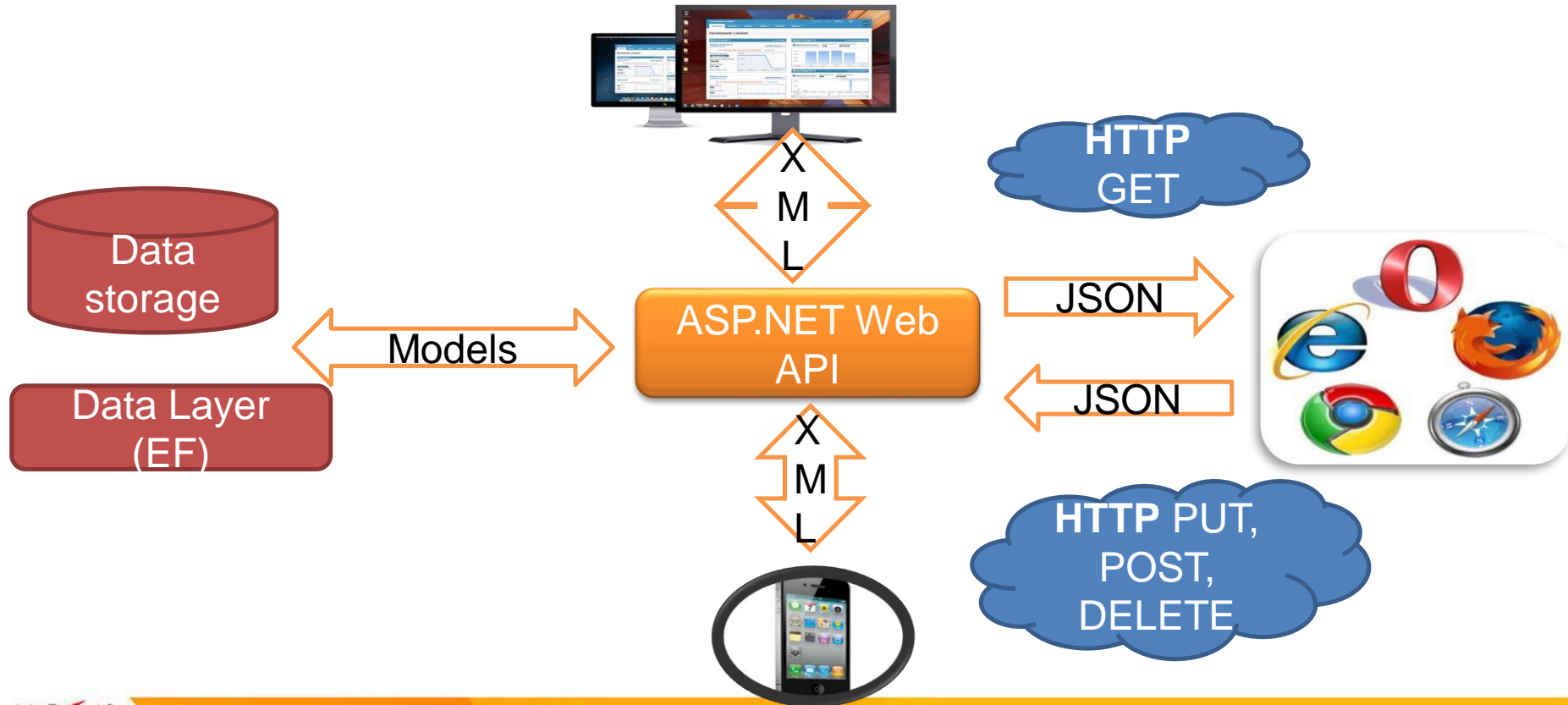
Web API

ASP.NET Web API

- Là một framework giúp xây dựng các HTTP service ([RESTful style](#)) cho các ứng dụng web, desktop, mobile một cách nhanh chóng, dễ dàng.



Vai trò của ASP.NET Web API



Các tính năng của Web API

ASP.NET Web API Features

From ASP.NET MVC

- ④ ASP.NET Routing
- ④ Model binding
- ④ Validation
- ④ Filters
- ④ Link generation
- ④ Testability
- ④ IoC integration
- ④ VS template
- ④ Scaffolding

From WCF Web API

- ④ Modern HTTP programming model
- ④ HttpClient
- ④ Task-based async
- ④ Formatting, content negotiation
- ④ Server-side query composition
- ④ Create custom help pages
- ④ Self-host
- ④ Tracing

So sánh Web API và WCF

WCF	ASP.NET Web API
<ul style="list-style-type: none">• Back-end Services• SOAP, WS-*• Transports: HTTP, TCP, UDP, Queues, WebSockets, custom• Message patterns: request-reply, one-way, duplex• Use WCF Web HTTP to add HTTP endpoints to existing WCF services• Use WCF Data Services for full OData support	<ul style="list-style-type: none">• Front-end Services• Media Types: JSON, XML, form-URL-encoded, custom• HTTP only• Request-reply only• REST, resource-centric• Use SignalR for asynchronous signaling (polling, long-polling, WebSockets)

Web API Controllers



Web API Controllers

- **Controller** là một đối tượng xử lý **HTTP** requests
- Tất cả API controllers đều kế thừa từ **ApiController**
- Mặc định **ASP.NET Web API** sẽ ánh xạ các **HTTP** requests tới những phương thức cụ thể - actions

Action	HTTP method	Relative URI	Method
Get a list of all posts	GET	/api/posts	Get()
Get a post by ID	GET	/api/posts/ <i>id</i>	Get(int id)
Create a new post	POST	/api/posts	Post(PostModel value)
Update a post	PUT	/api/posts/ <i>id</i>	Put(int id, PostModel value)
Delete a post	DELETE	/api/posts/ <i>id</i>	Delete(int id)
Get a post by category	GET	/api/posts?category= <i>category</i>	Get(string category)

Luồng xử lý mặc định của Web API

1

Web Request

2

Match a Route

3

API Controller
Responds

http://localhost:1337/api/posts

HTTP GET
Request

Tên Controller

```
public class PostsController : ApiController
{
    public string Get()
    {
        return "Some data";
    }
}
```

Kết quả trả về của Web API

- Mặc định trả về mã HTTP 200 khi mọi thứ OK
- Có lúc chúng ta cần trả về lỗi

```
public HttpResponseMessage Get(int id)
{
    if (dataExists)
    {
        return Request.CreateResponse(
            HttpStatusCode.OK, data);
    }
    else
    {
        return Request.CreateErrorResponse(
            HttpStatusCode.NotFound, "Item not found!");
    }
}
```

Web API Clients



Sử dụng HttpClient

- **HttpClient** là API mới nhất của **.NET** rất linh hoạt và dễ mở rộng để truy cập các HTTP service
- Có cùng mô hình lập trình như **ASP.NET Web API** ở phía server
 - **HttpRequestMessage** / **HttpResponseMessage**
- Có thể dùng Task pattern trên **.NET 4.0**
 - **async/await** trên **.NET 4.5**

VD HttpClient

```
var client = new HttpClient { BaseAddress = new
Uri("http://localhost:28670/") };
client.DefaultRequestHeaders.Accept.Add(new
MediaTypeWithQualityHeaderValue("application/json"));
HttpResponseMessage response = client.GetAsync("api/posts").Result;
if (response.IsSuccessStatusCode)
{
    var products = response.Content.ReadAsAsync<IEnumerable<Post>>().Result;
    foreach (var p in products)
    {
        Console.WriteLine("{0,4} {1,-20} {2}",
            p.Id, p.Title, p.CreatedOn);
    }
}
else
    Console.WriteLine("{0} ({1})",
        (int)response.StatusCode, response.ReasonPhrase);
```

Sử dụng Web API từ JS

- Web API có thể được sử dụng bởi JavaScript qua các lời gọi HTTP AJAX request

```
<ol id="posts"></ol>
<script>
  $.ajax({
    url: '/api/posts',
    success: function (posts) {
      var list = $('#posts');
      for (var i = 0; i < posts.length; i++) {
        var post = posts[i];
        list.append('<li>' + post.title + '</li>');
      }
    }
  });
</script>
```

Thanks for your listening

Q&A