**Bundling and Minification**

Bundling và minification là 2 kỹ thuật có thể sử dụng trong ASP.NET từ phiên bản 4.5 để cải thiện thời gian load request. Bundling và minification cải thiện thời gian load bằng cách giảm kích cỡ của requested assets(như CSS, JavaScript)

**REST và RESTFUL API**

**API** (**A**pplication **P**rogramming **I**nterface) là một tập các quy tắc và cơ chế mà theo đó, một ứng dụng hay một thành phần sẽ tương tác với một ứng dụng hay thành phần khác. API có thể trả về dữ liệu JSON, XML

REST (**RE**presentational **S**tate **T**ransfer) là phương thức tạo API với nguyên lý tổ chức nhất định. Nguyên lý này hướng dẫn lập trình viên tạo môi trường xử lý API request toàn diện.Là một kiểu kiến trúc để viết API.Là quy định cách sử dụng các HTTP method (GET,POST,PUT,DELETE,…) và cách định dạng các URL cho ứng dụng web để quản lý tài nguyên.

# Resful API là một tiêu chuẩn dùng trong việc thiết kế các API trong các ứng dụng web để quản lý tài nguyên. Là một trong những kiểu thiết kế API được sử dụng phổ biến trong các ứng dụng web, mobile,… để giao tiếp với nhau. ASP.NET Web API

Asp.Net Web Api là một framework cho phép xây dựng HTTP services mà có thể truy cập từ bất kỳ client nào bao gồm cả browser và thiết bị dị động. Nó là một platform phù hợp để xây dựng ứng dụng RESTful trên nền .NET framework.

Thứ tự chạy các file config là WebApiConfig, GlobalFilters, RouteConfig, BundleConfig

Có thể phát triển cả Resful và Non-Restful Web Service sử dụng ASP.NET Web Api framework, nhưng đa số được dùng để tạo RESTful service. Framework này không cung cấp bất kì loại kiến trúc nào để tạo services

Routing table trong Web API chứa route template được định nghĩa trongWebApiConfig.cs. Về WebApiConfig.cs định nghĩa “routeTemplate: "api/{controller}/{id}",”. Trong template này api là từ khóa bắt buộc(Từ khóa để xác định action trong WebApi controller hay MVC controller), {controller} sẽ được thay thế bằng giá trị tên controller, còn id là tham số tùy chọn có thể có hoặc không.

**Cách web api framework xử lý HTTP Request:**

Khi ASP.NET Web API Framework nhận HTTP request, nó cố gắng để match URI với một trong nhưng route templates khả thi trong routing table. Nếu không route template nào match được với URI, thì Web API Framework sẽ trả về 404 error đến client từ nơi bắt đầu của request.

-**Sự khác nhau giữa ASP.Net Web Api và ASP.Net Web MVC**

+Web MVC được sử sử dụng để tạo ứng dụng web trả về cả views và data còn Web API được sử dụng để tạo HTTP services toàn diện với cách dễ dàng và đơn giản chỉ trả về data và không có view.

+Web Api sử dụng HTTP method(định nghĩa trong webapiconfig.cs) để gọi đến action (MVC sử dụng URI định nghĩa trong file routeconfig.cs)

+Web Api giúp tạo REST-ful services trên nên .NET Framework và nó cũng hỗ trợ trao đổi nội dung(nó quyết định response format data tốt nhất mà có thể chấp nhận được bởi client. Nó có thể là JSON,XML, ATOM hoặc là một loại data khác),

+Web Api cũng quan tâm đến returning data trong một số loại định dạng như JSON,XML hoặc bất kì kiểu nào khác dự vào Accept header trong request. MVC chỉ return data ở kiểu JSON dựa vào JsonResult.

+Trong Web Api request được map với action dựa vào HTTP verbs( POST, GET, PUT, PATCH, and DELETE tương ứng với các hành động create, read, update và delete).

+ASP.Net Web api là một framework mới là một phần của ASP.NET framework core. Model binding, filter, routing và các tính năng của MVC khác ở trong Web APi khác với trong ASP.Net Web MVC và nó tồn tại trong System.Web.Http . Trong ASP.Net web MVC thì những tính năng trên tồn tại ở System.Web.Mvc.

+Có thể pha trộn giữa Web Api và MVC controller trong 1 project để xử lú AJAX request phức tạp mà kiểu dữ liệu trả về của nó là JSON, XML hoặc một format khác và xây dựng Http service toàn diện. Thông thường nó được gọi bởi Web Api self-hosting.

+Khi trộn giữa MVC và Web Api controller và muốn implement authorization thì ta có thể tạo 2 filters, một cho MVC và cái còn lại cho web api vì 2 cái khác nhau.

+Web Api có kiến trúc nhẹ và nó có thể sử dụng cho ứng dụng smartphone