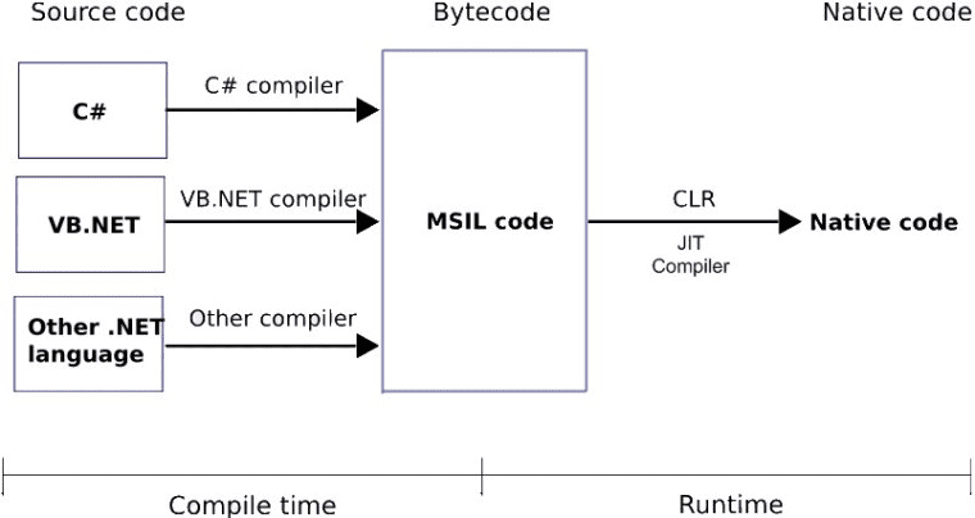
**NGÔN NGỮ - CÔNG NGHỆ SỬ DỤNG**

1. Ngôn ngữ
   1. Khái niệm

C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000. C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.

Trong các ứng dụng Windows truyền thống, mã nguồn chương trình được biên dịch trực tiếp thành mã thực thi của hệ điều hành.  
Trong các ứng dụng sử dụng .NET Framework, mã nguồn chương trình (C#, VB.NET) được biên dịch thành mã ngôn ngữ trung gian MSIL (Microsoft intermediate language).

Sau đó mã này được biên dịch bởi Common Language Runtime (CLR) để trở thành mã thực thi của hệ điều hành. Hình bên dưới thể hiện quá trình chuyển đổi MSIL code thành native code.



C# với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF (Windows Presentation Foundation), phát triển game, ứng dụng Web, ứng dụng Mobile trở nên rất dễ dàng.

* 1. Đặc trưng của ngôn ngữ lập trình C#
* C# là ngôn ngữ đơn giản

C# loại bỏ một vài sự phức tạp và rối rắm của những ngôn ngữ như Java và c++, bao gồm việc loại bỏ những macro, những template, đa kế thừa, và lớp cơ sở ảo (virtual base class).  
Ngôn ngữ C# đơn giản vì nó dựa trên nền tảng C và C++. Nếu chúng ta thân thiện với C và C++ hoặc thậm chí là Java, chúng ta sẽ thấy C# khá giống về diện mạo, cú pháp, biểu thức, toán tử và những chức năng khác được lấy trực tiếp từ ngôn ngữ C và C++, nhưng nó đã được cải tiến để làm cho ngôn ngữ đơn giản hơn.

* C# là ngôn ngữ hiện đại

Điều gì làm cho một ngôn ngữ hiện đại? Những đặc tính như là xử lý ngoại lệ, thu gom bộ nhớ tự động, những kiểu dữ liệu mở rộng, và bảo mật mã nguồn là những đặc tính được mong đợi trong một ngôn ngữ hiện đại. C# chứa tất cả những đặc tính trên. Nếu là người mới học lập trình có thể chúng ta sẽ cảm thấy những đặc tính trên phức tạp và khó hiểu. Tuy nhiên, cũng đừng lo lắng chúng ta sẽ dần dần được tìm hiểu những đặc tính qua các nội dung khoá học này.

* C# là ngôn ngữ thuần đối tượng

Lập trình hướng đối tượng (OOP: Object-oriented programming) là một phương pháp lập trình có 4 tính chất. Đó là tính trừu tượng (abstraction), tính đóng gói (encapsulation), tính đa hình (polymorphism) và tính kế thừa (inheritance). C# hỗ trợ cho chúng ta tất cả những đặc tính trên.

* C# là ngôn ngữ ít từ khóa

C# là ngôn ngữ sử dụng giới hạn những từ khóa. Phần lớn các từ khóa được sử dụng để mô tả thông tin. Chúng ta có thể nghĩ rằng một ngôn ngữ có nhiều từ khóa thì sẽ mạnh hơn. Điều này không phải sự thật, ít nhất là trong trường hợp ngôn ngữ C#, chúng ta có thể tìm thấy rằng ngôn ngữ này có thể được sử dụng để làm bất cứ nhiệm vụ nào.

* 1. **Nền tảng .NET**
* NET Framework được Microsoft đưa ra chính thức từ năm 2002. .NET Framework chỉ hoạt động trên Windows. Những nền tảng ứng dụng như WPF, Winforms, ASP.NET(1-4) hoạt động dựa trên .NET Framework.
* Mono là phiên bản cộng đồng nhằm mang .NET đến những nền tảng ngoài Windows. Mono được phát triển chủ yếu nhằm xây dựng những ứng dụng với giao diện người dùng và được sử dụng rất rộng rãi: Unity Game, Xamarin…
* Cho đến năm 2013, Microsoft định hướng đi đa nền tảng và phát triển .NET core. .NET core hiện được sử dụng trong các ứng dụng Universal Windows platform và ASP.NET Core. Từ đây, C# có thể được sử dụng để phát triển các loại ứng dụng đa nền tảng trên các hệ điều hành khác nhau (Windows, Linux, MacOS,…)

1. Công nghệ sử dụng
   1. Khái niệm

ASP.NET Core là gì? Nó là một open-source mới và framework đa nền tảng (cross-platform) cho việc xây dựng những ứng dụng hiện tại dựa trên kết nối đám mây, giống như web apps, IoT và backend cho mobile.

Ứng dụng ASP.NET Core có thể chạy trên .NET Core hoặc trên phiên bản đầy đủ của .NET Framework. Nó được thiết kế để cung cấp và tối ưu development framework cho những dụng cái mà được triển khai trên đám mây (clound) hoặc chạy on-promise.

Nó bao gồm các thành phần theo hướng module nhằm tối thiểu tài nguyên và chi phí phát triển, như vậy bạn giữ lại được sự mềm giẻo trong việc xây dựng giải pháp của bạn. Bạn có thể phát triển và chạy những ứng dụng ASP.NET Core đa nền tảng trên Windows, Mac và Linux.

Đồng thời nó đã trở thành một mã nguồn mở. Đây là một thay đổi rất lớn và theo mình là quan trọng nhất của ASP.NET Core. Điều mà trước đây khó có một lập trình viên nào có thể nghĩ đến. Có lẽ đó cũng là một xu thế mà các ngôn ngữ lập trình hiện nay đang hướng tới.

* 1. Đặc trưng của ASP.NET Core MVC
* PERFORMANCE

Hiệu năng (Performance) là một trong những đặc trưng quan trọng bậc nhất của bất kỳ ứng dụng nào. Ngay cả với ứng dụng web, Server có tài nguyên rất lớn nhưng các ứng dụng vẫn cần sử dụng lượng tài nguyên tối thiểu để thực hiện số lượng tối đa các tác vụ.

Application Performance không chỉ phụ thuộc vào Code như thế nào mà còn phụ thuộc vào kiến trúc của Framework trong nhân tố Performace. Với Asp.NET Core và Kestrel web server, khiến cho ứng dụng web hiệu năng hơn, tốn ít tài nguyên và chạy mượt hơn.

* CROSS-PLATFORM & CONTAINER SUPPORT

Sau khi [.Net Core](https://hachinet.vn/khoa-hoc/2/c-sharp-core)được công bố, người dùng Mac & Linux có thể phát triển các ứng dụng sử dụng Microsoft Technologies tuyệt vời trên các máy chạyMac & Linux.

Mặt khác, với Cloud. **DotNet Core** cho phép các nhà phát triển sử dụng Containers Technologies như Docker, Kuberenetes, …

* SINGLE PROGRAMMING MODEL FOR MVC AND WEB API

Trong MVC 5, controllers kế thừa từ lớp cơ sở **System.Web.Mvc.Controller** để phục vụ các HTTP requests trong khi Web API 2 controllers kế thừa từ **System.Web.Http.ApiController** để tạo các RESTful Web services sử dụng JSON & XML.

Với ASP.NET Core, MVC và Web API được merge vào cùng với nhau. Điều này được các nhà phát triển đánh giá cao vì nó làm cho việc phát triển trở nên đơn giản hơn.

* DEPENDENCY INJECTION

Built-in Dependency Injection là một trong những đặc trưng nổi bật của .**Net Core**. Nó làm cho các tác vụ trên các ngữ cảnh logging contexts, database contexts,… được truyền vào trong MVC controllers đơn giản hơn.

* SPA TEMPLATE

**Asp.Net Core** với các template **Angular** & **React** SPA Templates. Khi chúng ta tạo các dự án sử dụng các SPA template thì tất cả các package kèm theo Angular & React sẽ được tự động cài đặt và sử dụng NPM. Nó cũng hỗ trợ cho developer sử dụng Angular CLI dễ dàng hơn.

* WWWROOT FOLDER FOR STATIC FILES

Trong Asp.Net Framework [developers](https://hachinet.vn/khoa-hoc-developer)tạo các thư mục cho **Static Files (Images, CSS, Scripts)** với các tên thư mục tùy ý. Trong Asp.Net Core thư mục ngầm định là **“wwwroot”** cho tất cả các static files.