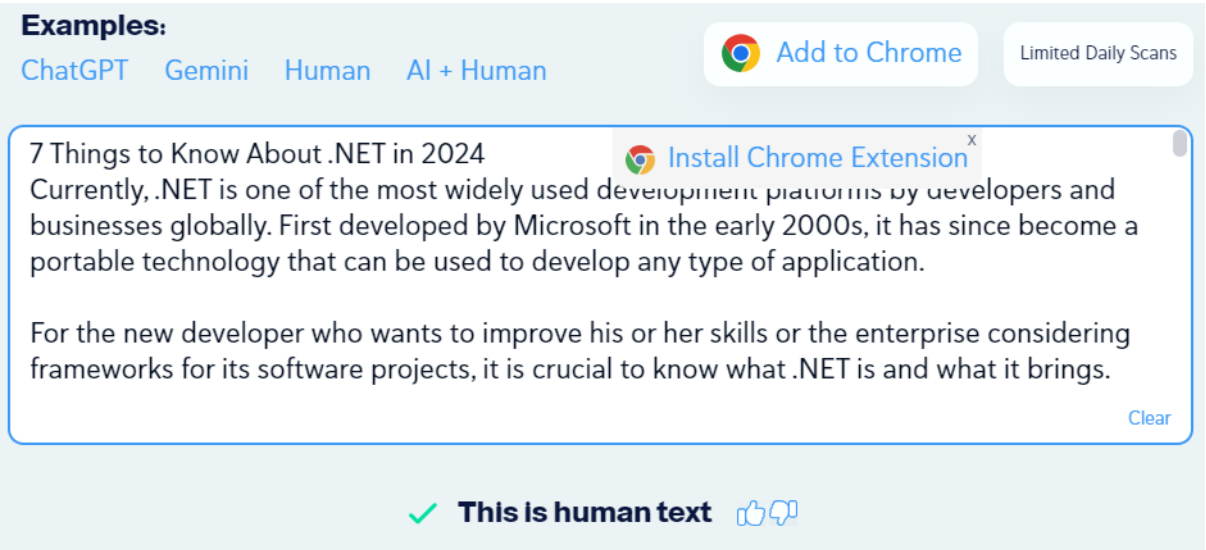
7 Things to Know About .NET in 2024

Currently, .NET is one of the most widely used development platforms by developers and businesses globally. First developed by Microsoft in the early 2000s, it has since become a portable technology that can be used to develop any type of application.

For the new developer who wants to improve his or her skills or the enterprise considering frameworks for its software projects, it is crucial to know what .NET is and what it brings.

In this article, we detail 7 things you should know about .NET in 2024 to be able to compare it with other platforms. But before we go any further, it would be useful to give a brief background of .NET and its evolution over the years.



7 Things to Know About .NET

In 2002, Microsoft released the first version of .NET as a commercial development framework that was tied to Windows. During that period, it assisted in centralizing the development of applications for fragmented Windows operating

systems. In the years that followed, open-source solutions started to appear on the market. In response, Microsoft ported .NET to other operating systems with .NET Core in 2016 and then merged the two lines with .NET 5.

The result today is a flexible, unified .NET 9 framework for building apps on Windows, Linux, macOS, mobile, web, cloud, and more. And there is still more innovation to come that will further improve .NET capabilities and platform support.

1. .NET is a Development Framework Created by Microsoft

.NET (pronounced “dot net”) is a free and open-source development framework. It provides the necessary tools and libraries to build many kinds of applications on Windows, from simple command-line apps to advanced web and mobile apps. Every dot net development outsourcing company and individual freelance developers actively use these features.

The key components of .NET include:

- **Common Language Runtime (CLR) –** Manages code execution and memory allocation
- **Base Class Library –** A large collection of reusable classes and types
- **Languages & Compilers –** Supports multiple programming languages like C#, Visual Basic, F#
- **IDEs like Visual Studio –** Integrated Development Environment for coding .NET apps

The biggest advantage of .NET is that it supports multiple languages, so developers are not limited to just one. It also makes building secure and robust applications easier through automatic memory management and inbuilt tools.

2. It Simplifies Cross-Platform Development

Initially, when.NET was released, it only targeted on development of Windows applications. That being said, in 2014, Microsoft released.NET Core that supports macOS, Linux, and Docker containers. This made .NET development possible for web servers, cloud infrastructure, and operating systems, except for Windows.

The latest .NET 9 framework builds on this with better cross-platform capability. Currently, the developers are able to create .NET applications on Windows, macOS, and a number of Linux distributions, including Red Hat, Ubuntu, SUSE, and others. This makes coding for various platforms easier to accomplish.

Microsoft also offers the tooling that is necessary to deploy to specific platforms. For instance, there is Visual Studio that is used to build applications specific to Windows, and there is Visual Studio Code for building applications for multiple platforms.

3. It Has Support for Microservices Architecture

Monolithic architecture is like building a single giant software application to handle all tasks. In contrast, microservices architecture breaks down app capabilities into smaller modular services. Each microservice focuses on a single capability and can deployed independently.

For example, an e-commerce app could have microservices for the product catalog, user authentication, payment processing, etc.

.NET makes developing microservices easier through ASP.NET Core. It provides inbuilt support for creating standalone,

containerized services that have their own code and data storage. These services can interact with each other via APIs.

Microservices built with .NET Core can be deployed on both Windows and Linux environments. This is perfect for public cloud platforms like Microsoft Azure, AWS, and Google Cloud that use Linux virtual machines. The modular architecture also allows individual services to be updated without affecting others.

4. C# is the Most Popular .NET Language

While .NET supports multiple languages, C# (pronounced "C sharp") is by far the most popular. It was created by Microsoft alongside .NET specifically for .NET development.

Here are some key reasons for C# popularity:

- **Easy to Learn:** C# syntax is simple and elegant, with similarities to Java and C++. This allows most developers to start building apps quickly.
- **Fully Object-oriented:** Everything in C#, including types and functions, exists within classes and objects. This fits perfectly with .NET's design.
- **Modern Features:** C# is constantly evolving with new features like async/await, LINQ, nullable reference types, etc. This prevents the language from feeling outdated.
- **Cross-Platform:** With .NET Core, C# can now be used to build apps for Windows, macOS, Linux, web, mobile, and more.
- **In Demand Skill:** A developer knowing C# along with .NET Framework/Core is highly sought after across industries.

C# skills almost go hand-in-hand with .NET skills. Learning C# to build .NET applications is a great career investment for developers.

5. It Integrates Well with Popular Frameworks

The .NET ecosystem features several popular web and mobile app frameworks that integrate seamlessly with .NET Core services. Some top examples are:

ASP.NET Core

An open-source framework for building web apps and APIs. It handles request routing, HTTP requests, security, etc., so developers can focus on app logic.

Xamarin

Enables building native iOS, Android & Windows apps with C# and sharing code across platforms. Xamarin apps have native performance since they don't use WebViews.

Blazor

Allows building interactive web UIs using C# instead of JavaScript. Blazor uses WebAssembly to run .NET code directly in browsers.

Unity

A top gaming engine providing 2D & 3D game development capabilities. Unity games can integrate advanced graphics while using C# for game logic.

So whether you want to build serverless web apps, mobile games or anything in between – there is likely a .NET integrated framework for it.

6. .NET Skills are Highly Valued in Jobs

.NET has been around for over 20 years and powers many enterprise systems and websites. Companies adopt it mainly because .NET applications are stable, secure, and scalable.

This high adoption leads to great demand for .NET developers across roles like:

- **Web Developer –** Build websites, web apps and APIs using ASP.NET Core
- **Mobile Developer –** Build cross-platform iOS, Android and Windows mobile apps
- **Backend Developer –** Work on .NET Core services and microservices
- **Game Developer –** Make games for PC, consoles and mobile using Unity
- **Cloud Developer –** Manage infrastructure and deploy .NET apps on Cloud platforms like Azure
- **AI/ML Developer –** Use .NET to integrate apps with Cognitive Services like machine learning

Websites like LinkedIn, StackOverflow and Hired.com consistently rate .NET among the top few in-demand tech skills. Learning .NET can open up abundant job opportunities.

7. Microsoft Offers Free Learning Resources

Getting started with .NET development is easy, thanks to the wealth of free resources provided by Microsoft –

Documentation – Comprehensive docs with tutorials, guides, API references, and more on all things .NET.

Microsoft Learn – Interactive modules covering ASP.NET, C#, and other skills from basics to advanced.

Visual Studio Community – Free version of Visual Studio IDE for students and open-source development.

.NET Sandbox – Browser-based sandbox environment to build apps with C#/.NET without any setup.

YouTube Channel – Video tutorials on .NET topics are updated weekly.

Virtual Training – Free virtual .NET training workshops hosted by .NET team members.

So whether you prefer reading docs, watching videos or hands-on learning – Microsoft has enough high-quality, free resources to master .NET!

Summary

Here are the key points we covered about Microsoft's .NET platform:

- It provides the framework, tools and languages for building apps on Windows and now for cross-platform, too.
- .NET Core simplified cloud and Linux development while still supporting Windows.
- Microservices architecture is made easier using ASP.NET Core services.
- C# language is immensely popular among .NET developers due to its ease of use and cross-platform support.
- .NET easily integrates with frameworks like ASP.NET, Xamarin, Unity etc.
- .NET skills are highly in demand across various developer roles.
- Tons of free resources are available for learning .NET and C#.

For developers looking for a robust, flexible and constantly evolving technology stack to build their skills on – .NET is definitely a top choice in 2024 and beyond!