

ECOSSISTEMA .NET

Palavra Chave: .NET

Tempo estimado: 15 minutos.

Objetivos do vídeo: Ser capaz de explicar a trajetória do .NET

Título da aula: Ecosistema .NET



Sejam todos bem vindos a mais um episódio, aqui na sociedade do código.
Como vocês estão? Tudo bem?

Na aula de hoje vamos falar sobre o ecossistema .NET mais afundo.

Para entendermos melhor o .NET, preciso que vocês tenham noção do que é um framework, e para isso vamos estudar sobre esse termo também.

Um framework é uma série de ações e estratégias que visam solucionar um problema bem específico.

Isso no contexto geral de framework, não apenas focando em tecnologia.

Então, dentro desse conceito, podemos entender que o framework será algo pré preparado, ou seja, algo já pronto, para nos auxiliar no desenvolvimento de um software.

Pense da seguinte forma:

Um programador provavelmente já criou algo que você está pensando em fazer.

Um aplicativo mobile, por exemplo.

E se alguém já teve a preocupação de escrever as linhas de código base, para a criação de um aplicativo mobile, por que você não poderia utilizar essa base, as estratégias criadas, para avançar em seu objetivo mais rapidamente?

Isso é exatamente o que um framework fará.

Ele cria uma abstração das funções já construídas anteriormente, para que você apenas as utilize, com o objetivo de ganhar praticidade e desempenho.

Ou seja, você não precisa mais inventar a roda do zero. Ela já está pronta.

Agora que você já tem noção do que é um framework, podemos avançar para o .NET

Primeiramente, eu quero avisá-los que deixei o pdf com o timeline dos lançamentos do .NET que utilizaremos para essa aula nos detalhes do vídeo, mas antes de você olhar esse documento, peço que preste muita atenção nessa aula porque é aqui que todo mundo se confunde.

Você vai ver uma dúzia de nomes bem semelhantes e isso pode confundir sua cabeça. Então larga tudo que você está fazendo e presta atenção.

A Microsoft iniciou seus trabalhos com o .NET em meados de 2002, lançando a primeira versão do .NET Framework. O .Net Framework 1.0.

Desde então, sucessivas versões do .NET Framework foram lançadas, e no finalzinho de 2016, a Microsoft deu um grande passo e apresentou sua nova ferramenta: o .NET Core.

O .NET Core é a evolução do .NET Framework e veio para dar suporte a outros sistemas operacionais, como Linux e MacOS.

Naquela época, havia uma pressão dos usuários para que a Microsoft fornecesse suporte para computação na nuvem.

Então a criação do .NET Core possibilitou que milhões de pessoas que já utilizavam o .NET Framework agora pudessem trabalhar com cloud computing.

Porém, a Microsoft tinha um grande desafio...

Como migrar milhões de usuários de um sistema para outro?

As pessoas estavam acostumadas a utilizar o .NET Framework e deveriam passar para o .NET Core. Como tornar essa passagem mais suave?

A resposta foi reescrever o .NET Framework por inteiro.

Então o .NET Core é a reescrita do .NET Framework.

Foi uma decisão complexa, que demandou muito trabalho e foi extremamente custoso. E nas primeiras versões do .NET Core, ele continha apenas o essencial do .NET Framework, ele ainda não era 100% completo.

Mas, com a evolução do .NET Core, para as versões 2.0, 3.0, 3.1, o sistema ficou cada vez mais completo, possuindo quase a totalidade de recursos que o .NET Framework possuía.

Dessa forma, o uso do .NET Framework se tornou obsoleto.

Hoje em dia, caso você crie uma solução .NET, dificilmente utilizará o .NET Framework.

Você provavelmente vai utilizar o .NET Core 3.0 ou o .NET 5.0 e 6.0, dos quais falaremos mais adiante.

Essa transformação gigantesca nos sistemas da Microsoft, demandou a criação de uma camada, que fizesse a ligação entre o sistema legado e o novo sistema.

Atenção aqui!

Quando falo sobre sistema legado, me refiro a softwares e ferramentas que não são mais utilizados, ou não possuem mais suporte.

Então para que os dois sistemas fossem compatíveis, a Microsoft criou o .NET Standard, que facilitou a utilização do .NET Framework com o .NET Core.

Ou seja, agora você poderia utilizar os dois sistemas em um único projeto, sem nenhum prejuízo.

Isso facilitava muito a migração de sistemas antigos para o novo modelo.

Mas o .NET Standard não é mais um framework, beleza?!

Ele é apenas uma camada de comunicação entre os dois frameworks.

Mas agora chega a cereja do bolo. Chegou o momento em que o .NET Core superou o .NET Framework em conteúdo.

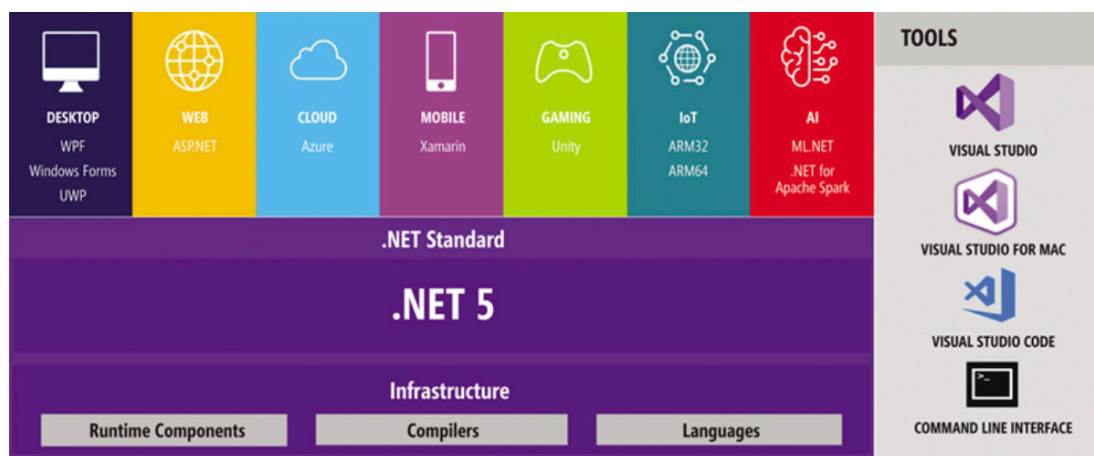
E com a chegada do .NET 5.0, a nomenclatura Core foi abolida, e passamos a chamá-lo somente de .NET

O .NET 5 é a unificação dos sistemas da Microsoft.

Mas, como assim unificação.

Agora o sistema .NET, que antes era dividido em suas porções, mobile, web, desktop, cloud entre outras funcionalidades, passa a ser unificado.

Dá só uma olhada nessa imagem oficial da Microsoft para entender melhor o que está acontecendo.



Através dessa imagem podemos entender que o plano da Microsoft é fazer com que as aplicações compartilhem de um mesmo código, linguagem, compiladores entre outras coisas.

Essa interoperabilidade ainda não estava 100% implementada com o .NET 5, mas desenhava o retrato da intenção que a Microsoft tinha, como ela mesma diz no artigo que deixarei nos detalhes desse vídeo.

Agora, mais importante que o .NET 5, e extremamente recente à data deste vídeo, é o lançamento do .NET 6.

Com o .NET 6 a promessa da Microsoft sobre unificar suas plataformas fica mais tangível.

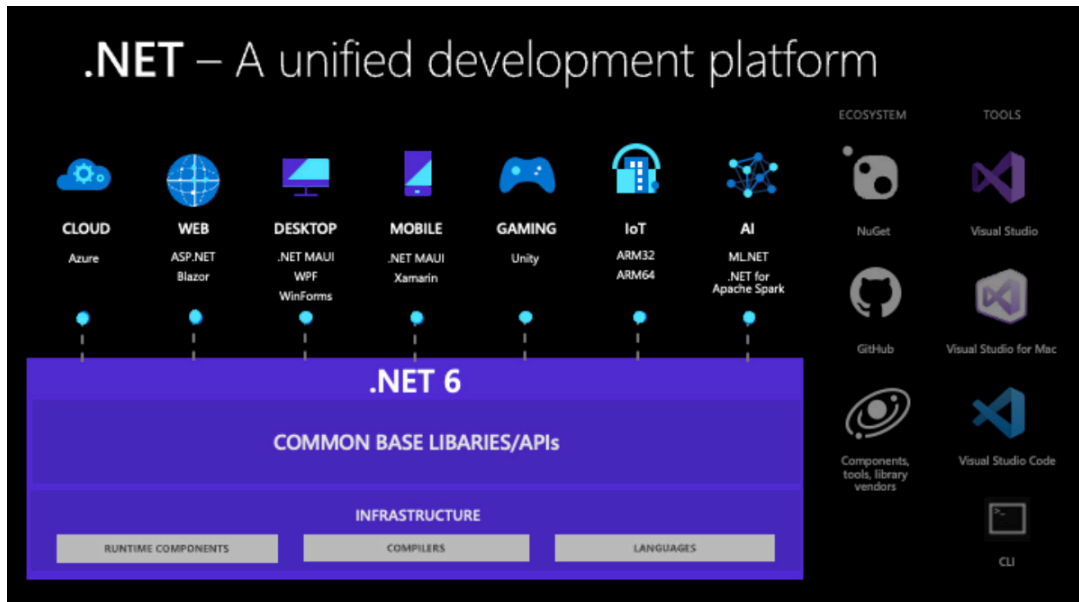
A principal mudança para esta versão é a performance. O time da Microsoft trabalhou arduamente para que a performance de seus aplicativos fosse melhorada.

E, além disso, temos a previsão do MAUI, que é o extensor do Xamarin. Ele vem para melhorar o desenvolvimento de aplicativos móveis, fazendo com que seja possível desenhar telas mais bonitas, ter mais compartilhamento de código entre as plataformas, mais performance, e dar suporte para desenvolvimento Android, iOS, Windows e MacOS.

Você já pode utilizar a preview do .NET MAUI com o .NET 6, mas até o momento desse vídeo não temos a versão final disponibilizada.

Além dessas novidades, temos dezenas de outras funções adicionadas à versão, e eu vou deixar um link nos detalhes deste vídeo, com um artigo do Medium que detalha todos os conteúdos sobre as novidades do .NET 6.

E essa imagem aqui é como o ecossistema .NET está atualmente



Com essa apresentação, podemos perceber que a Microsoft tem grandes planos para suas tecnologias, e você que está aprendendo C# pode estar por perto de toda essa evolução.

Pessoal, essa foi nossa aula de hoje, espero que tenham gostado e caso tenham alguma dúvida deixem aqui nos comentários que vamos te ajudar, beleza?

Na próxima aula daremos introdução ao estudo do C# na prática.

Nos vemos na próxima aula. Fui!

Links:

<https://mattwarren.org/2018/10/02/A-History-of-.NET-Runtimes/>

<https://executecommands.com/microsoft-net-framework-version-history/>

OBRIGADO.