

# .NET Conf 2021

del ConoSur

## Lanzamiento de .NET 6 y Visual Studio 2022



ORGANIZAN:





.NET CONF  
DEL CONO SUR

# Aplicaciones empresariales con Blazor y .NET 6



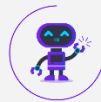
Lautaro Carro | @lauchacarro



CONO SUR TECH  
Comunidad de Tecnología

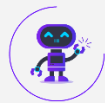


# WebAssembly



WebAssembly es un formato de instrucciones en binario que se ejecuta en el navegador.

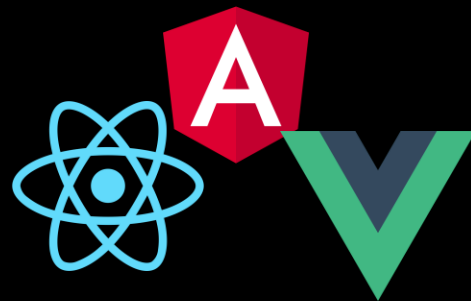




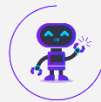
Al ser un formato binario, WebAssembly es mucho más rápido y potente que Javascript.



# JavaScript, el primer lenguaje FullStack



# WebAssembly: El futuro del desarrollo web



# DotNet Anywhere

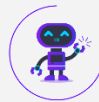




# Blazor



# ¿Qué es Blazor?



Blazor es una plataforma de trabajo (framework) para la creación de aplicaciones web utilizando C#, HTML y CSS.

Funciona con WebAssembly para correr C# en el navegador y también funciona en el servidor.





.NET CONF  
DEL CONO SUR

# Novedades de Blazor en .NET 6



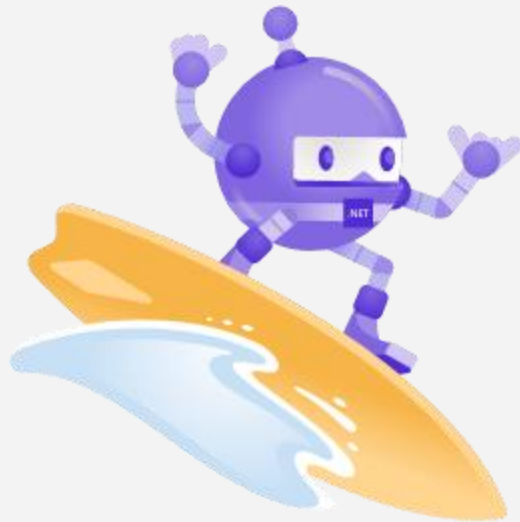
CONO SUR TECH  
Comunidad de Tecnología



# Nueva Plantilla para Blazor



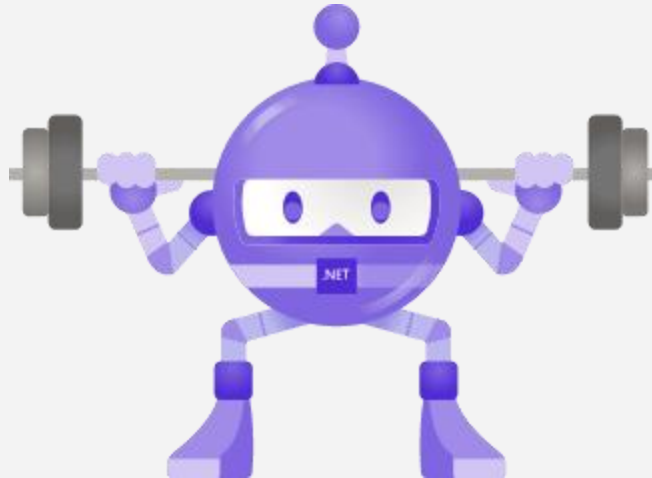
- Implicit using por defecto
- Top-level statements por defecto
- Bootstrap 5.1 sin dependencia de JavaScript
- Css Isolated por defecto
- Las páginas de ejemplo contienen <PageTitle>

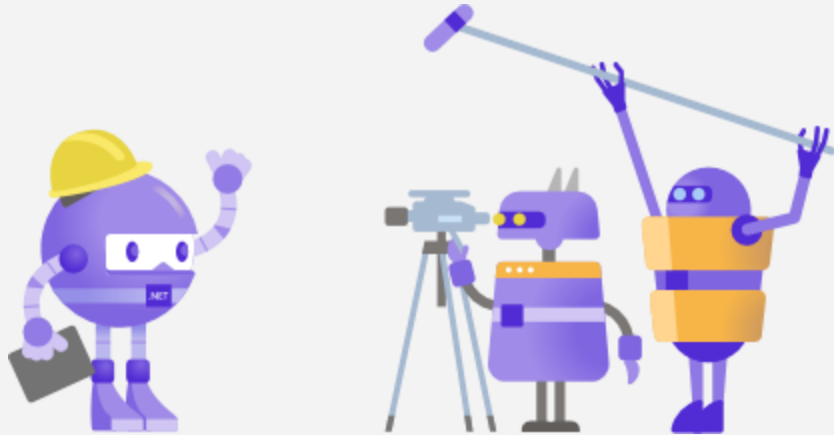


# Error Boundaries



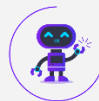
- Un componente para poder interceptar excepciones en nuestra aplicación y poder personalizar los mensajes de error.



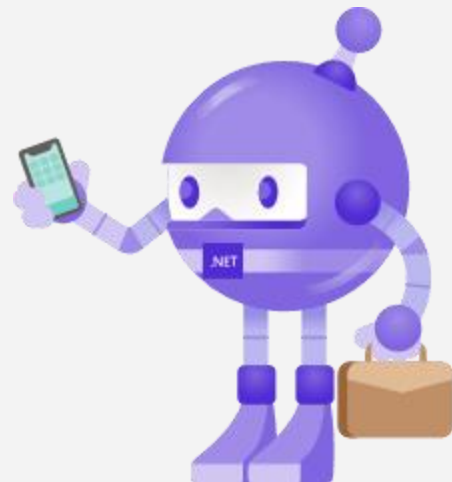


- `<PageTitle>` Nos permite cambiar el título del documento HTML
- `<HeadContent>` nos permite agregar tags como `<meta>` al contenido del Head del documento HTML

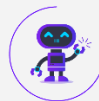
# Parámetros Requeridos



- El atributo [EditorRequired] es utilizado para decorar los parámetros que queremos que sean requeridos.
- Nos aparece un Warning en la lista de errores cuando llamamos al componente sin asignarle dicho parámetro.

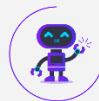


# Manipulación de Query Params



- El atributo `[SupplyParameterFromQuery]` es utilizado para decorar los parámetros.
- Aquellos parámetros que tengan este atributo tomarán el valor proveniente de la URL.

# Renderizar componentes dinámicamente



- El componente `<DynamicComponent>` es utilizado para renderizar componentes dinámicamente.
- `<DynamicComponent>` tiene el parámetro `Type` para pasar el componente a renderizar.
- También tiene el parámetro `Parameters` para pasar un `Dictionary<string, object>` con el nombre y valor de cada parámetro del componente que pasamos en el parámetro `Type`.







- <FocusOnNavigate> es un componente que tiene el parámetro Selector. Cada vez que se navega de una página a otra se le da foco al selector que se le indique.



## Otras Novedades



- Conservación del estado en Blazor Hosted Prerendering
- Dependencias Nativas de WebAssembly
- Compilación AOT (Compilación anticipada) (*Ahead-of-time compilation*)
- Renderizar componentes desde Javascript
- CustomEvents en Blazor
- Generar componentes de Angular o React usando Blazor



# Links para complementar



- [lauchacarro/LanzamientoNET6Conosur.Demo \(github.com\)](https://github.com/lauchacarro/LanzamientoNET6Conosur.Demo)
- <https://github.com/SteveSandersonMS/BlazeOrbital>
- [.NET 6: Las novedades en ASP.NET MVC y Blazor | campusMVP.es](https://campusMVP.es)
- [Final Blazor Improvements Before .NET 6 \(telerik.com\)](https://telerik.com)
- [Announcing .NET 6 - The Fastest .NET Yet - .NET Blog \(microsoft.com\)](https://microsoft.com)
- [AspLabs/src/BlazorCustomElements at main · aspnet/AspLabs \(github.com\)](https://github.com/aspnet/AspLabs)
- [samples/samples/aspnetcore/blazor/JSComponentGeneration at main · aspnet/samples \(github.com\)](https://github.com/aspnet/samples)
- [New Blazor WebAssembly capabilities in .NET 6 - YouTube](https://www.youtube.com/watch?v=...)
- [Enterprise-grade Blazor apps with .NET 6 - YouTube](https://www.youtube.com/watch?v=...)

# .NET Conf 2021

del ConoSur

## Lanzamiento de .NET 6 y Visual Studio 2022



ORGANIZAN:

