



GUSTAVO BELLINI BIGARDI

Aplicações web modernas utilizando C# com Blazor WASM

MVP WEEK LATAM



Glauber
Jannuzzi



Walter
Coan



Rodrigo
Crespi



Gustavo
Bigardi



Sara
Barbosa



Emmanuel
Brandão



Mario
Trentim



Luis
Beltran



Rafael
Mendonça



Flávio
Farias



Rodrigo
Dornel



Alessandro
Trovato



Bruno
Brito



Alexandre
Prado



Jefferson
Castilho



Carlos dos
Santos



Daniel
Donda



Albert
Tanure



Idevar
Junior



Marcelo
Adade



Thiago
Lunardi



Arthur Luz



Claudenir
Andrade



Armando
Lacerda



Angelo
Belchior



Murilo
Miranda



JUNDEVELOPERS



WE ARE THE
DIGITAL
MAKERS

O QUE FAZEMOS?

CRIAMOS PRODUTOS DIGITAIS ATRAVÉS DE PROJETOS DE
DESENVOLVIMENTO DE SOFTWARE SOB MEDIDA



MOBILE
APPS



WEB
APPLICATIONS



APIS



MACHINE
LEARNING



UX



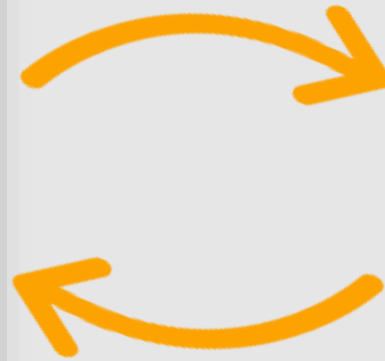
DESIGN
THINKING

TEMOS VAGAS ABERTAS!

ENVIE SEU CURRÍCULO OU SE INSCREVA NO SITE DE VAGAS DA DEXTRA

WWW.DEXTRA.COM.BR/CARREIRAS





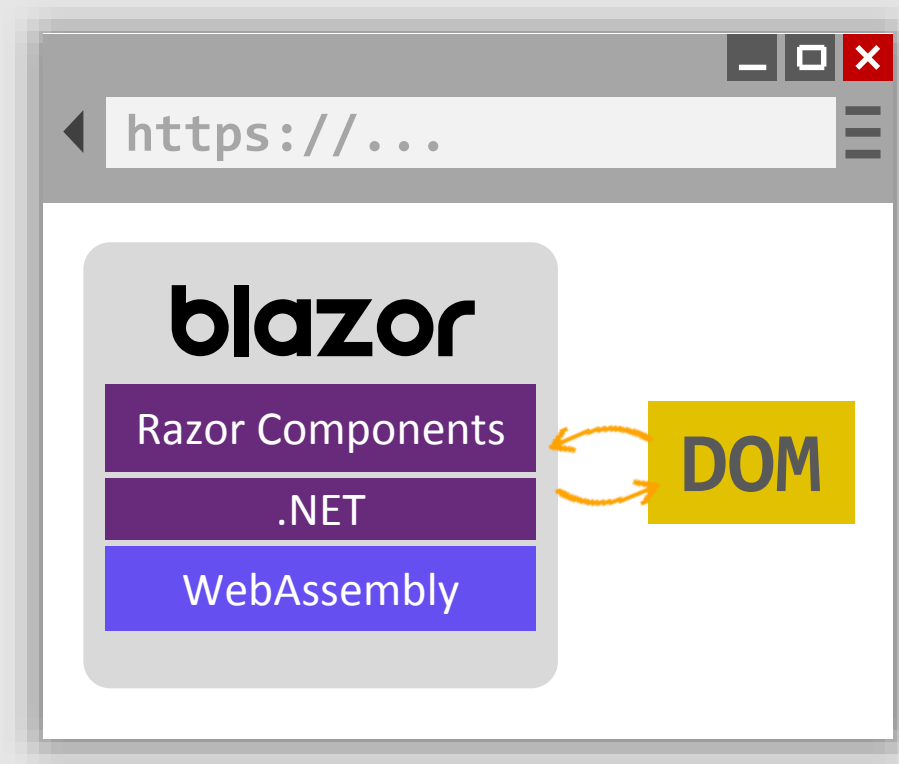
Blazor

- Produzir aplicações Web com C# ao invés de JS
- Produzir components reutilizáveis com C# e Razor
- Compartilhamento de Código entre cliente e servidor.
- Executar bibliotecas JS ou APIs do browser

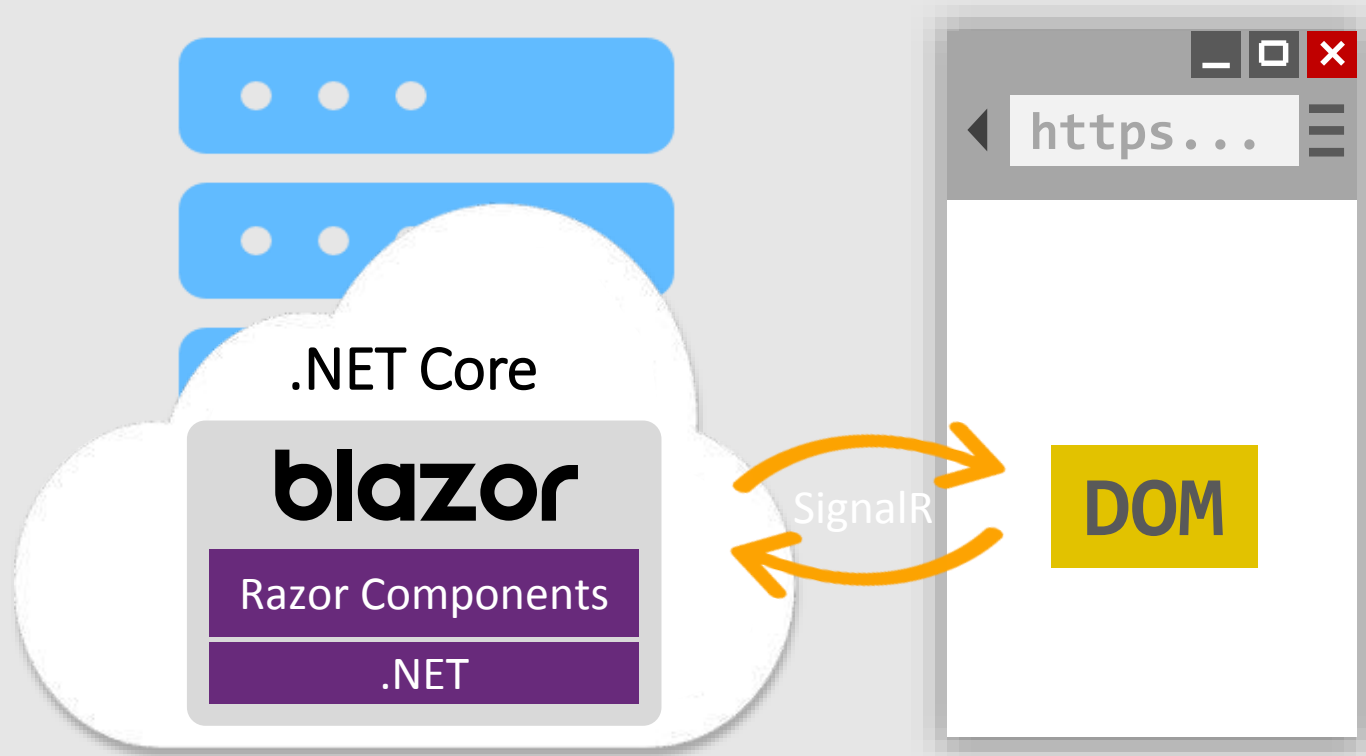
.NET

WA

Arquitetura básica do Blazor WebAssembly



Arquitetura básica do Blazor Server Side



Vantagens e desvantagens dos modelos

Blazor WebAssembly

Pro:

- Age como SPA, boa experiência ao usuário
- Utiliza recursos do cliente
- Suporte a offline, recursos estáticos e PWA.

Con:

- Download grande
- Requer suporte ao WA
- Ainda em preview

Blazor Server

Pro:

- Menor download, carrega rápido
- Suporte ao ambiente completo .NET
- Código permanece no servidor
- Arquitetura de front simplificada

Con:

- Latência e instabilidade de rede
- Sem suporte a offline
- Aumento no consumo de servidor.

Últimas novidades do Blazor

3.2.0 Preview 1 (28/01)	3.2.0 Preview 2 (10/03)	3.2.0 Preview 3 (26/03)	3.2.0 Preview 4 (16/04)
Versão atualizada 3.2	Integração com assets estáticos do ASP.NET	Suporte a Debug do WASM (limitado)	Acesso ao ambiente host na inicialização
Mudanças no Startup	Autenticação por Token	Auto-rebuild no Visual Studio	Melhorias em loggers
Redução no tamanho dos arquivos publicados	Templates com ASP.NET Core Identity e Identity Server	Suporte a Configurações	Compressão com Brotli
Suporte ao cliente do SignalR para WASM	Melhorias no cache da aplicação	Novos métodos para trabalhar com JSON no HttpClient	Carregamento do runtime e assemblies em paralelo
	Melhorias e template para aplicações PWA		Simplificação nas configurações do Linker
			Suporte a Localization (resx)
			Documentação das APIs no Intellisense

Iniciando com o Blazor

- <https://blazor.net>
- Instalar o .NET Core 3.1 LTS
- (Windows) Instalar o Visual Studio 2019 16.4 ou Preview
- (Mac/Linux) Instalar Visual Studio Code com a extensão C#
- Instalar os templates pelo CLI do .NET Core

```
dotnet new -i Microsoft.AspNetCore.Components.WebAssembly.Templates::3.2.0-preview4.20210.8
```



DEMO



DEMO

https://github.com/gustavobigardi/mvpweek_blazor



DÚVIDAS?

thanks!

