

# Realtime app con Blazor ed Azure



Sabato 24 settembre 2022



Nicola Paro 

# Nicola Paro

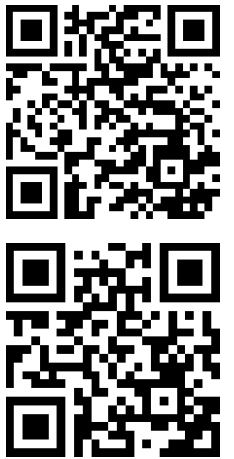
Cloud Solutions Engineer in beanTech



Linkedin: <https://www.linkedin.com/in/nicolaparo/>

Github: <https://github.com/nicolaparo>

Email: [nicola.paro@gmail.com](mailto:nicola.paro@gmail.com)

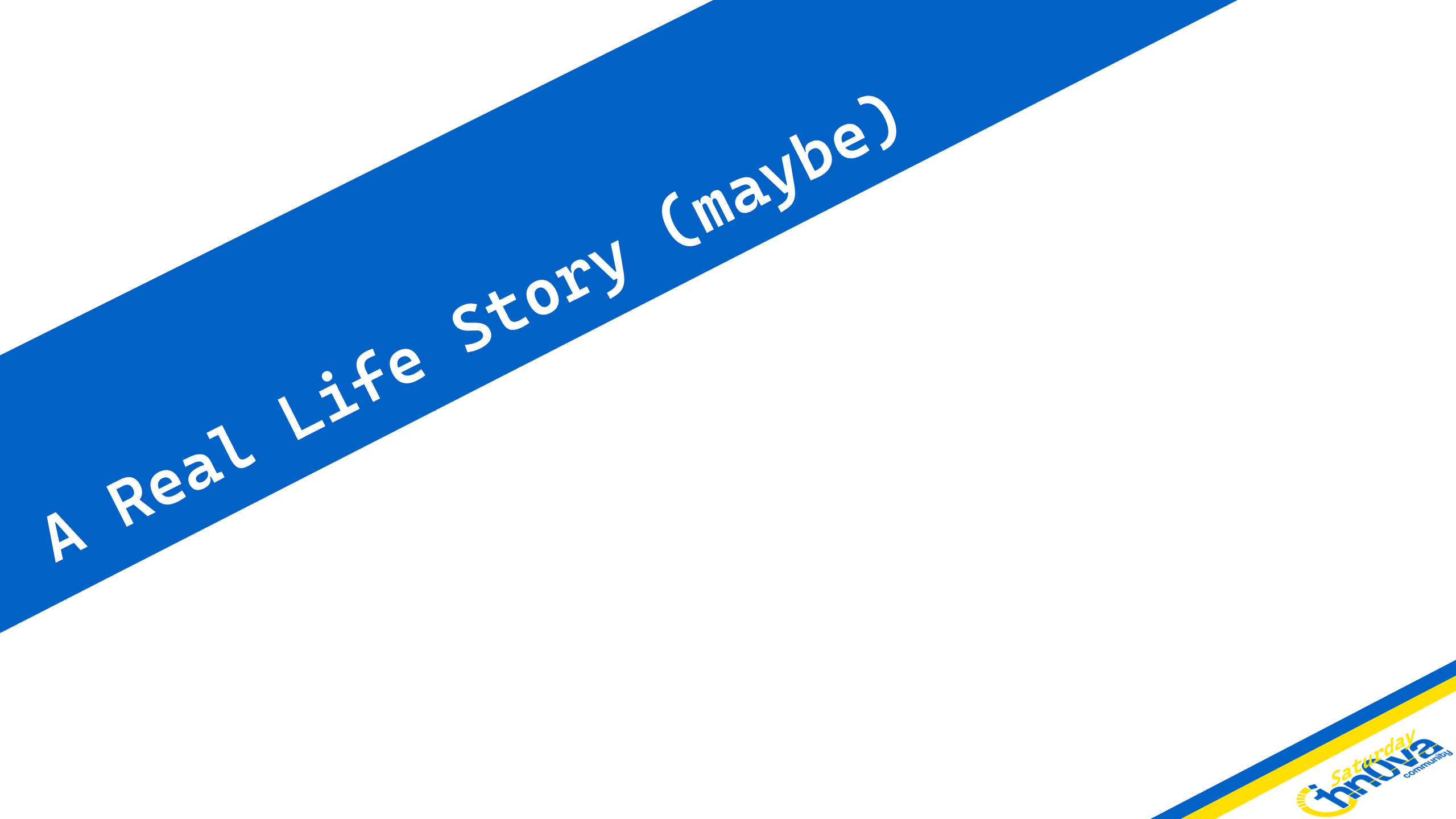


2020 - 2022



Grazie ai nostri sponsor





A Real Life Story (maybe)

# C'era una volta Itemfactory S.R.L. ...

- Azienda specializzata nella produzione di oggetti
- Linee di produzione già connesse ad un HMI
- I dati arrivano già in Cloud
- Vuole un applicativo MES per la gestione degli ordini di produzione
  - Lo vuole Desktop, Mobile e Web
  - Vuole che si aggiorni in tempo reale
  - Vuole che costi poco

The HMI interface shows the following data and controls:

Machine Name	Items per Minute
MACH01	0

Production Order	Fan Speed
	0

Set Items per Minute	Temperature °C
- 0 +	24.00

Set Fan Speed (rpm)	Consumption KW
- 0 +	0.00

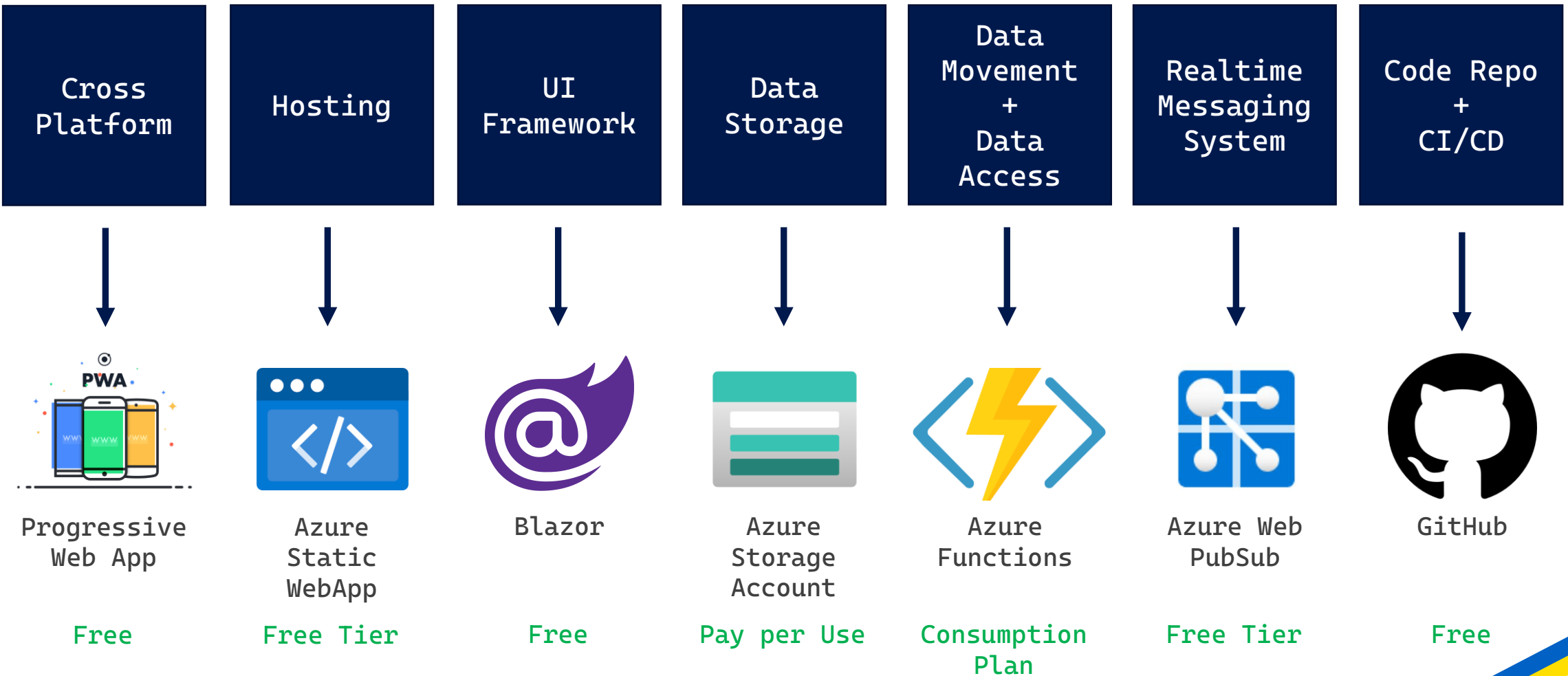
Total	Good	Rejects
0	0	0

A large red circular button labeled "EMERGENCY STOP" is located at the bottom center of the interface.

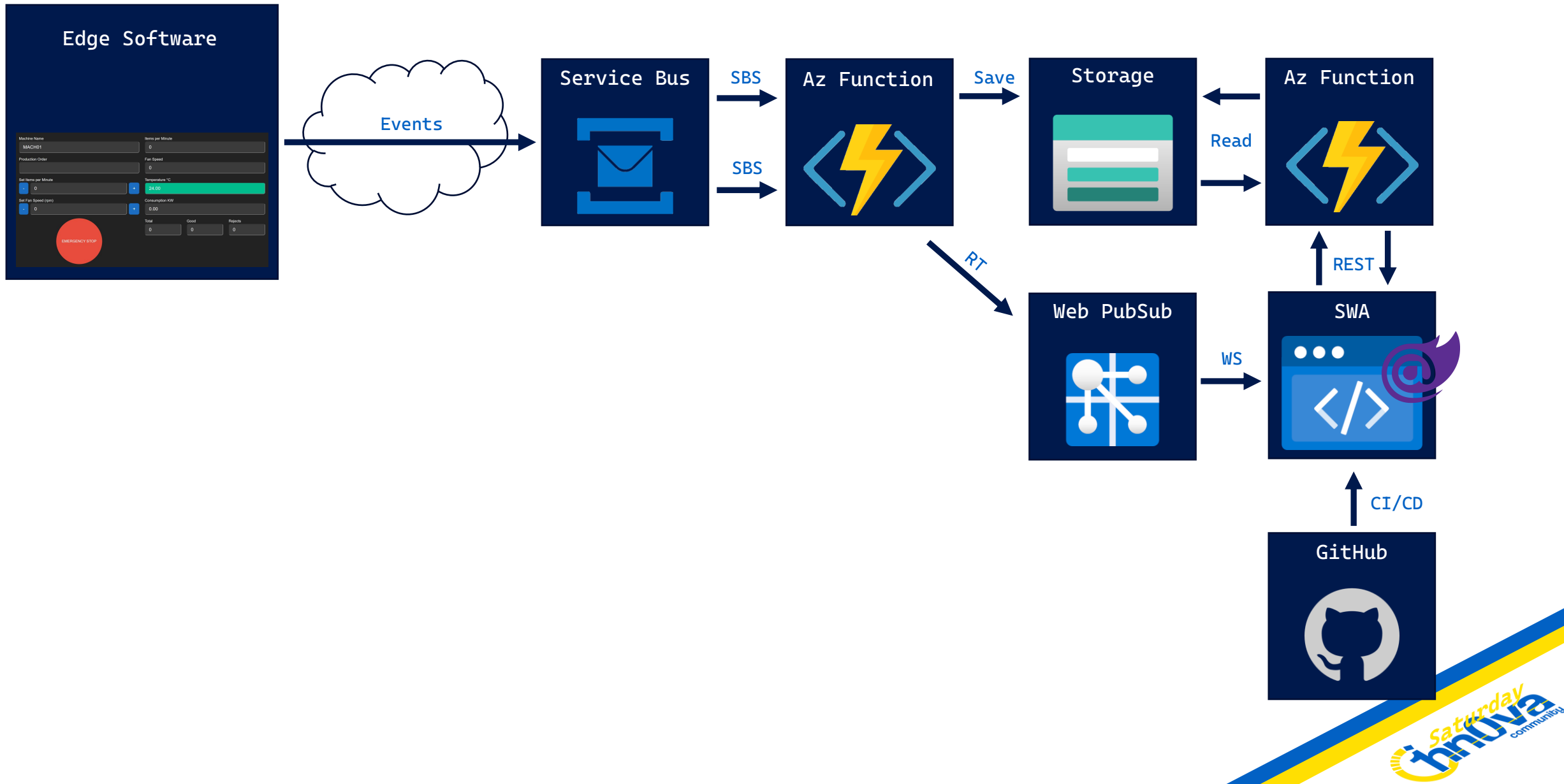
HMI 

(facciamo finta che lo sia)

# The Recipe



# The Architecture Schema



Blazor e PWA





# Cos'è Blazor

Blazor è un framework per la creazione di interfacce web interattive client-side utilizzando .NET.

Cosa possiamo fare?

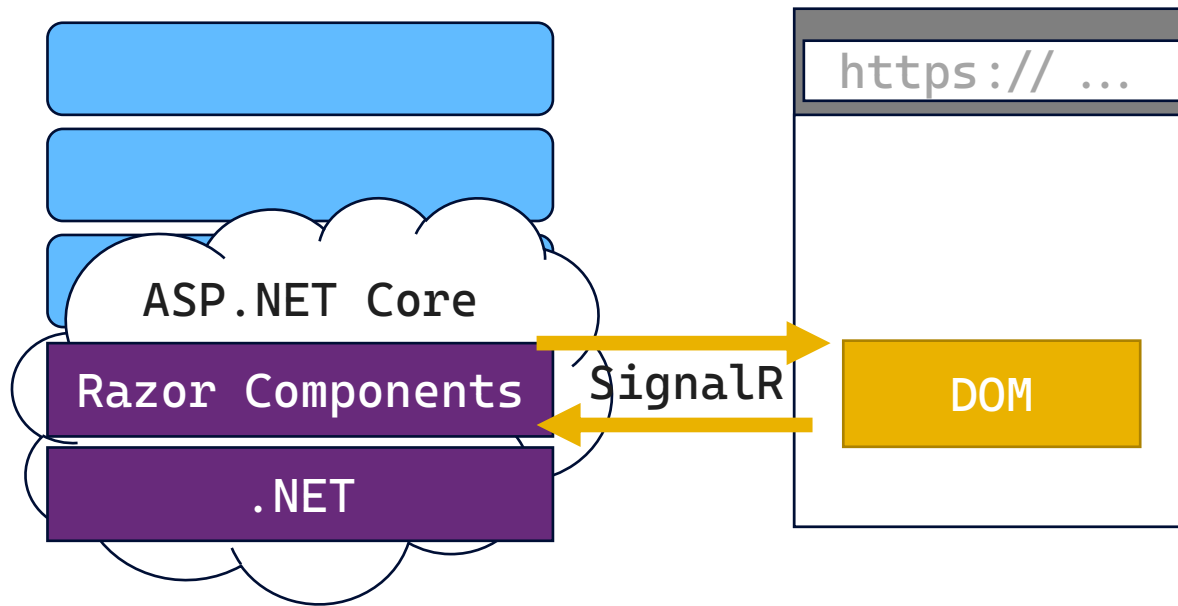
- Creare interface web utilizzando C# invece che JS\*
- Condividere logica tra server e client
- Utilizzare librerie .NET anche lato client\*\*

\* Non è vero, JS rimane per molte cose l'unica strada percorribile

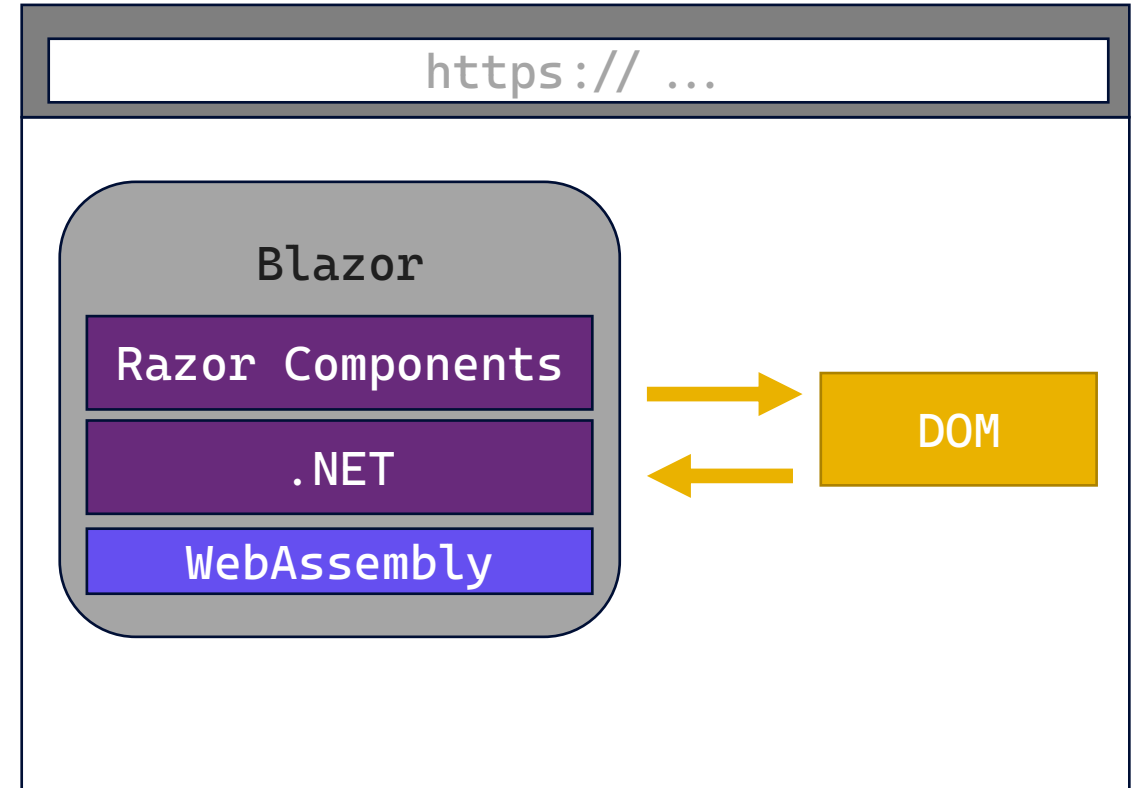
\*\* Con alcune restrizioni.

# Blazor Server vs WebAssembly

## Blazor Server



## Blazor WebAssembly (WASM)



# Cos'è una PWA?

È un applicativo web installabile.

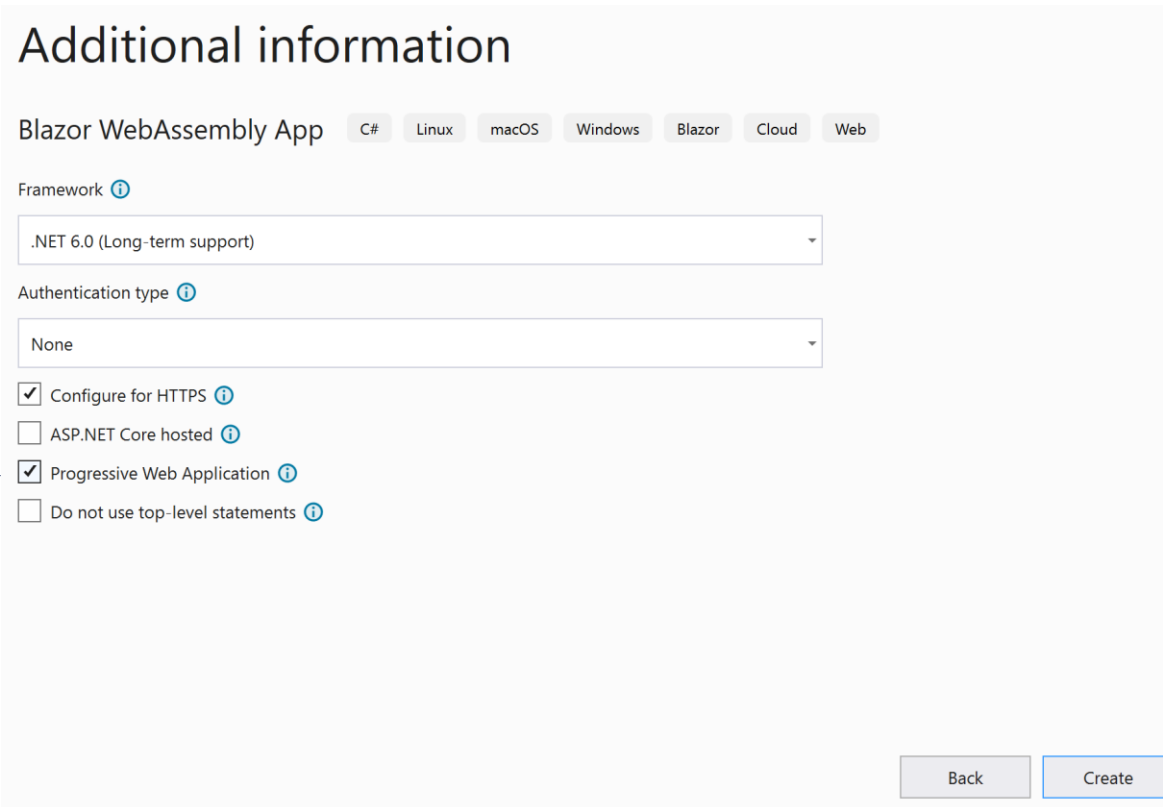
Deve soddisfare tutti i seguenti requisiti:

- Contenere un `manifest.json`.
- Servito in HTTPS
- Contenere un'icona per la Shortcut
- Deve registrare un `service worker`  
richiesto da tutti i browser Chromium-based

# Come posso creare una PWA con Blazor?

Opzioni:

1. Creo un'app ASP.NET Core e aggiungo gli assets per renderlo una PWA
2. `dotnet new blazorwasm -o AppName --pwa`
3. Così:



The screenshot shows the 'Additional information' page for creating a Blazor WebAssembly App. The 'Framework' is set to '.NET 6.0 (Long-term support)' and the 'Authentication type' is 'None'. The 'Progressive Web Application' checkbox is checked, which is highlighted by a blue arrow. The 'Do not use top-level statements' checkbox is also checked. The 'Back' and 'Create' buttons are at the bottom right.

Additional information

Blazor WebAssembly App C# Linux macOS Windows Blazor Cloud Web

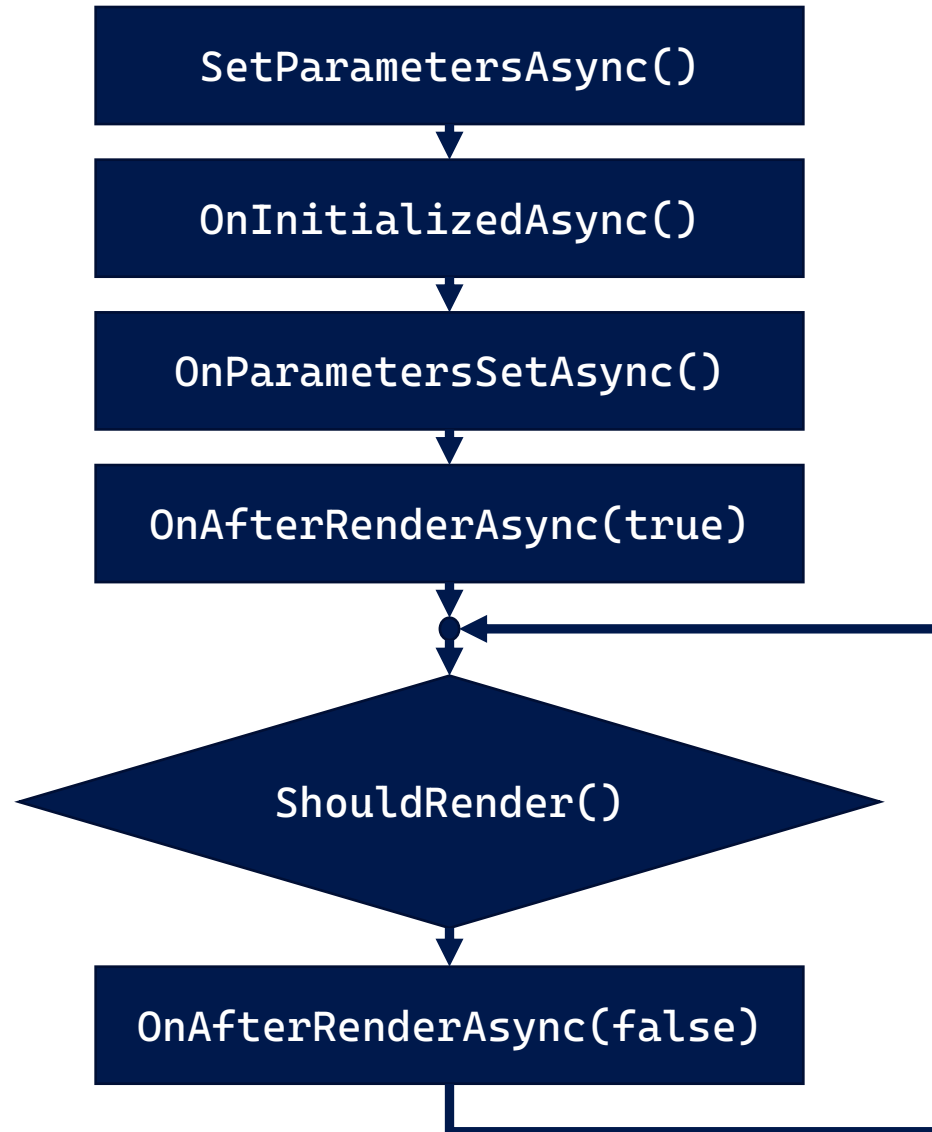
Framework ⓘ  
.NET 6.0 (Long-term support)

Authentication type ⓘ  
None

☒ Configure for HTTPS ⓘ  
☐ ASP.NET Core hosted ⓘ  
☒ Progressive Web Application ⓘ  
☒ Do not use top-level statements ⓘ

Back Create

# Razor Components Lifecycle



Special Guest

`DisposeAsync()`

`Dispose()` e `DisposeAsync()` sono invocati all'unload del componente se il componente implementa `IDisposable` o `IAsyncDisposable`

# Blazor Page Structure

```
@page "/"  
@inject FactorySensors sensors  
@implements IDisposable
```

```
<h1>Hello World</h1>
```

```
@code {  
    protected override Task OnAfterRenderAsync(bool firstRender)  
    {  
        return base.OnAfterRenderAsync(firstRender);  
    }  
  
    public void Dispose() { }  
}
```

# Azure Web PubSub

# Azure Web PubSub



Servizio gestito che supporta WebSockets serverless.  
Pensato per applicazioni che richiedono aggiornamenti in tempo reale

Aggiornamenti  
ad alta  
frequenza

Live  
dashboards e  
monitoring

Cross-  
platform live  
chat

Real-time  
location on  
map

Real-time  
targeted ads

Collaborative  
apps

Notifiche  
Push

Real-time  
broadcasting

IoT e  
dispositivi  
connessi

Automazione



# SignalR vs Azure Web PubSub



- Stai già usando SignalR
- Esiste un client SignalR nel linguaggio scelto
  - Clients supportati da MS:











- Hai necessità di più protocolli di trasporto
  - WebSocket
  - Server sent events
  - Long polling



- Hai bisogno solo di WebSocket
- Subprotocollo custom over WebSocket
  - MQTT over WebSocket
  - AMQP over WebSocket
- È sufficiente un'infrastruttura «lite» per la tua applicazione

\* Azure Web PubSub è solo PaaS

# Azure Web PubSub Pricing

Pricing tier	<input checked="" type="radio"/> Free For individual dev/test	<input type="radio"/> Standard For production workloads	<input type="radio"/> Premium(Preview) For production workloads with more supported features
Features			
 Connections	Up to 20 connections	1,000 connections/unit	1,000 connections/unit
 Included Messages	20,000/Day	1,000,000/Unit/Day	1,000,000/Unit/Day
 Additional Messages	-	Unlimited	Unlimited
 SLA	-	99.9%	99.95%
 SSL	✓	✓	✓
 Autoscale ⓘ	-	-	✓
 Availability Zone ⓘ	-	-	✓ (In supported regions) ↗
 Custom Domains	-	-	✓
Pricing			
Estimated Price	-	41.41 EUR/Month/Unit	51.44 EUR/Month/Unit
Additional Message Costs ⓘ	-	0.84 EUR per million messages	0.84 EUR per million messages

# DEMO – Azure Web PubSub

# Sending Data with Azure Web PubSub

- ServiceClient C#  
Azure.Messaging.WebPubSub
- AzFunction Bindings  
Microsoft.Azure.WebJobs.Extensions.WebPubSub

```
[FunctionName(nameof(EventNotifier))]  
public static async Task NotifyEventAsync(  
    [ServiceBusTrigger("%TopicName%", "%SubscriptionName%", Connection = "ServiceBusConnectionString")] string rawpayload  
    , [WebPubSub(Hub = "alarms", Connection = "WebPubSubConnectionString")] IAsyncCollector<WebPubSubAction> alarms  
    , ILogger log  
)  
{  
    var payload = Payloads.DeserializePayload(rawpayload);  
  
    if (payload is null)  
        return;  
  
    if (payload is AlarmPayload)  
    {  
        await alarms.AddAsync(WebPubSubAction.CreateSendToAllAction(rawpayload));  
        await alarms.FlushAsync();  
    }  
}
```

# Get AccessUri for Azure Web PubSub

- Tramite il portale Azure

- ServiceClient C#

```
var clientAccessUri = await serviceClient.GetClientAccessUriAsync();
```

- AzFunction Bindings

```
[FunctionName(nameof(NegotiateAlarms))]  
public static WebPubSubConnection NegotiateAlarms(  
[HttpTrigger(AuthorizationLevel.Anonymous, "get", "post", Route = "alarms/negotiate")] HttpRequest req,  
[WebPubSubConnection(Hub = "alarms", Connection = "WebPubSubConnectionString")] WebPubSubConnection connection)  
{  
    return connection;  
}
```

# DEMO – Notifiche in pagina



# Azure Static WebApp

# Cos'è Azure Static WebApp

È un servizio che consente di creare applicazioni Web statiche.

La pubblicazione dei nuovi contenuti avviene tramite un sistema di CI/CD.

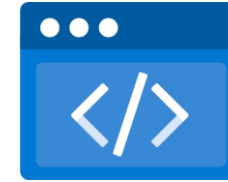
Si integra nativamente con GitHub e con Azure DevOps.



# Az Blob Static Website vs Az Static WebApp



- ✓ SSL
- 😞 CI/CD non gestito
- ✗ No Multiregion
- ✓ Custom Domain
- ✗ No HTTPS con Custom Domains



- ✓ SSL
- ✓ CI/CD integrate
- ✓ Globalmente distribuito
- ✓ Custom Domain
- ✓ HTTPS con Custom Domain
- ✓ API proxy integrato

# Azure Static WebApp – Api

## Free Tier

- Azure Functions

## Standard Tier

- Azure Functions
- Azure Api Management
- Azure AppService
- Azure Container Apps

# Azure Static WebApp Pricing

Plan/Features	<div><input checked="" type="radio"/></div> <b>Free</b> For hobby or personal projects	<div><input type="radio"/></div> <b>Standard</b> For general purpose production apps
Price	Free	7.59 EUR/per app/month
Included bandwidth	100GB per subscription	100GB per subscription
Bandwidth overage	Free	0.17 EUR per GB per subscription
Custom domains	2 per app	5 per app
SSL certificates	Free	Free
Custom authentication	-	✓
Private endpoints	-	✓
Max app size	250MB	500MB
Staging environments	3	10
Azure Functions	Managed	Managed or Bring your own
Enterprise-grade edge	-	14.77 EUR/per app/month

# DEMO – Azure Static WebApp

# Creare una Azure Static WebApp

Basics   Tags   Review + create

App Service Static Web Apps is a streamlined, highly efficient solution to take your static app from source code to global high availability. Pre-rendered content is distributed globally with no web servers required. [Learn more](#)

## Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Nicola Paro

Resource Group \* ⓘ

rg-blazormes

[Create new](#)

## Static Web App details

Name \*

mynewstaticwebapp

## Hosting plan

The hosting plan dictates your bandwidth, custom domain, storage, and other available features. [Compare plans](#)

Plan type

- ☒ Free: For hobby or personal projects
- ☐ Standard: For general purpose production apps

## Azure Functions and staging details

Region for Azure Functions API and staging environments \*

West Europe

## Deployment details

Source

- ☒ GitHub   ☐ Azure DevOps   ☐ Other

GitHub account

nicolaparo

[Change account](#)



# Creare una Azure Static WebApp

**i** If you can't find an organization or repository, you might need to enable additional permissions on GitHub. ✕

Organization \*

nicolaparo

Repository \*

blazormesdemo

Branch \*

main

## Build Details

Enter values to create a GitHub Actions workflow file for build and release. You can modify the workflow file later in your GitHub repository.

Build Presets

Blazor

**i** These fields will reflect the app type's default project structure. Change the values to suit your app.

App location \* **i**

Client

Api location **i**

Api

Output location **i**

wwwroot

[Preview workflow file](#)

[Review + create](#)

[< Previous](#)

[Next : Tags >](#)

## Framework

Angular

Blazor

Next.js

Nuxt.js

React

Svelte

Vue.js

## Static site generator

Gatsby

Hugo

VuePress

## Other

Custom

# Creare una Azure Static WebApp



<https://victorious-tree-017109303.1.azurestaticapps.net>

Q & A



# Nicola Paro

Cloud Solutions Engineer in beanTech



Linkedin: <https://www.linkedin.com/in/nicolaparo/>

Github: <https://github.com/nicolaparo>

Email: [nicola.paro@gmail.com](mailto:nicola.paro@gmail.com)



2020 - 2022



Thanks!



Sabato 24 settembre 2022

Marco Parenzan 