Blazor WASM/PWA - Configurazione AppSettings per più clienti

Obiettivo

Gestire pubblicazioni diverse della stessa app Blazor WebAssembly/PWA per più clienti, evitando di modificare manualmente `appsettings.json` dopo la pubblicazione.

Nelle app Blazor WASM/PWA i file pubblicati vengono gestiti anche dal service worker e dal manifest degli asset:

```
service-worker-assets.js
```

Se si modifica manualmente un file statico dopo la pubblicazione, ad esempio `appsettings.json`, si possono creare problemi di cache, integrità o caricamento della PWA.

La soluzione corretta è:

```
1. mantenere un appsettings.json comune
2. creare un file appsettings per ogni cliente
3. tenere i file cliente fuori da wwwroot
4. usare un profilo di pubblicazione per cliente
5. fare copiare al .csproj solo il file del cliente pubblicato
```

---

1. Struttura cartelle consigliata

Nel progetto Blazor WASM, ad esempio `Soluziona.Apps.Magazzino`, usare questa struttura:

```
Soluziona.Apps.Magazzino
├── SettingClienti
│   ├── Comside
│   │   └── appsettings.Comside.json
│   └── Demo
│       └── appsettings.Demo.json
│
├── Properties
│   └── PublishProfiles
│       ├── Comside.pubxml
│       └── Demo.pubxml
│
└── wwwroot
    ├── appsettings.json
    ├── appsettings.Development.json
    ├── index.html
    ├── manifest.webmanifest
    ├── service-worker.js
    └── ...
```

Regola

Dentro `wwwroot` devono rimanere solo:

```
appsettings.json
appsettings.Development.json
```

I file dei clienti devono stare fuori da `wwwroot`, dentro:

```
SettingClienti\<NomeCliente>\appsettings.<NomeCliente>.json
```

Esempio:

```
SettingClienti\Comside\appsettings.Comside.json
SettingClienti\Demo\appsettings.Demo.json
```

Non mettere dentro `wwwroot`:

```
appsettings.Comside.json
appsettings.Demo.json
```

altrimenti rischiano di essere pubblicati tutti insieme.

---

2. `appsettings.json` comune

Il file:

```
wwwroot\appsettings.json
```

deve contenere solo i valori comuni a tutti i clienti.

Esempio:

```
{
  "App": {
    "NomeApp": "Liste di Carico",
    "SottoTitoloApp": "Gestione preparazione ordini",
    "Versione": "1.0.6",
    "Build": "2026.09.09.006",
```

```
    "Ambiente": "Production",
    "PaginaIniziale": "/calendario"
  },
  "Api": {
    "TimeoutSecondi": 30
  },
  "Offline": {},
  "SyncListeDiCarico": {
    "IntervalloSecondi": 30,
    "GiorniIndietro": 15,
    "GiorniAvanti": 15,
    "EseguiAllAvvio": true
  }
}
```

Nota

In questo file può mancare:

```
"BaseUrl"
```

perché viene valorizzato dal file cliente, ad esempio:

```
appsettings.Comside.json
```

A runtime Blazor caricherà:

```
appsettings.json
+
appsettings.Comside.json
```

3. `appsettings.Development.json`

Il file:

```
wwwroot\appsettings.Development.json
```

viene usato per lo sviluppo locale.

Esempio:

```
{
  "App": {
    "IdApplicazione": "17630f17-7626-416f-961b-486fbd64fd4f",
    "CodiceApplicazione": "APPLISTEDICARICO",
    "CodiceAzienda": "3",
    "Ambiente": "Development"
  },
  "Api": {
    "BaseUrl": "https://localhost:7021/",
    "TimeoutSecondi": 30
  }
}
```

Questo file non deve essere pubblicato in produzione.

4. AppSettings per cliente

Demo

Percorso:

```
SettingClienti\Demo\appsettings.Demo.json
```

Contenuto:

```
{
  "App": {
    "IdApplicazione": "17630f17-7626-416f-961b-486fbd64fd4f",
    "CodiceApplicazione": "APPLISTEDICARICO",
    "CodiceAzienda": "3",
    "Ambiente": "Production"
  },
  "Api": {
    "BaseUrl": "https://soluzionacoreapi.soluziona.cloud:443/",
    "TimeoutSecondi": 30
  }
}
```

Comside

Percorso:

```
SettingClienti\Comside\appsettings.Comside.json
```

Contenuto:

```
{
  "App": {
    "IdApplicazione": "90208117-20b4-446c-a771-42bce3fb63c5",
    "CodiceApplicazione": "ListeDiCarico",
    "CodiceAzienda": "59",
    "Ambiente": "Production"
  },
  "Api": {
    "BaseUrl": "https://192.168.232.122:4433/",
    "TimeoutSecondi": 30
  }
}
```

---

5. Modifica del `.csproj`

Aprire il file `.csproj` del progetto Blazor WASM, ad esempio:

```
Soluziona.Apps.Magazzino.csproj
```

La configurazione deve:

```
1. escludere SettingClienti dagli item automatici del SDK
2. escludere eventuali appsettings ambiente da wwwroot
3. copiare il file cliente corretto dentro wwwroot prima della build/publish
4. includere esplicitamente il file copiato come Content
```

5.1 Esclusione file cliente e appsettings ambiente

Nel primo `PropertyGroup` aggiungere:

```
<DefaultItemExcludes>$(DefaultItemExcludes);SettingClienti\**\*;wwwroot\appsettings.*.json</De
faultItemExcludes>
```

Esempio:

```
<PropertyGroup>
  <TargetFramework>net10.0</TargetFramework>
  <Nullable>enable</Nullable>
  <ImplicitUsings>enable</ImplicitUsings>
  <OverrideHtmlAssetPlaceholders>true</OverrideHtmlAssetPlaceholders>
  <ServiceWorkerAssetsManifest>service-worker-assets.js</ServiceWorkerAssetsManifest>

  <!-- Esclude i file cliente dagli item automatici del .NET SDK -->

<DefaultItemExcludes>$(DefaultItemExcludes);SettingClienti\**\*;wwwroot\appsettings.*.json</De
faultItemExcludes>
</PropertyGroup>
```

Questa riga esclude:

```
SettingClienti\...
wwwroot\appsettings.Demo.json
wwwroot\appsettings.Comside.json
wwwroot\appsettings.Development.json
```

Non esclude:

```
wwwroot\appsettings.json
```

perché il file comune deve sempre essere pubblicato.

5.2 Reincludere `appsettings.Development.json`

Poiché `wwwroot\appsettings.*.json` viene escluso dagli item automatici, `appsettings.Development.json` va reincluso esplicitamente, ma senza pubblicarlo.

```
<Content Include="wwwroot\appsettings.Development.json">
  <CopyToOutputDirectory>PreserveNewest</CopyToOutputDirectory>
  <CopyToPublishDirectory>Never</CopyToPublishDirectory>
</Content>
```

5.3 Target per preparare l’appsettings cliente

Aggiungere questo target prima di:

```
</Project>
```

```
<Target Name="PreparaAppSettingsCliente"
        BeforeTargets="AssignTargetPaths"
        Condition="'$(WasmApplicationEnvironmentName)' != ''
                and '$(WasmApplicationEnvironmentName)' != 'Development'
                and
Exists('$(MSBuildProjectDirectory)\SettingClienti\$(WasmApplicationEnvironmentName)\appsetting
s.$(WasmApplicationEnvironmentName).json')">

  <Message Importance="High"
           Text="Preparo appsettings cliente:
SettingClienti\$(WasmApplicationEnvironmentName)\appsettings.$(WasmApplicationEnvironmentName)
.json" />

  <!-- Elimina eventuali appsettings cliente generati da build precedenti -->
  <ItemGroup>
    <AppSettingsClienteInWwwRoot
Include="$(MSBuildProjectDirectory)\wwwroot\appsettings.*.json"

Exclude="$(MSBuildProjectDirectory)\wwwroot\appsettings.Development.json" />
  </ItemGroup>

  <Delete Files="@(AppSettingsClienteInWwwRoot)" />

  <!-- Copia il file cliente dentro wwwroot prima della build/publish -->
  <Copy
SourceFiles="$(MSBuildProjectDirectory)\SettingClienti\$(WasmApplicationEnvironmentName)\appse
ttings.$(WasmApplicationEnvironmentName).json"

DestinationFiles="$(MSBuildProjectDirectory)\wwwroot\appsettings.$(WasmApplicationEnvironmentN
ame).json"
        SkipUnchangedFiles="false" />

  <!-- Include esplicitamente l'appsettings cliente generato come Content -->
  <ItemGroup>
    <Content Include="wwwroot\appsettings.$(WasmApplicationEnvironmentName).json">
      <CopyToOutputDirectory>PreserveNewest</CopyToOutputDirectory>
      <CopyToPublishDirectory>PreserveNewest</CopyToPublishDirectory>
    </Content>
  </ItemGroup>
```

```
</Target>
```

---

6. `.csproj` completo finale

```
<Project Sdk="Microsoft.NET.Sdk.BlazorWebAssembly">

  <PropertyGroup>
    <TargetFramework>net10.0</TargetFramework>
    <Nullable>enable</Nullable>
    <ImplicitUsings>enable</ImplicitUsings>
    <OverrideHtmlAssetPlaceholders>true</OverrideHtmlAssetPlaceholders>
    <ServiceWorkerAssetsManifest>service-worker-assets.js</ServiceWorkerAssetsManifest>

    <!-- Esclude i file cliente dagli item automatici del .NET SDK -->

<DefaultItemExcludes>$(DefaultItemExcludes);SettingClienti\**\*;wwwroot\appsettings.*.json</De
faultItemExcludes>
  </PropertyGroup>

  <ItemGroup>
    <PackageReference Include="DevExpress.Blazor" Version="26.1.4" />
    <PackageReference Include="DevExpress.Blazor.Themes" Version="26.1.4" />
    <PackageReference Include="Microsoft.AspNetCore.Components.WebAssembly" Version="10.0.11"
/>
    <PackageReference Include="Microsoft.AspNetCore.Components.WebAssembly.DevServer"
Version="10.0.11" PrivateAssets="all" />
    <PackageReference Include="Microsoft.Extensions.Http" Version="10.0.11" />
    <PackageReference Include="System.Security.Cryptography.Xml" Version="10.0.11" />
  </ItemGroup>

  <ItemGroup>
    <ProjectReference Include="..\Soluziona.Apps.UI\Soluziona.Apps.UI.csproj" />
    <ProjectReference Include="..\Soluziona.Apps.Utility\Soluziona.Apps.Core.csproj" />
    <ProjectReference Include="..\SoluzionaApps.ApiClient\Soluziona.Apps.ApiClient.csproj" />
```

```
  </ItemGroup>

  <ItemGroup>
    <ServiceWorker Include="wwwroot\service-worker.js" PublishedContent="wwwroot\service-
worker.published.js" />
  </ItemGroup>

  <ItemGroup>
    <Content Update="wwwroot\web.config">
      <CopyToOutputDirectory>Always</CopyToOutputDirectory>
    </Content>

    <!-- Development escluso dai Content automatici, quindi lo reincludiamo esplicitamente -->
    <Content Include="wwwroot\appsettings.Development.json">
      <CopyToOutputDirectory>PreserveNewest</CopyToOutputDirectory>
      <CopyToPublishDirectory>Never</CopyToPublishDirectory>
    </Content>
  </ItemGroup>

  <ItemGroup>
    <Folder Include="Properties\PublishProfiles\" />
  </ItemGroup>

  <Target Name="PreparaAppSettingsCliente"
          BeforeTargets="AssignTargetPaths"
          Condition="'$(WasmApplicationEnvironmentName)' != ''
                  and '$(WasmApplicationEnvironmentName)' != 'Development'
                  and
Exists('$(MSBuildProjectDirectory)\SettingClienti\$(WasmApplicationEnvironmentName)\appsetting
s.$(WasmApplicationEnvironmentName).json')">

    <Message Importance="High"
             Text="Preparo appsettings cliente:
SettingClienti\$(WasmApplicationEnvironmentName)\appsettings.$(WasmApplicationEnvironmentName)
.json" />

    <!-- Elimina eventuali appsettings cliente generati da build precedenti -->
    <ItemGroup>
      <AppSettingsClienteInWwwRoot
Include="$(MSBuildProjectDirectory)\wwwroot\appsettings.*.json"
```

```
Exclude="$(MSBuildProjectDirectory)\wwwroot\appsettings.Development.json" />
    </ItemGroup>

    <Delete Files="@(AppSettingsClienteInWwwRoot)" />

    <!-- Copia il file cliente dentro wwwroot prima della build/publish -->
    <Copy
SourceFiles="$(MSBuildProjectDirectory)\SettingClienti\$(WasmApplicationEnvironmentName)\appse
ttings.$(WasmApplicationEnvironmentName).json"

DestinationFiles="$(MSBuildProjectDirectory)\wwwroot\appsettings.$(WasmApplicationEnvironmentN
ame).json"
          SkipUnchangedFiles="false" />

    <!-- Include esplicitamente l'appsettings cliente generato come Content -->
    <ItemGroup>
      <Content Include="wwwroot\appsettings.$(WasmApplicationEnvironmentName).json">
        <CopyToOutputDirectory>PreserveNewest</CopyToOutputDirectory>
        <CopyToPublishDirectory>PreserveNewest</CopyToPublishDirectory>
      </Content>
    </ItemGroup>

  </Target>

</Project>
```

Nota importante per Git

Il target copia automaticamente il file cliente dentro `wwwroot`.

Esempio:

```
SettingClienti\Comside\appsettings.Comside.json
→ wwwroot\appsettings.Comside.json
```

Questo file in `wwwroot` è generato automaticamente e non deve essere committato.

Aggiungere al `.gitignore` in fondo:

```
# AppSettings cliente generati automaticamente dal publish
**/wwwroot/appsettings.*.json

# Mantiene versionato appsettings.Development.json
!**/wwwroot/appsettings.Development.json
```

`wwwroot\appsettings.json` non viene escluso da questa regola, quindi rimane versionato normalmente.

7. Profili di pubblicazione

I profili vanno creati da Visual Studio con:

```
Tasto destro sul progetto
→ Publish
→ New profile
→ Folder
```

Dopo la creazione, modificare i file `.pubxml`.

Profilo Demo

Percorso:

```
Properties\PublishProfiles\Demo.pubxml
```

Contenuto:

```
<?xml version="1.0" encoding="utf-8"?>
<Project>
  <PropertyGroup>
    <DeleteExistingFiles>true</DeleteExistingFiles>
    <ExcludeApp_Data>false</ExcludeApp_Data>
    <LaunchSiteAfterPublish>true</LaunchSiteAfterPublish>

    <LastUsedBuildConfiguration>Release</LastUsedBuildConfiguration>
    <LastUsedPlatform>Any CPU</LastUsedPlatform>

    <PublishProvider>FileSystem</PublishProvider>
    <PublishUrl>C:\Repos\Pubblicati\AppListeDiCarico\Demo</PublishUrl>
    <WebPublishMethod>FileSystem</WebPublishMethod>
    <_TargetId>Folder</_TargetId>

    <WasmApplicationEnvironmentName>Demo</WasmApplicationEnvironmentName>
  </PropertyGroup>
</Project>
```

Profilo Comside

Percorso:

```
Properties\PublishProfiles\Comside.pubxml
```

Contenuto:

```
<?xml version="1.0" encoding="utf-8"?>
<Project>
  <PropertyGroup>
    <DeleteExistingFiles>true</DeleteExistingFiles>
```

```
    <ExcludeApp_Data>false</ExcludeApp_Data>
    <LaunchSiteAfterPublish>true</LaunchSiteAfterPublish>

    <LastUsedBuildConfiguration>Release</LastUsedBuildConfiguration>
    <LastUsedPlatform>Any CPU</LastUsedPlatform>

    <PublishProvider>FileSystem</PublishProvider>
    <PublishUrl>C:\Repos\Pubblicati\AppListeDiCarico\Comside</PublishUrl>
    <WebPublishMethod>FileSystem</WebPublishMethod>
    <_TargetId>Folder</_TargetId>

    <WasmApplicationEnvironmentName>Comside</WasmApplicationEnvironmentName>
  </PropertyGroup>
</Project>
```

---

8. Proprietà fondamentale: `WasmApplicationEnvironmentName`

La riga più importante del profilo è questa:

```
<WasmApplicationEnvironmentName>Comside</WasmApplicationEnvironmentName>
```

oppure:

```
<WasmApplicationEnvironmentName>Demo</WasmApplicationEnvironmentName>
```

Questa proprietà decide quale ambiente Blazor deve usare.

Esempio:

```
WasmApplicationEnvironmentName = Comside
→ carica appsettings.json
→ carica appsettings.Comside.json
```

```
WasmApplicationEnvironmentName = Demo
→ carica appsettings.json
→ carica appsettings.Demo.json
```

Il valore deve combaciare esattamente con:

```
nome cartella cliente
nome file appsettings
```

Esempio corretto:

```
Comside
→ SettingClienti\Comside\appsettings.Comside.json
```

Esempio errato:

```
COMSIDE
→ SettingClienti\Comside\appsettings.Comside.json
```

---

9. Nota su `App:Ambiente`

Dentro gli appsettings cliente si può lasciare:

```
"Ambiente": "Production"
```

Questo valore è una configurazione applicativa interna.

Non decide quale file `appsettings.*.json` viene caricato.

Il file caricato viene deciso da:

```
<WasmApplicationEnvironmentName>Comside</WasmApplicationEnvironmentName>
```

non da:

```
"Ambiente": "Production"
```

Revisione #5
Creato 2026-09-10 06:19:14 UTC da Admin
Aggiornato 2026-09-29 14:45:31 UTC da Admin