

Publicação e disponibilização do Portal Conciliador + Api em ambiente produção

Dependências:

1 - Ter sido criado do App no Azure, lembrando que o mesmo deve ser configurado como um SPA

Microsoft Azure

Home > Diretório Padrão | App registrations > Conciliador

Conciliador | Authentication

Search resources, services, and docs (G+/J)

Search

Got feedback?

Overview

Quickstart

Integration assistant

Manage

Branding & properties

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles

Owners

Roles and administrators

Manifest

Platform configurations

Depending on the platform or device this application is targeting, additional configuration may be required such as redirect URIs, specific authentication settings, or fields specific to the platform.

+ Add a platform

Single-page application

Quickstart Docs

Redirect URIs

The URIs we will accept as destinations when returning authentication responses (tokens) after successfully authenticating or signing out users. The redirect URI you send in the request to the login server should match one listed here. Also referred to as reply URLs. [Learn more about Redirect URIs and their restrictions](#)

http://localhost:4200

Add URI

Grant types

✓ Your Redirect URI is eligible for the Authorization Code Flow with PKCE.

Configure platforms

Web applications

Web

Build, host, and deploy a web server application. .NET, Java, Python

Single-page application

Configure browser client applications and progressive web applications. Javascript.

Confira no manifesto do app do Azure se está como SPA

```
56 "replyUrlsWithType": [  
57   {  
58     "url": "http://localhost:4200",  
59     "type": "Spa"  
60   }  
61 ],  
62 "requiredResourceAccess": [
```

No exemplo acima o endereço da url do cliente será

"url": "<https://conciliador.futuro.digital>"

NOTA

O nome do HostName (DNS) e certificado devem ser fornecidos para configuração do IIS.

2 - No servido onde está o IIS devem estar instalados o **Runtime do Net.Core 6**, o **url-rewrite** e o **windows-hosting-bundle-installer**.

Link do ur-rewrite

<https://iis-umbraco.azurewebsites.net/downloads/microsoft/url-rewrite>

Link do host-bundle

<https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-windows-hosting-bundle-installer>

Caso o Host-bundle não instale o runtime do net.core instale o mesmo com o link abaixo:

<https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-windows-x64-installer>

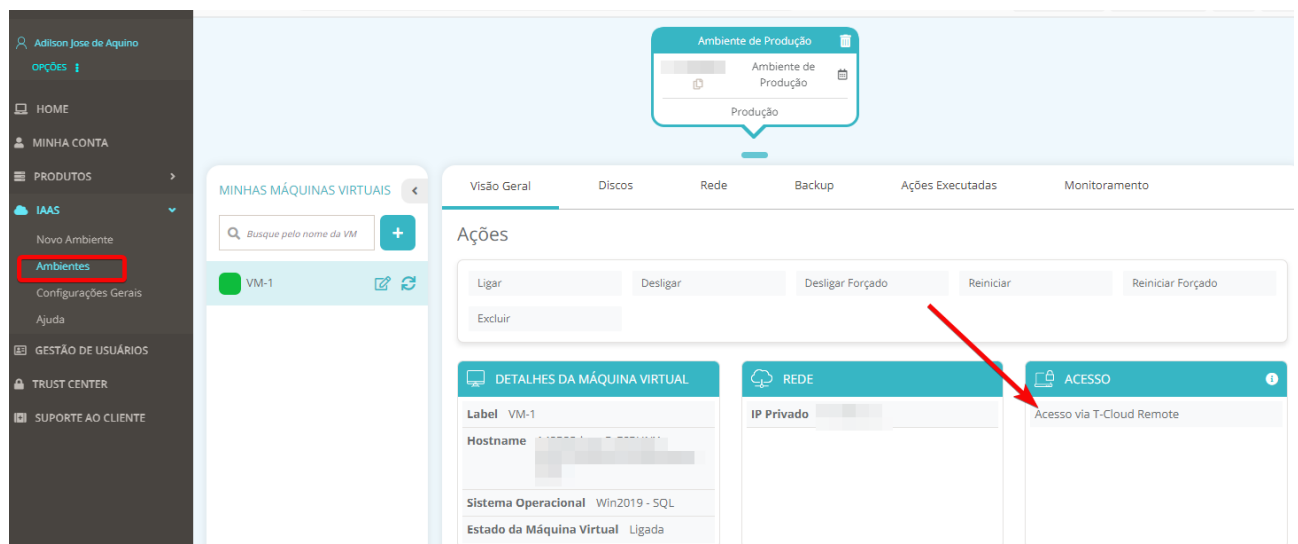
Essa configuração é para o Net.Core 6, caso venha a ser atualizado para o 7 ou posterior runtime e host-bundle devem ser instalados de acordo com a versão.

3 - Base de dados acessível com usuário e senha para configuração da connection string;

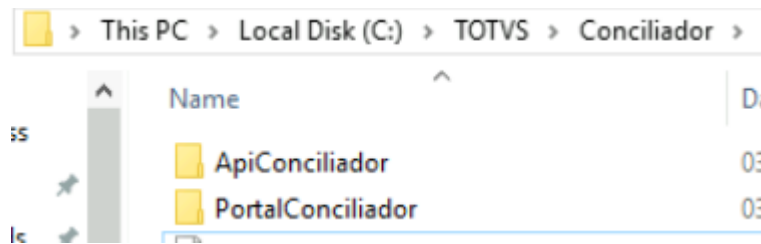
4 - Execução do script **ScriptTabelasConciliadorDB.sql** para criação das tabelas no banco de dados.

5 - Acesso o servidor através do T-Cloud da TOTVS

<https://tcloud.totvs.com.br/login>



6 - Execute os dois instaladores, um do Portal do Conciliador e o outro da Api do Conciliador. Serão criadas duas pastas no caminho de instalação TOTVS.



Os arquivos "Web.config" do portal e da API são diferentes e devem constar nos seus respectivos diretórios.

Conteúdo do Web.config da Api

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <location path="." inheritInChildApplications="false">
    <system.webServer>
      <handlers>
        <add name="aspNetCore" path="*" verb="*"
              modules="AspNetCoreModuleV2" resourceType="Unspecified" />
      </handlers>
      <aspNetCore processPath="dotnet"
                  arguments=".\ConciliadorApi.dll"
                  stdoutLogEnabled="false"
                  stdoutLogFile=".\logs\stdout"
                  hostingModel="inprocess" />
    </system.webServer>
  </location>
  <system.webServer>
    <rewrite>
      <rules>
        <remove name="Angular Routes" />
        <rule name="Angular Routes" enabled="false"
              stopProcessing="true">
          <match url="*" />
          <conditions logicalGrouping="MatchAll">
            <add input="{REQUEST_FILENAME}" matchType="IsFile"
                  negate="true" />
            <add input="{REQUEST_FILENAME}"
                  matchType="IsDirectory" negate="true" />
          </conditions>
          <serverVariables />
          <action type="Rewrite" url="./index.html" />
        </rule>
      </rules>
    </rewrite>
  </system.webServer>
</configuration>
```

```
        </rules>
    </rewrite>
</system.webServer>
</configuration>
```

Conteúdo do Web.config do Portal

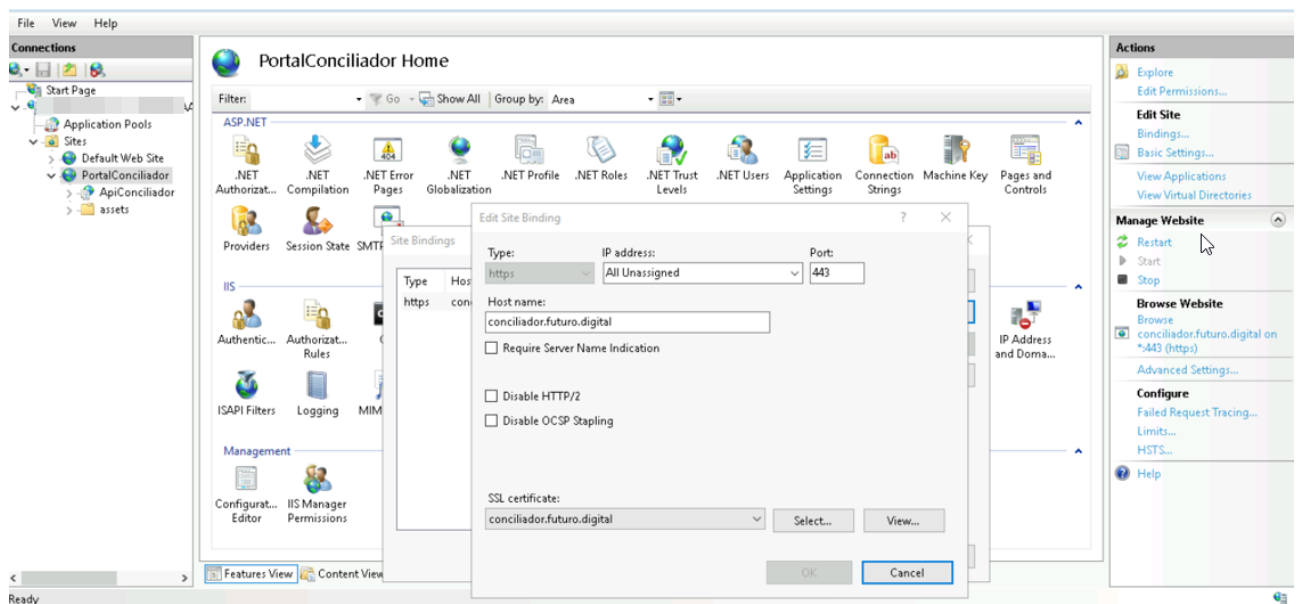
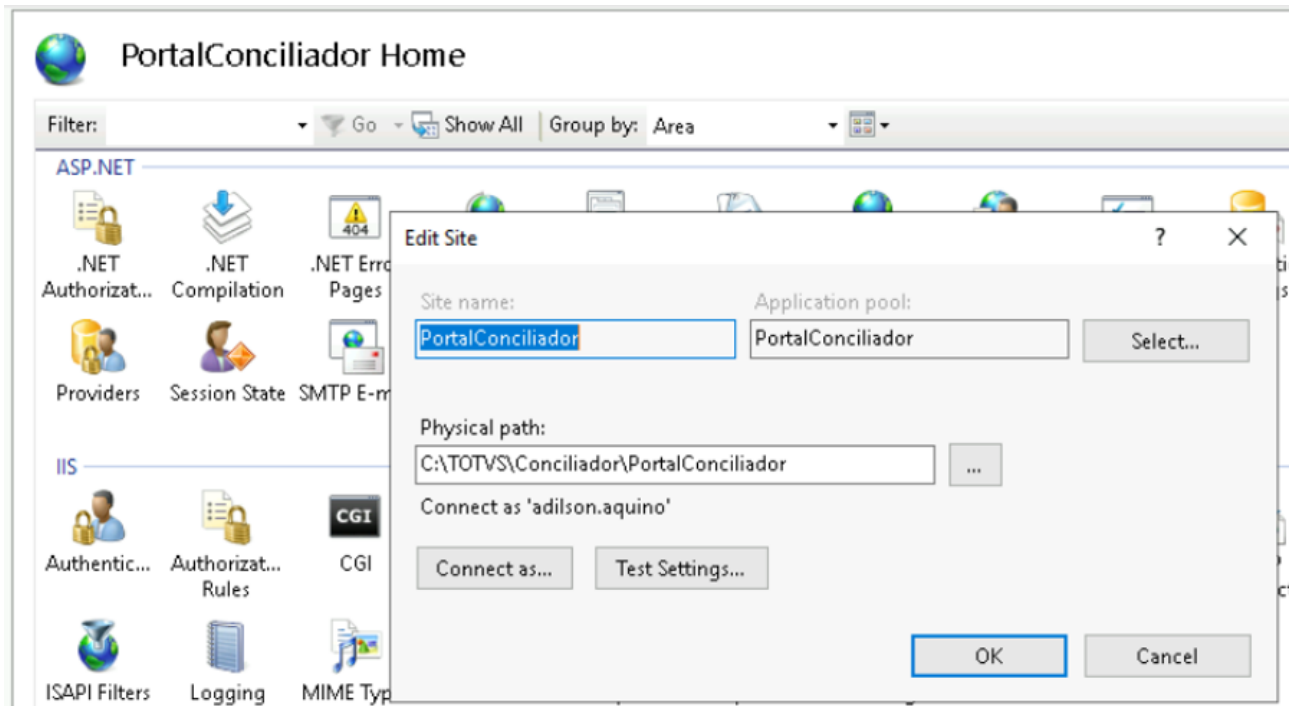
```
<?xml version="1.0" encoding="utf-8"?>
<configuration>

<system.webServer>
  <rewrite>
    <rules>
      <rule name="Angular Routes" stopProcessing="true">
        <match url=".*" />
        <conditions logicalGrouping="MatchAll">
          <add input="{REQUEST_FILENAME}" matchType="IsFile" negate="true"
/>
          <add input="{REQUEST_FILENAME}" matchType="IsDirectory"
negate="true" />
        </conditions>
        <action type="Rewrite" url="./index.html" />
      </rule>
    </rules>
  </rewrite>
  <staticContent>
    <remove fileExtension=".json"/>
    <mimeMap fileExtension=".json" mimeType="application/json" />
    <mimeMap fileExtension=".webmanifest"
mimeType="application/json" />
  </staticContent>
</system.webServer>
</configuration>
```

Configuração do IIS

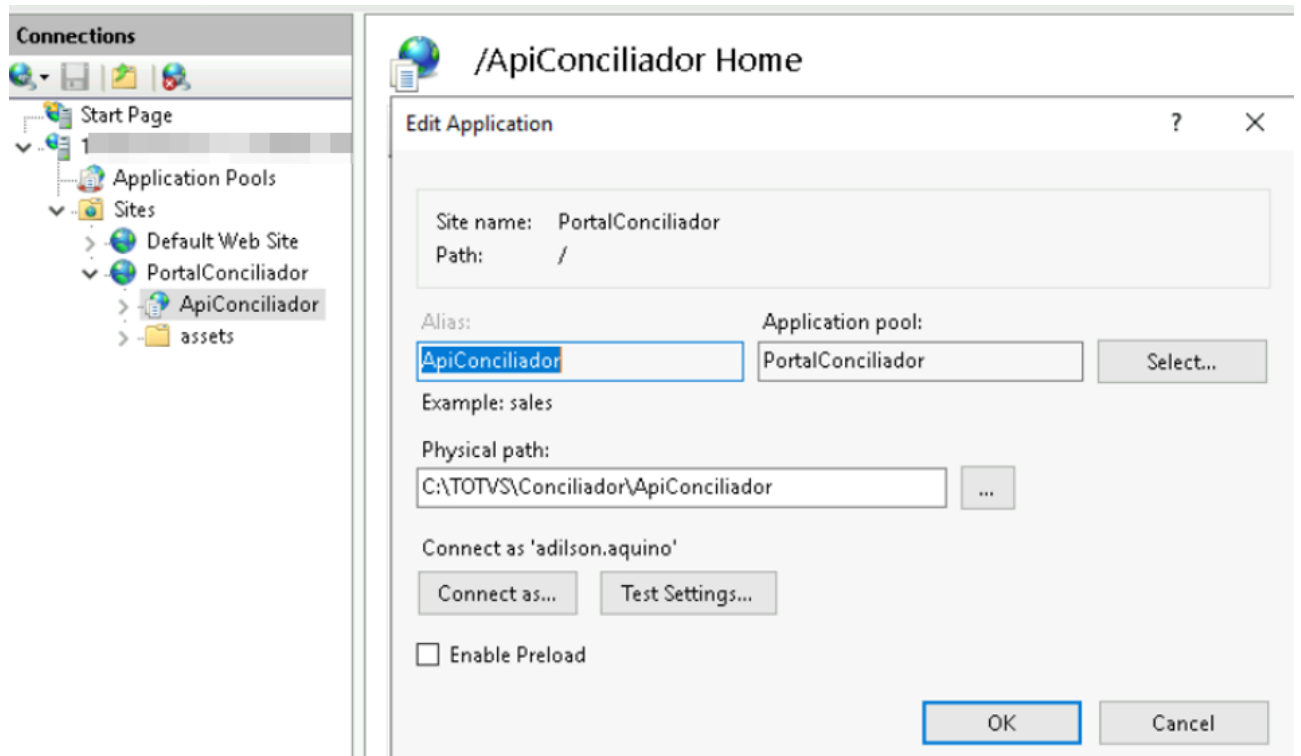
Configuração do Portal

Crie um novo site no IIS com nome PortalConciliador. Este site deve usar https na porta 443, ter o hostname informado, bem como o certificado SSL.



Configuração da API

Dentro do site “PortalConciliador” insira um novo “Application” com nome “ApiConciliador”



Configure o acesso “Connect as”.

Realizadas essas configuração o portal estará pronto para ser acessado.