

Set up SSO for Yapoli on Azure AD

To set up a SSO access on Yapoli via Azure AD it is necessary to make the link on Azure before registering on Yapoli to define the parameters.

To do so, follow these steps:

1st Step:

On Azure AD go on APP Registrations > New Registration. The screen below must be shown. Enter any name of your liking, verify which set of users must have permission to access on “Supported account types” and, on “Redirect URI”, select the “Web” platform and enter the following url: ***https://<dominio-da-plataforma-na-yapoli>/login-oauth***. Example: <https://cliente.yapoli.com/login-oauth>. If the login is made from a custom domain, enter this domain, ex: <https://client.com/login-oauth> . Click on “Register”.

[Home](#) > [Diretório Padrão](#) | [App registrations](#) >

Register an application ...

* Name

The user-facing display name for this application (this can be changed later).

Supported account types

Who can use this application or access this API?

- Accounts in this organizational directory only (Diretório Padrão only - Single tenant)
- Accounts in any organizational directory (Any Azure AD directory - Multitenant)
- Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Select a platform ▼	e.g. https://example.com/auth
---------------------	--

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from [Enterprise applications](#).

By proceeding, you agree to the [Microsoft Platform Policies](#) 


Register

2nd Step:

To create a new password where Yapoli can authorize your domain, go to Certificates & Secrets and, on the tab “Client Secrets” click on “New Client Secret”. A window will open on the right, choose your identifier for this password and how long until the password expires. After that, the generated password value will show on the list just once. Click to copy that password and store it to use on the Yapoli registration.

 Got feedback?

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

 Application registration certificates, secrets and federated credentials can be found in the tabs below. ✕

Certificates (0) **Client secrets (0)** Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

 New client secret

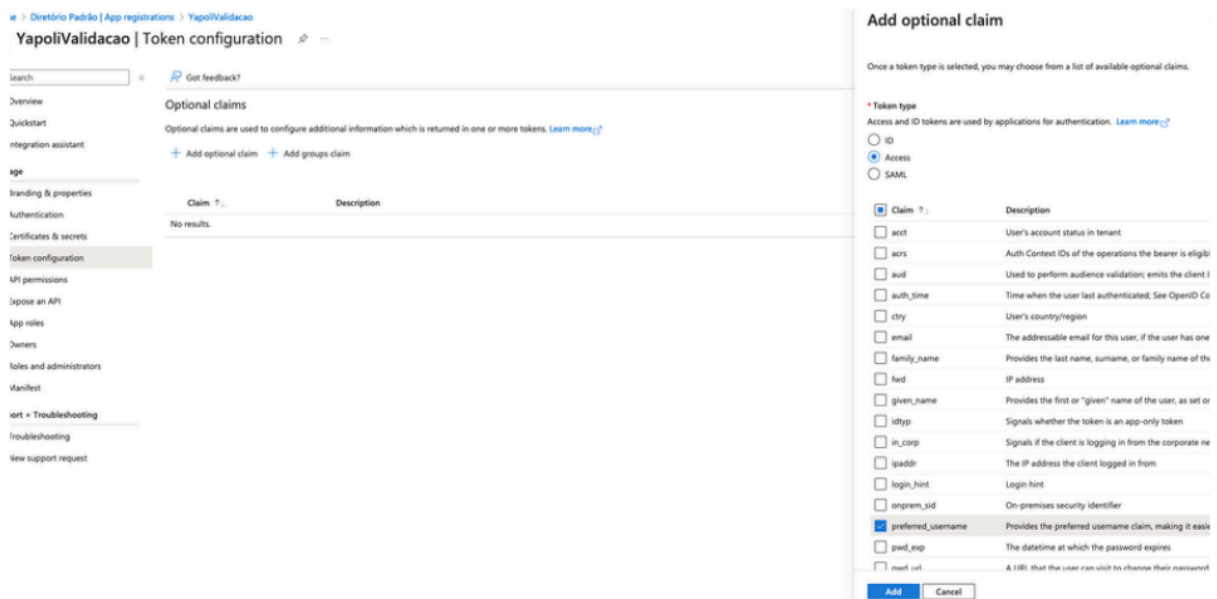
Description	Expires	Value 	Secret ID
-------------	---------	---	-----------

No client secrets have been created for this application.

3rd Step:

Go to “Token Configuration” and click on “Add optional claim”. On the menu shown on the right, click on “Access” and on the list shown below, “preferred_username”. At the end, click on “Add”.

Obs: This step is necessary so that the preferential name is the user identifier. This identifier needs to be the user email that is used to enter the Yapoli platform.



Add optional claim

Once a token type is selected, you may choose from a list of available optional claims.

Token type

Access and ID tokens are used by applications for authentication. [Learn more](#)

ID

Access

SAML

<input checked="" type="checkbox"/> Claim	Description
<input type="checkbox"/> acct	User's account status in tenant
<input type="checkbox"/> acrs	Auth Context IDs of the operations the bearer is eligible for
<input type="checkbox"/> aud	Used to perform audience validation; emits the client ID
<input type="checkbox"/> auth_time	Time when the user last authenticated. See OpenID Connect
<input type="checkbox"/> cty	User's country/region
<input type="checkbox"/> email	The addressable email for this user, if the user has one
<input type="checkbox"/> family_name	Provides the last name, surname, or family name of the user
<input type="checkbox"/> fwp	IP address
<input type="checkbox"/> given_name	Provides the first or "given" name of the user, as set on the user profile
<input type="checkbox"/> idtyp	Signals whether the token is an app-only token
<input type="checkbox"/> in_corp	Signals if the client is logging in from the corporate network
<input type="checkbox"/> ipaddr	The IP address the client logged in from
<input type="checkbox"/> login_hint	Login hint
<input type="checkbox"/> onprem_sid	On-premises security identifier
<input checked="" type="checkbox"/> preferred_username	Provides the preferred username claim, making it easier to use for authentication
<input type="checkbox"/> pwd_exp	The datetime at which the password expires
<input type="checkbox"/> reset_url	A URL that the user can visit to choose their password

4th Step:

Go to “API permissions” and click on “Add a permission”. A screen will open on the right, select “Microsoft Graph” and “Delegated Permissions”. On the list that will show up, select the options “email”, “openid” and “profile”. Click on “Add permissions”.

The screenshot shows the Azure AD API permissions configuration interface. The left sidebar contains a navigation menu with 'API permissions' selected. The main area displays 'Configured permissions' for 'Microsoft Graph' with a table showing 'User.Read' permission. A 'Request API permissions' dialog is open on the right, showing the 'Microsoft Graph' API selected and the 'Delegated permissions' type chosen. The 'Select permissions' section lists 'email', 'openid', and 'profile' permissions, all of which are checked. The 'Add permissions' button is visible at the bottom of the dialog.

API / Permissions name	Type	Description	Admin consent required
Microsoft Graph (1)			
User.Read	Delegated	Sign in and read user profile	No

Permission	Admin consent required
<input checked="" type="checkbox"/> email View users' email address	No
<input type="checkbox"/> offline_access Maintain access to data you have given it access to	No
<input checked="" type="checkbox"/> openid Sign users in	No
<input checked="" type="checkbox"/> profile View users' basic profile	No

5th Step:

Go to “Overview”. First, copy and store the “Application (client ID)” (primeiro print). Then, on the top menu click on “Endpoints”. On the list that will show up (print 2), copy the URLs that appear on the options “OAuth 2.0 authorization endpoint (v2)” and “OAuth 2.0 token endpoint (v2)”. Copy and store the values of these URLs just like the password on the 2nd step and the application ID.

Got a second? We would love your feedback on Microsoft identity platform (previously Azure AD for developer). →

Essentials

Display name	: YapoliValidacao	Client credentials	: 0.certificate.1.secret
Application (client) ID	: 0968b9fe-6aee-408b-8980-f78e250bdd31	Redirect URIs	: 1.web.0.spa.0.public.client
Object ID	: f6481fc2-b583-41ef-bcd4-1e4578504b77	Application ID URI	: Add an Application ID URI
Directory (tenant) ID	: 999137ef-aafc-40b2-b879-2a9356095647	Managed application in L	: YapoliValidacao
Supported account types	: My organization only		

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Learn more](#)

[Get Started](#) [Documentation](#)

Endpoints

OAuth 2.0 authorization endpoint (v2)

<https://login.microsoftonline.com/999137ef-aafc-40b2-b879-2a9356095647/oauth2/v2.0/authorize>

OAuth 2.0 token endpoint (v2)

<https://login.microsoftonline.com/999137ef-aafc-40b2-b879-2a9356095647/oauth2/v2.0/token>

6th Step:

Go to the Yapoli platform, login on a profile with permission to administer clients. Go to Manager > Advanced Settings > Single Sign-On.

Click on + to add a new one and fill in the following fields as below (the omitted fields follow Yapoli's SSO documentation standard):

- Client ID: Enter the application ID copied on 5th step
- Scope: Enter openid email profile
- Authorization URL: enter the copied value from "OAuth 2.0 authorization endpoint (v2)" on 5th step
- Token URL: enter the copied value from "OAuth 2.0 token endpoint (v2)" on 5th step
- Secret Key: enter the copied password on 2nd step (attention: this password can no longer be obtained by either Azure or Yapoli after this operation ends)
- Type of request: : FORM
- Identification: ID_TOKEN
- Identification field: Enter: preferred_username

Click on SAVE after these steps are done and go to the Yapoli login page. An option with the SSO name inserted on the 6th step will be shown.

Obs: All these steps may be translated in other languages. Here, we considered the english configuration.