

How to include Microsoft Authentication in a Blazor Web Application

1. Create a Blazor Web App including Identity with Visual Studio 2022 Community Edition

2. Load the `Microsoft.AspNetCore.Authentication.MicrosoftAccount` with Nuget

3. Add this code in the application middleware (Program.cs)

```
builder.Services.AddAuthentication(options =>
{
    options.DefaultScheme = IdentityConstants.ApplicationScheme;
    options.DefaultSignInScheme = IdentityConstants.ExternalScheme;
})
.AddMicrosoftAccount(options=>
{
    options.ClientId = "XXXXXXXXXXXXXXXXXX";
    options.ClientSecret = "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";
})
.AddIdentityCookies();
```

4. Login in Azure and navigate to **App registrations** service

The screenshot shows the Microsoft Azure portal interface. The browser address bar displays `portal.azure.com/#view/Microsoft_AAD_RegisteredApps/ApplicationsListBlade`. The page title is "Microsoft Azure" with a search bar. The left sidebar contains navigation links: "Create a resource", "Home", "Dashboard", "All services", "FAVORITES", "All resources", "Resource groups", "Quickstart Center", "App Services", "Function App", "SQL databases", "Azure Cosmos DB", "Virtual machines", and "Load balancers". The main content area is titled "App registrations" and includes a "New registration" button. Below this, there is a notification about the transition from ADAL to MSAL. The "Owned applications" tab is selected, showing a list of applications. A search bar is present above the list. One application is listed: "terraform" with a client ID of "60c58c1a-11...".

Display name	Application ID
terraform	60c58c1a-11...

5. Application Registration -> New Registration

portal.azure.com/#view/Microsoft_AAD_RegisteredApps/CreateApplicationBlade/quickStartType~/null/isMSAApp~/false

Microsoft Azure

Home App registrations

Register an application

Supported account types

Who can use this application or access this API?

- ☐ Accounts in this organizational directory only (particular only - Single tenant)
- ☐ Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
- ☒ Accounts in any organizational directory (Any Microsoft Entra ID tenant - 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.

Web

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](#)

6. We select the option “Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)”

We also select the option **Redirect URI (optional)**->**Web**->

And we enter the Blazor web application URL: **https://localhost:7006/signin-microsoft**

We press the **Register** button

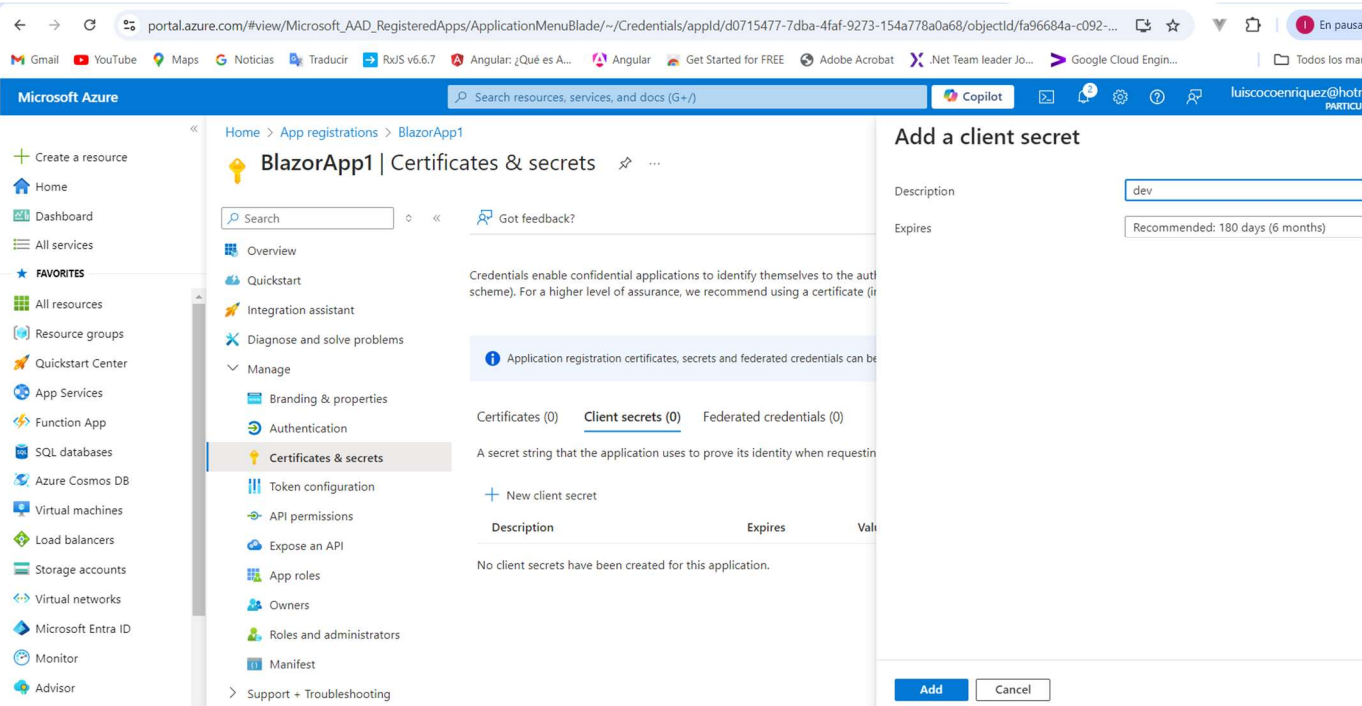
8. We press in the link “Add a certificate or secret”

The screenshot shows the Microsoft Azure portal interface. The left sidebar contains navigation options like 'Create a resource', 'Home', 'Dashboard', 'All services', and 'FAVORITES'. The main content area displays the 'BlazorApp1' application registration overview. The 'Overview' tab is selected, showing details such as 'Display name: BlazorApp1', 'Application (client) ID: d0715477-7dba-4faf-9273-154a778a0a68', 'Object ID: fa96684a-c092-4ab0-abdf-d9ce723d5737', 'Directory (tenant) ID: e099cebd-5eea-41a3-88db-bcb9a9c8a83e', and 'Supported account types: All Microsoft account users'. The 'Client credentials' section is highlighted with a red box, and the link 'Add a certificate or secret' is also highlighted with a red box.

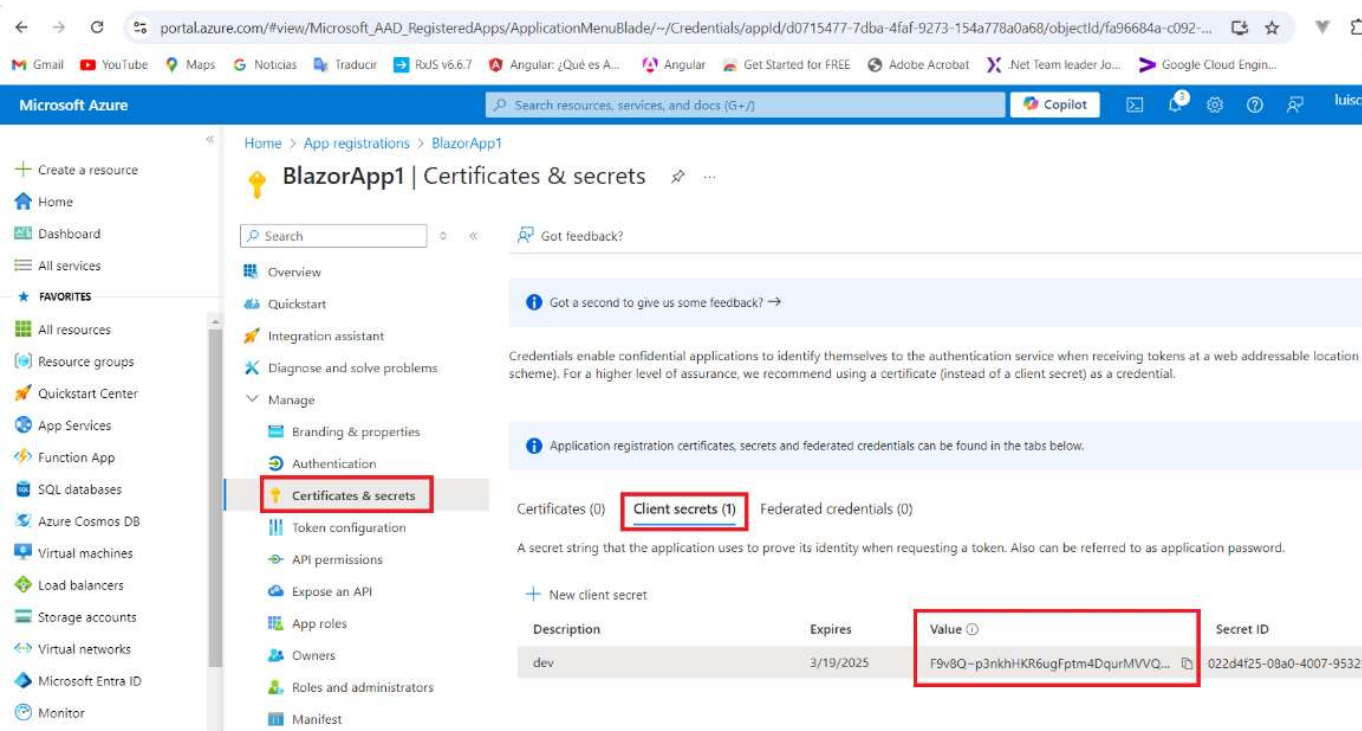
9. In the Client secrets tab we press on the “New client secret”

The screenshot shows the Microsoft Azure portal interface. The left sidebar contains navigation options like 'Create a resource', 'Home', 'Dashboard', 'All services', and 'FAVORITES'. The main content area displays the 'BlazorApp1' application registration 'Certificates & secrets' page. The 'Client secrets (0)' tab is highlighted with a red box, and the 'New client secret' button is also highlighted with a red box. The page shows a table with columns 'Description', 'Expires', 'Value', and 'Secret ID'. The table is currently empty, and a message states 'No client secrets have been created for this application.'

10. We set the new secret name, for example: **dev**



Secret: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

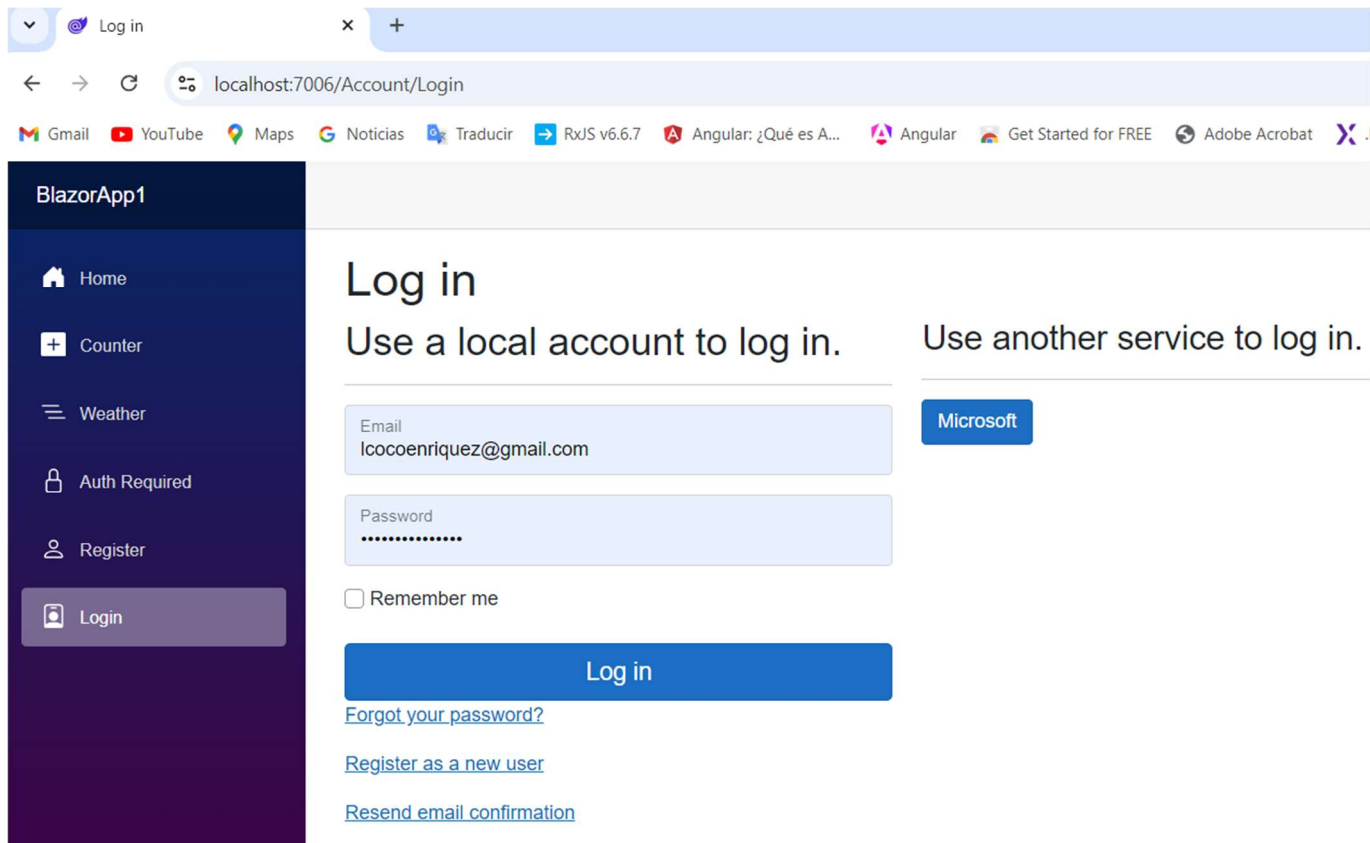


11. Update in the middleware (Program.cs) the `ClientSecret` value with the value copied from

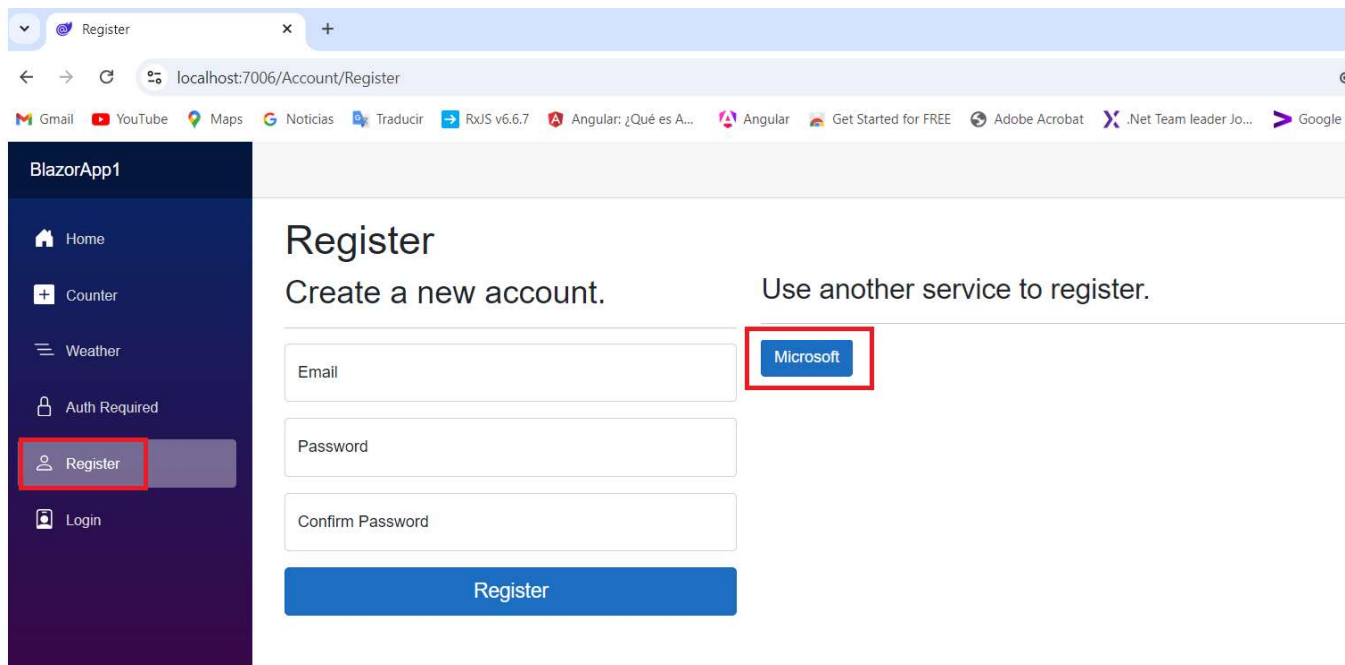
```
builder.Services.AddAuthentication(options =>
{
    options.DefaultScheme = IdentityConstants.ApplicationScheme;
    options.DefaultSignInScheme = IdentityConstants.ExternalScheme;
})
.AddMicrosoftAccount(options=>
{
    options.ClientId = "d0715477-7dba-4faf-9273-154a778a0a68";
    options.ClientSecret = "F9v8Q~p3nkhHKR6ugFptXXXXXXXXXXXXXXXXXXXX " ;
})
.AddIdentityCookies();
```

12. Now we run the Blazor Web App and we navigate to the Register and Login pages

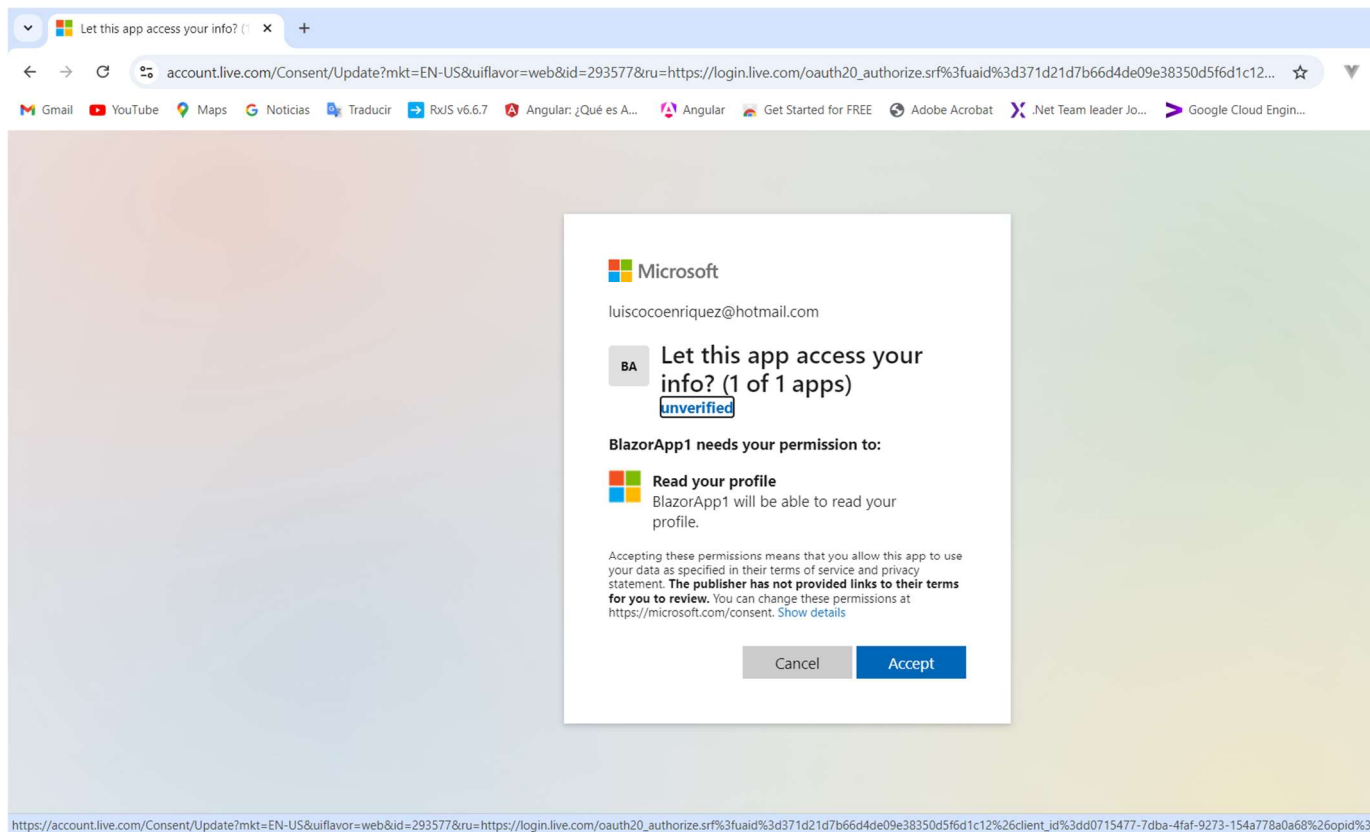
The screenshot shows a web browser window with the address bar displaying 'localhost:7006/Account/Register'. The browser's address bar and tabs are visible at the top. Below the browser window, the application's interface is shown. On the left, there is a dark blue sidebar with the title 'BlazorApp1' and a list of navigation items: 'Home', 'Counter', 'Weather', 'Auth Required', 'Register' (which is highlighted), and 'Login'. The main content area has a light gray background. It features a large heading 'Register' and a subheading 'Create a new account.'. To the right of this, there is a link 'Use another service to register.' with a 'Microsoft' button next to it. Below the subheading, there are three input fields labeled 'Email', 'Password', and 'Confirm Password'. At the bottom of the form is a large blue button labeled 'Register'.



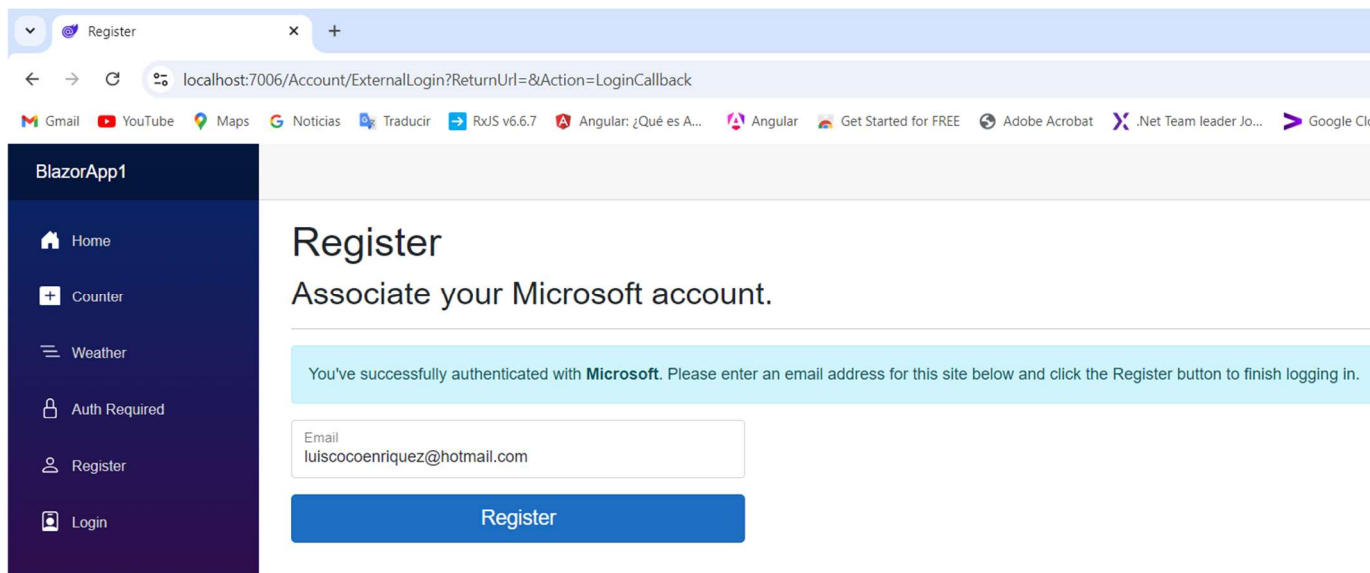
13. We press in the Microsoft button for Registering a new user in the application



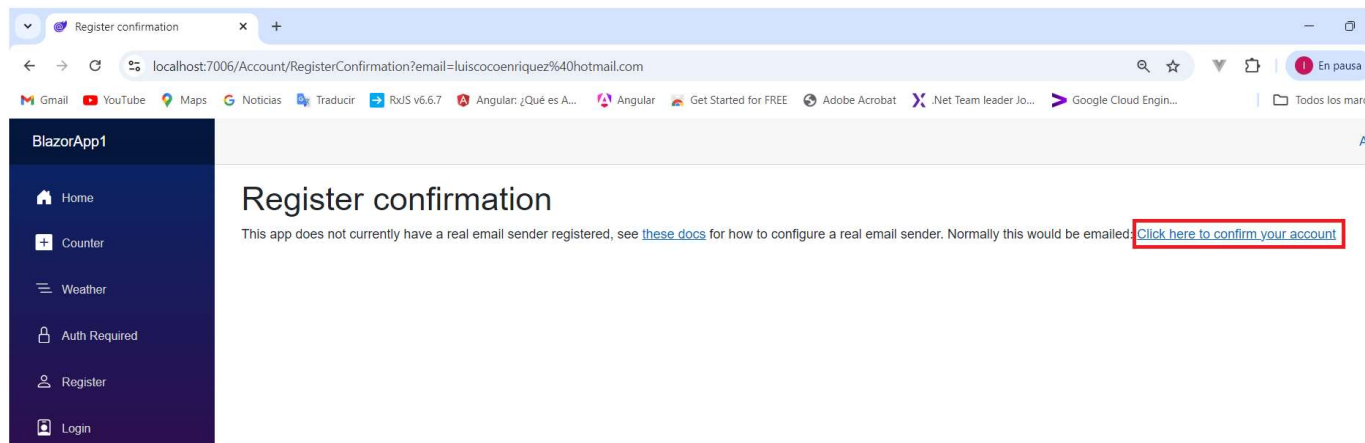
We are redirected to the **Microsoft Authentication process**, we enter our Microsoft Account username: luiscocoenriquez@hotmail.com and the password



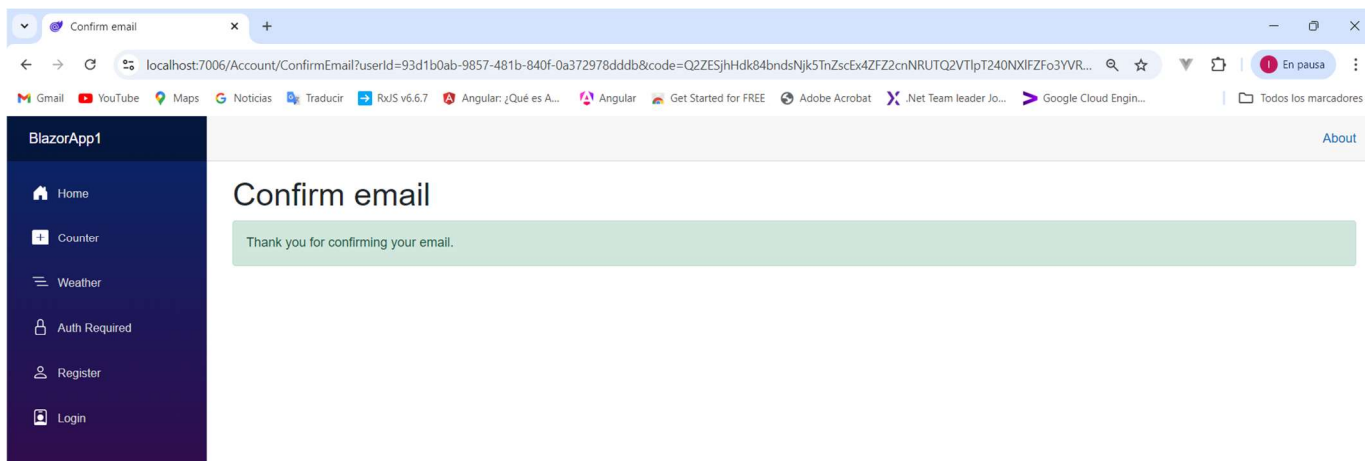
14. After the Microsoft authentication is successfully, we press in the Register button for creating the new account.



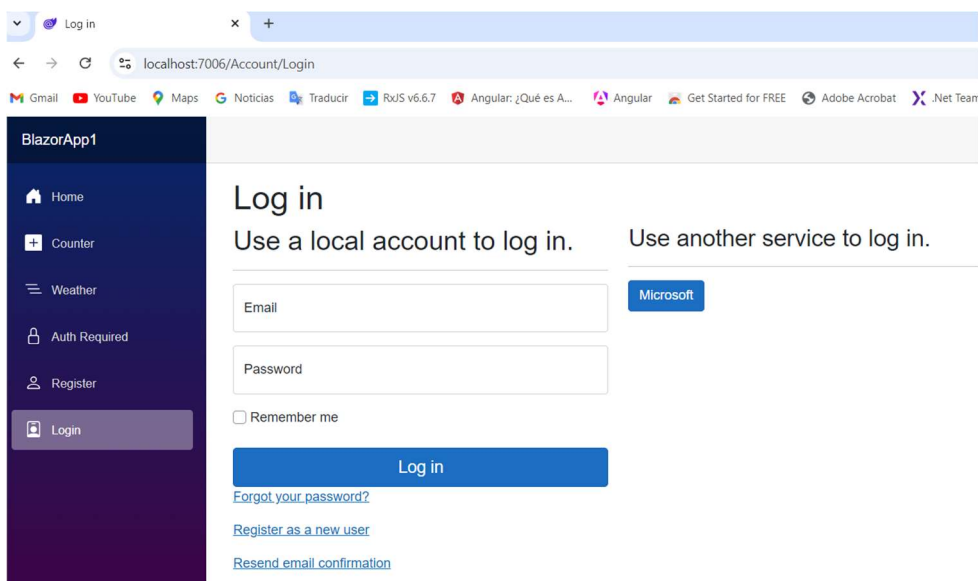
15. We press in the link “Click here to confirm your account” to confirm the new account



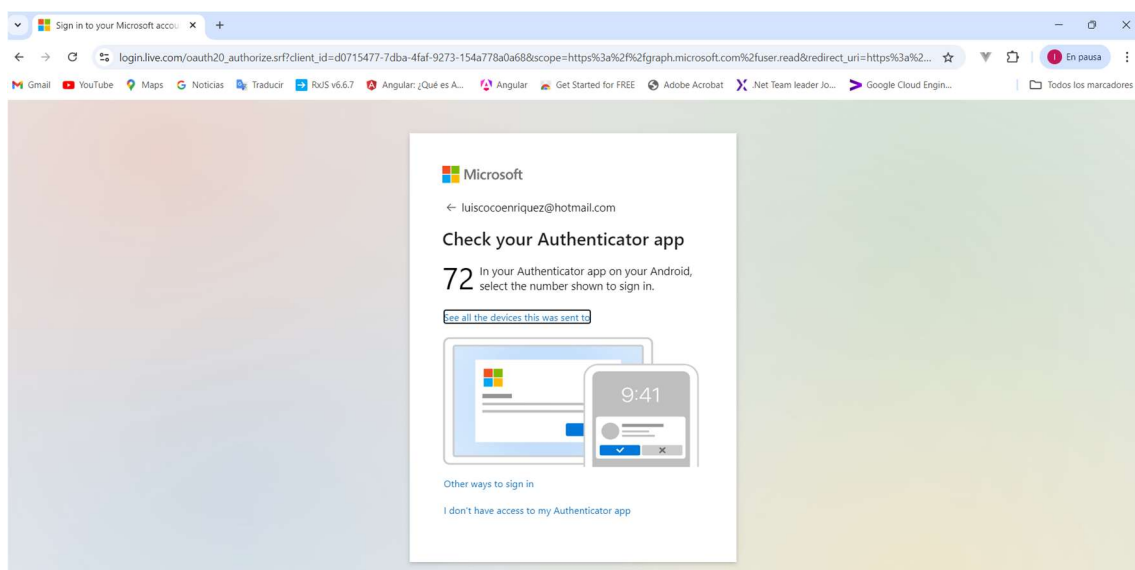
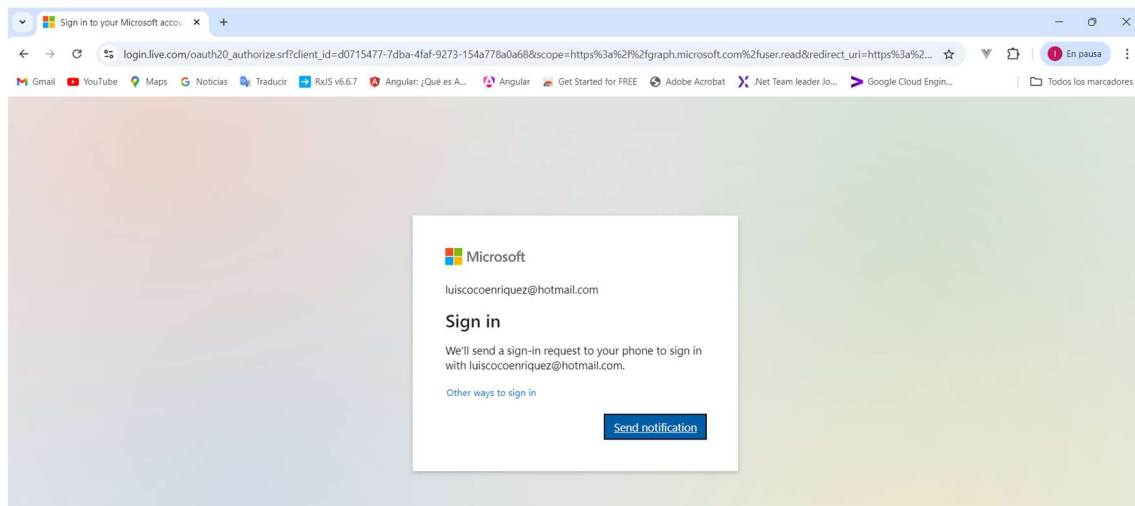
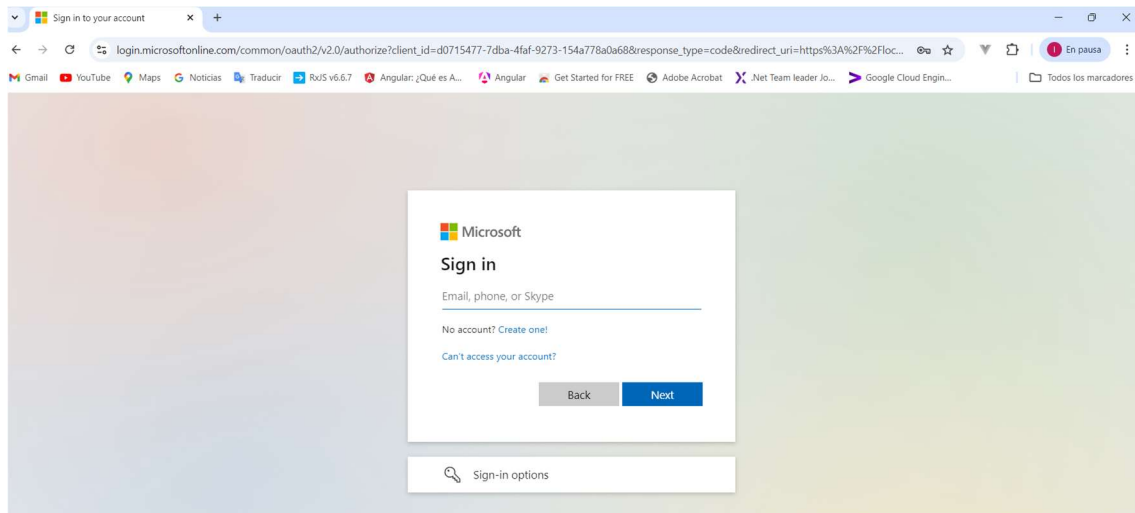
16. We get the confirmation email message



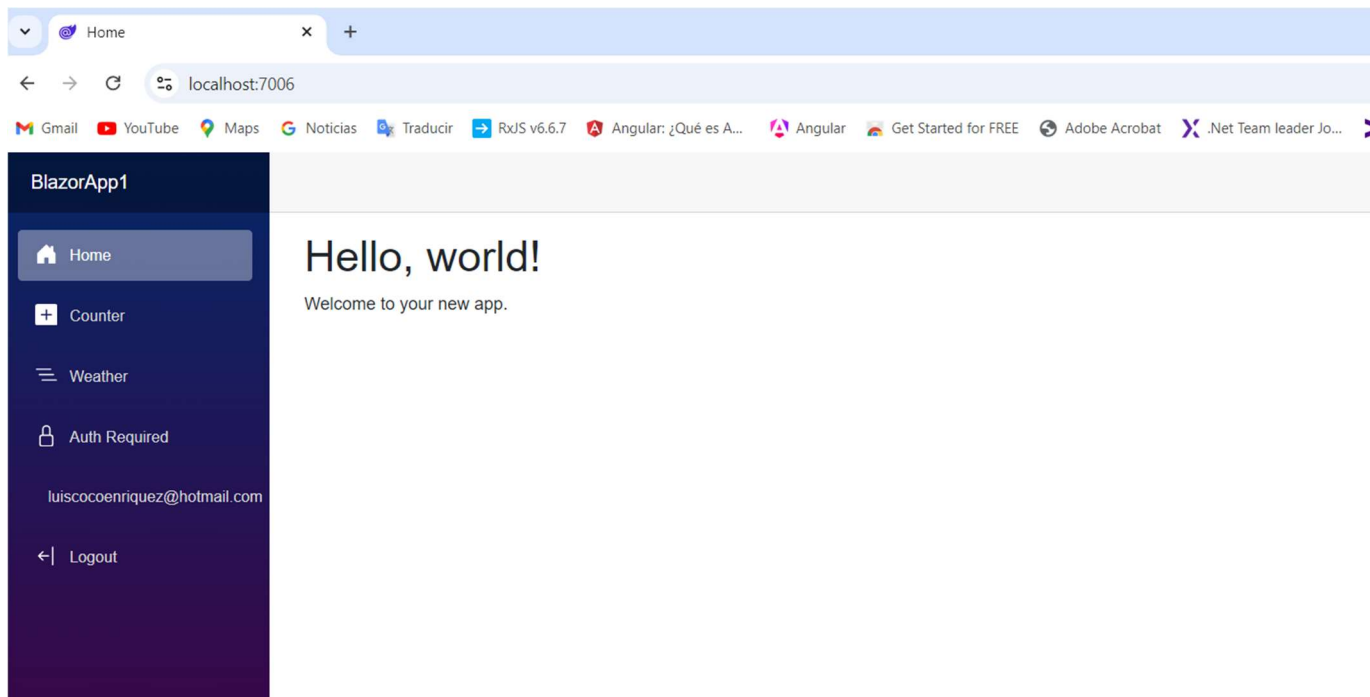
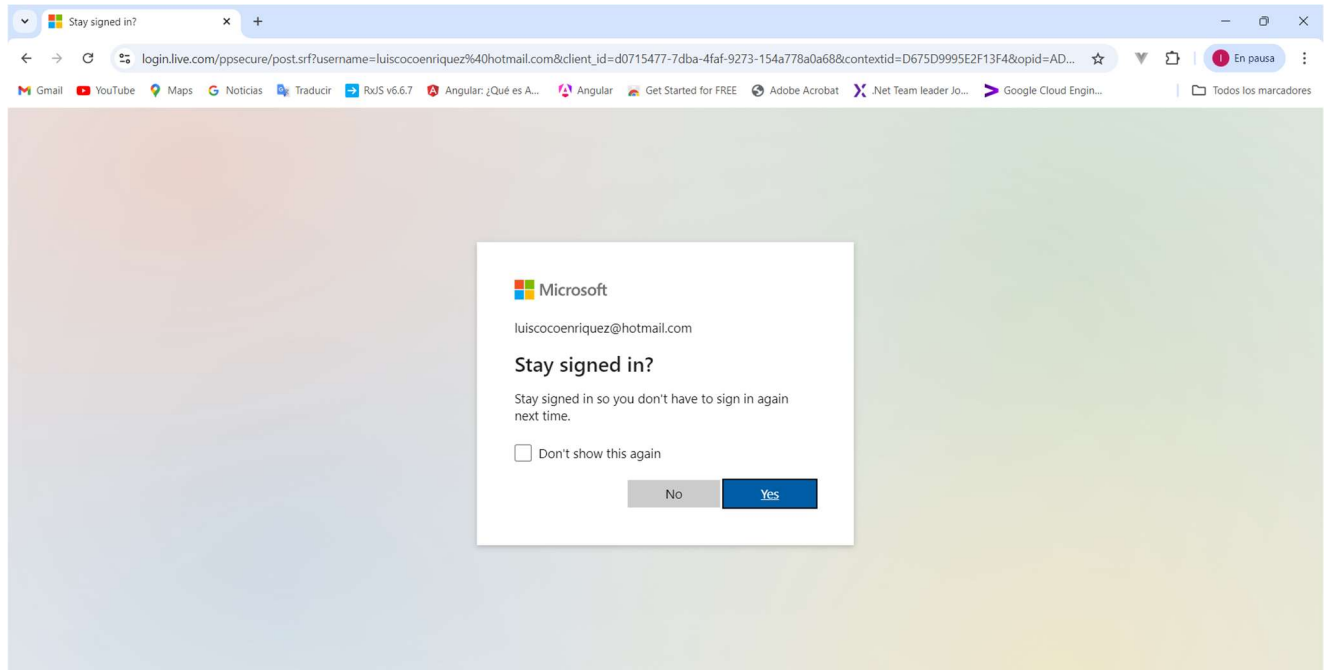
17. For accessing the application with the Microsoft Login, we press in the Microsoft button



18. After pressing the Microsoft button in the Login page we are asked to enter the username and password for Microsoft Authentication



We open Microsoft Authenticator application in our Mobile Phone and then press the 72 value and approve button for accessing our Blazor Web App.



19. We open the personal data panel and confirm the External Authentication with Microsoft

