

Configuring CORS

(Demo prep)

-
-

Configuring the Identity microservice for CORS

It is time to try out our updated front-end portal with the microservices we created and secured in this course. If you have not done it already, please grab the source code for the front-end as indicated in the previous lesson. And also remember that you will need Node.JS in your box to be able to run the application.

As you can see, I have already downloaded and extracted the frontend source code under Play.Frontend in my projects directory. So let's switch to Play.Frontend...

1. Open VS Code terminal and switch to d:\projects
2. Switch to Play.Frontend and:

`code . -r`
3. "This is a revised version of the frontend single page application used in previous modules, so it has several modifications and additions to account for identity and authorization related scenarios. We will explore what has changed later, but for now let's start by installing all the dependencies..."
4. Install the dependencies:
`npm install`
5. Start the frontend:
`npm start`
6. Start the Identity microservice
7. Start the front-end
8. Browse to front-end
9. Notice Catalog and Inventory sections not there
10. Try Login
11. Show CORS error in Broser console

In Play.Identity

12. Update appsettings.Development.json

```
{
  "Logging": {
    ...
  },
  "AllowedOrigin": "http://localhost:3000"
}
```

13. Add AllowedOriginSetting constant to Startup:

```
public class Startup
{
    private const string AllowedOriginSetting = "AllowedOrigin";
    ...
}
```

14. Update Configure method in Startup:

```
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
    if (env.IsDevelopment())
    {
        ...
        app.UseSwaggerUI(c => c.SwaggerEndpoint("/swagger/v1/swagger.json", "Play.Catalog.Service v1"));

        app.UseCors(builder =>
        {
            builder.WithOrigins(Configuration[AllowedOriginSetting])
                .AllowAnyHeader()
                .AllowAnyMethod();
        });
    }
    ...
}
```

15. Run Identity microservice and try login again

16. Notice unknown client error in Identity console

In the next lesson we will add the proper configuration to authorize our front-end client to access our microservices REST APIs.