

Trying out the updated Frontend portal

Script start

It's time to explore the new store experience that is included as part of the updated frontend portal. You will find all the files for this new version of the frontend in the frontend zip file attached to this lesson.

So, if you still have the old frontend files in your box, please delete them and then download the zip file and extract it on the same location where you had the older files.

In my case I have already done that and I have my new frontend files in my Play.Frontend directory you can see here. So I'll open my VS Code instance in that directory.

In Frontend repo

1. Run npm install
2. Talk briefly about:
 - src\App.js
 - src\components\Home.js
 - src\components\NavMenu.js
 - src\components\Store.js
 - src\components\form\PurchaseForm.js
 - src\components\form\PurchaseModal.js
 - (optional) src\components\constants.js
 - public\config.js
 - src\components\api-authorization\AuthorizeService.js (trading.fullaccess in ensureUserManagerInitialized)

In Identity repo

3. Update appsettings.Development.json:

```
"IdentityServerSettings": {  
  "Clients": [  
    {  
      "ClientId": "frontend",  
      ...  
      "AllowedScopes": [  
        ...  
        "inventory.fullaccess",  
      ]  
    }  
  ]  
}
```

```

    "trading.fullaccess",
    "IdentityServerApi",
    ...
  ],
  ...
},
...
]
}

```

In Trading repo

4. Update appsettings.Development.json:

```

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

```

5. Update Startup:

```

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

    ...

    // This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
    public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
    {
        if (env.IsDevelopment())
        {
            ...
            app.UseSwaggerUI(c => c.SwaggerEndpoint("/swagger/v1/swagger.json", "Play.Trading.Service
v1"));

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

```

```
    });  
  }  
  ...  
}  
...  
}
```

In Frontend repo

6. Run npm start
7. Start all microservices
8. Browse to frontend at <http://localhost:3000>
9. Login as player1
10. Go to the Store section
11. Do a Purchase
12. Notice purchase dialog does not go away
13. Notice the browser console errors
14. Notice after a refresh the purchase did complete

In the next lesson we will learn about SignalR and how it can provide real time updates into our frontend.