**Demo of the whole project can be accessed here** [**https://commoditydashboardui.azurewebsites.net/**](https://commoditydashboardui.azurewebsites.net/)

**Dashboard API**

After building the API project successfully. Please access <http://localhost:50000/swagger> to see the screen below. It has been deployed to the Azure cloud and can be access here at <https://commoditydashboard.azurewebsites.net/swagger/index.html>

Graphical user interface, application

Description automatically generated

**API Highlights:**

1. **It is built with latest and greatest framework and Nuget packages ASP.NET Core 6.0.5**
2. **It uses Clean Architecture with CQRS pattern.**
3. **It uses Microstoft IdentityServer and JWT token for Authorization.**
4. **It uses lates EF Core with Code First approach. Seed Data is populated from the Excel sheet.**
5. **It uses Automapper and Repository patten for extra level of abstraction**
6. **There are three Test projects to cover the logic at different levels with two of the integration tests use InMemory DB.**

**Desired to do the following but could not due to time constraint**

1. **Handle Excptions**
2. **Add Validators to Query and Commands using CQRS**
3. **Add Logging**
4. **Deployment to Azure and Much more.**

**UI highlights:**

**ng serve will render the following UI, Please press the login button on the top right to see the dashboard**

Graphical user interface

Description automatically generated with low confidence

Graphical user interface, application

Description automatically generated

1. **It is using latest Angular 13.3.8**
2. **It uses Angular material and ng2-charts**
3. **It uses authentication module**
4. **It was designed to be http call savvy.**
5. **There are component and service tests**

**Desired to do the following but could not due to time constraint**

1. **Improvement in the UI.**
2. **Testing coverage**
3. **E2E tests in Cypress**
4. **Use of AuthInterceptor and common APIservice with error handling**
5. **Lazy loaded Modules**