Digital Retailers Inc. (Phase 3 Assessment)

Frameworks and APIs used:

ASP.NET CORE MVC with Entity Framework Core and Identity Api.

Database Schema:

Diagram

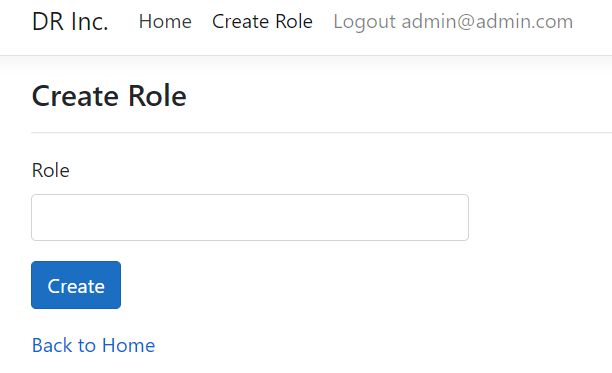
Description automatically generated with medium confidence

Since I used Identity API, we can see tables like AspNetUsers, Roles, UserRoles. I added one more table called Laptops which is referring to the primary key of the AspNetUsers by Userid.

That way it became easy to grab the details of the seller.

I have created an Administration Controller which can add Roles to the AspNetRoles. There is no registration for the Admin as specified by the Instructor. I have given only that feature for the Admin and the figure for that is given below.

Navbar changes accordingly depending on the Role of the User which you will see on the upcoming figures.



With the help of Identity Api I created two Roles called Seller and Customer. So users can Register themselves accordingly.

Graphical user interface, application

Description automatically generated

By default, any User will land upon the Login Page as shown in the below figure.

Graphical user interface, text, application

Description automatically generated

Based on the User type they will be redirected to the respective dashboard. Let us see about the Seller dashboard first. In seller controller I implemented two features called Add Laptop and Sales Report. Add Laptop looks like below.

Graphical user interface, application

Description automatically generated

Sales Report looks like below.

Chart, scatter chart

Description automatically generated

Seller can see the No. of Sales a laptop got and also can perform CRUD operations on the Laptops he has added.

Let us now look at Customer Dashboard.

Graphical user interface

Description automatically generated

This is how the Customer Index Page Looks like.

They can browse through all the laptops. To know more about a particular laptop, they can press “Details” button.

Once they press the button the below figure demonstrates how the Details page looks like.

Graphical user interface

Description automatically generated

They can navigate to Index Page using “Back” Button. If they want to proceed to buy, they can press “Buy Now”. They will be redirected to Payments page. The page looks like below.

Graphical user interface, text, application

Description automatically generated

There they can enter the card details and proceed to “Submit”. This will redirect to Order Page, where they can the see the order summary. The page looks like below.

Graphical user interface, text, application, email

Description automatically generated

I am Authorizing everything with the help of Microsoft.AspNetCore.Authorization. Seller can’t access customer controller and vice versa.

This can be done by using

[Authorize(Roles = "Customer")] and [Authorize(Roles = "Seller")].

If they still try to access, they will get this page as a result.

Graphical user interface, application

Description automatically generated

Since I am using Authorization, without Logging in if anyone tries to enter into a controller using url, they will be redirected to the Login Action. That way we can ensure security.

THE END