Week 10 Assignment: Programming Lab – .NET Core for

Master of Science

Information & Communications Technology

Nicholas Smith

University of Denver University College

March 12th, 2023

Faculty: Kris Jamsa

Director: Cathie Wilson

Dean: Michael J. McGuire, MLS

Table of Contents

Table of Figures

Definitions

ASP.NET – Active Server Pages within the .NET framework

HTML – Hypertext Markup Language

LINQ – Language-Integrated Query

XML – Extensible Markup Language

Overview

The purpose of this lab assignment is to create a .NET core application with identity management to display specific content based on a user role. In addition to this the application will require the following:

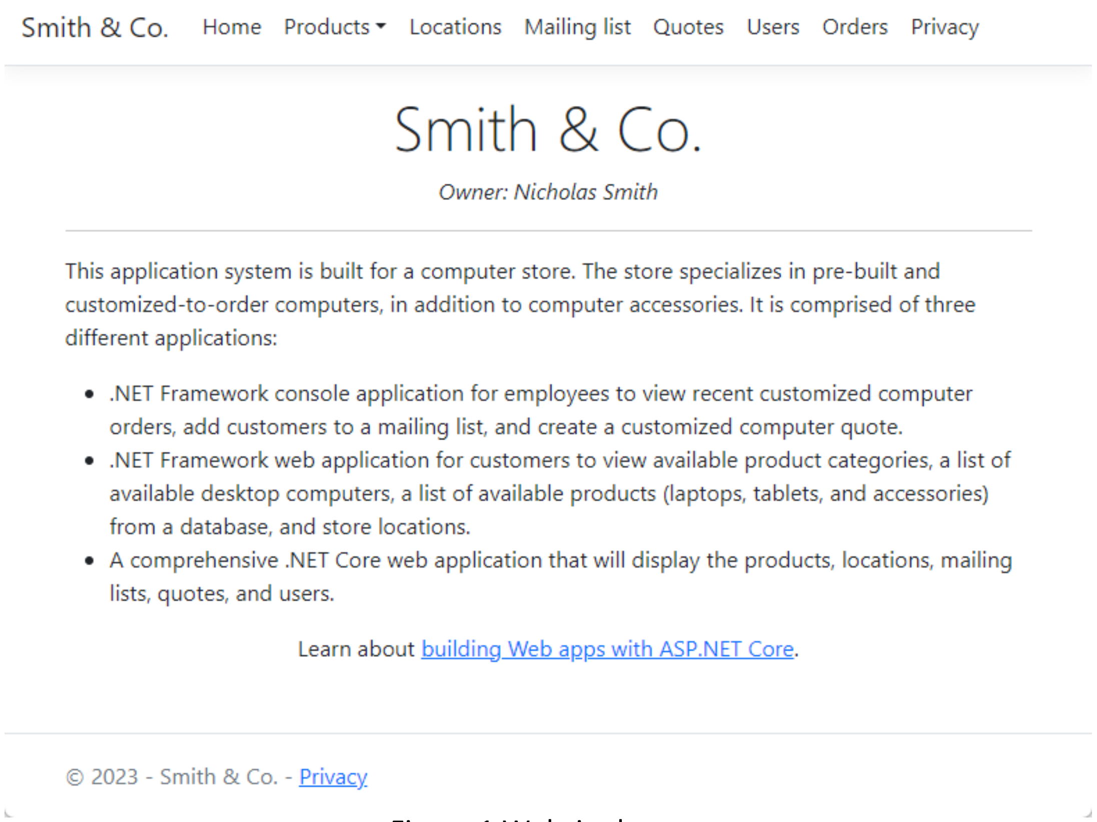
Using the same solution, create a CORE application that meets the following requirements:

- Create a Core Web Application and a login form on the main page.
- Implement user authentication, so only certain users will be able to access the following pages that share a common menu at the top of the page.
- Create a page called products that lists all the products, grouped by the type field, from the SQLite products table that was used in week 4.
- Create a page called locations that displays all the stores from the XML file used in week 5.
- Create a page called mailing lists that loops through the files in the mailing list directory and outputs the users information used in week 6.
- Create a page called quotes that provides a downloadable list of all the quotes generated in week 8.
- Create a page called users that displays the user's names from the users.xml file generated in week 9.

Scenario

This application system is built for a computer store. The store specializes in pre-built and customized-to-order computers, in addition to computer accessories. It is comprised of three different applications:

- .NET Framework console application for employees to view recent customized computer orders, add customers to a mailing list, and create a customized computer quote.
- .NET Framework web application for customers to view available product categories, a list of available desktop computers, a list of available products (laptops, tablets, and accessories) from a database, and store locations.
- A comprehensive .NET Core web application that will display the products, locations, mailing lists, quotes, and users.



Smith & Co.

Owner: Nicholas Smith

This application system is built for a computer store. The store specializes in pre-built and customized-to-order computers, in addition to computer accessories. It is comprised of three different applications:

- .NET Framework console application for employees to view recent customized computer orders, add customers to a mailing list, and create a customized computer quote.
- .NET Framework web application for customers to view available product categories, a list of available desktop computers, a list of available products (laptops, tablets, and accessories) from a database, and store locations.
- A comprehensive .NET Core web application that will display the products, locations, mailing lists, quotes, and users.

Learn about building Web apps with ASP.NET Core.

© 2023 - Smith & Co. - Privacy

Figure 1 Website homepage.

Application Mock-up

This application is designed with ASP .NET 6.0 for the website, .NET Framework 4.8 for the console application, and has a shared class library for the computer store resources that is targeted for .NET Standard 2.0. The files generated by both the website and console application can be viewed by each other. As new files are generated they are able to be loaded by the respective applications independently as requested by the applications. Utilizing a shared component method of programming this application ensures that standards are consistent throughout the applications processing, and reduce development time by centralizing efforts for feature updates and bug fixes.

It is also notable to mention that this mock-up was developed using Visual Studio 2022 (version 17.5.1) with the workloads “ASP.NET and web development”, “.NET desktop development”, and “Office/SharePoint development”. Git which is a repository management software was also used to document and track changes during the development process.

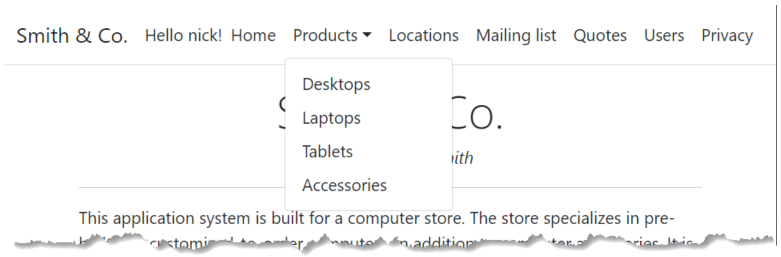
ASP.NET Core Web Application

Site Framework

The project version used for the website development was “ASP.NET Core Web App (Model-View-Controller). This allows for Razor pages to be used for generating the HTML that will be sent to the client computers.

The user whether customer or employee accessing the computer store’s website will see the splash screen homepage as shown in Figure 1.

At the top is a menu that is present on all pages, see Figure 2. This contains a products dropdown to group product related pages together (i.e desktops, laptops, tablets, and accessories). The .NET Framework pages are laptops, tablets, accessories. The .NET Core pages are locations, mailing list, quotes, and users. These were combined into the ASP.NET Framework to all be in the .NET Core website.

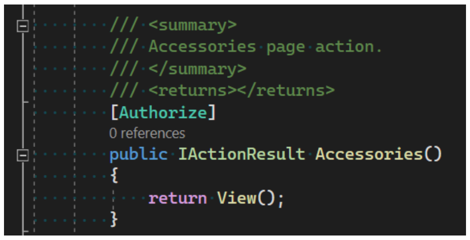


This application system is built for a computer store. The store specializes in pre-

Figure 2 Website menu – products expanded.

Authorized Access

There are authorized only pages set in this application utilizing the authorize attribute made available via the library Microsoft.AspNetCore.Authorization. Within the controllers class each action result type method ties together the model and view. This is the area that applying the "[Authorize]" attribute is required. This attribute also allows for policy and role properties to be set to further restrict who can access the page. For this mock-up there is a public page setup and a private without additional roles. To enable this feature it requires development of either an XML helper class or simply adding an SQL database.



```
/// <summary>
/// Accessories page action.
/// </summary>
/// <returns></returns>
[Authorize]
0 references
public IActionResult Accessories()
{
    return View();
}
```

Figure 3 Example use of the authorize attribute.

The authorized only restriction was added to the following pages: Accessories, Download, Laptops, Locations, Orders, MailingList, Products, Quotes, Tablets and Users. This left the index and policy page open to the public while still being within the home controller.

For the user to access these pages the application redirects the user with a authentication service to the /Account/Login page, see Figure 4. This page will not go away until the user has successfully authenticated. If they enter incorrect information the website will prompt the user so they can correct their mistake, see Figure 5.

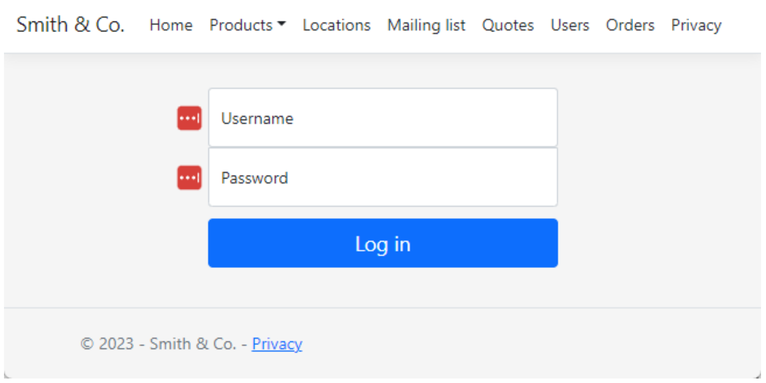
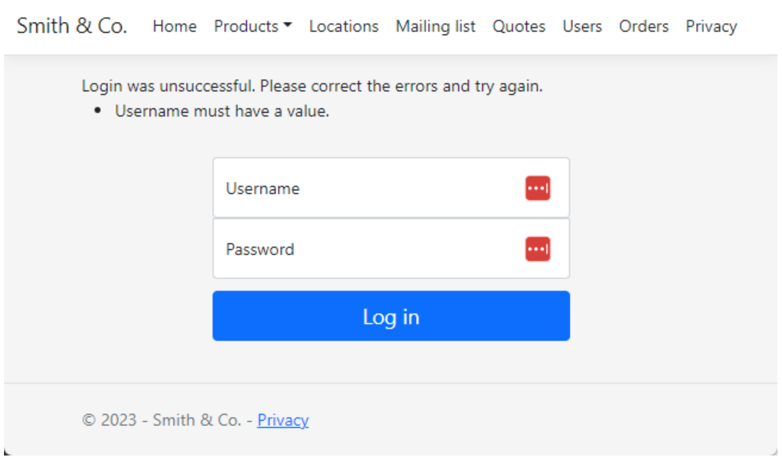


Figure 4 Login page.



Login was unsuccessful. Please correct the errors and try again.

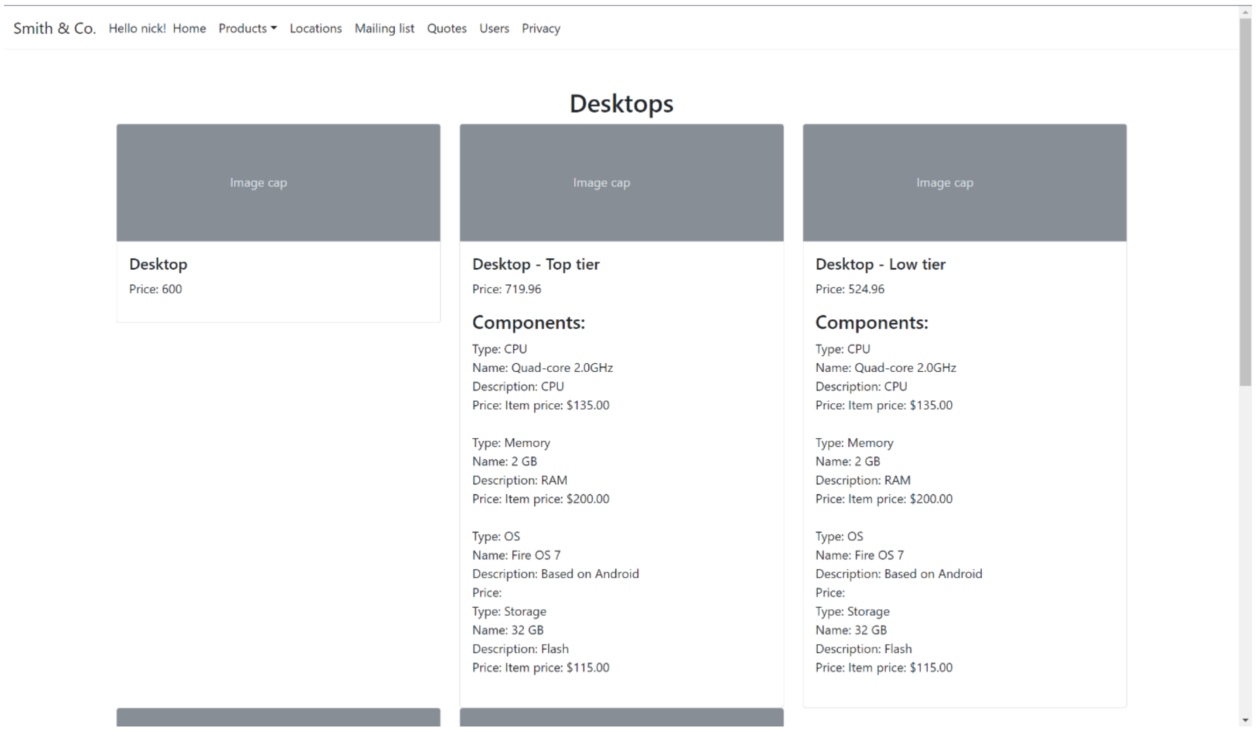
- Username must have a value.

Figure 5 Login page, prompt error.

Products

Desktops

The desktops page displays all the desktops that can be ordered.



Desktops

Desktop
Price: 600

Desktop - Top tier
Price: 719.96

Components:
Type: CPU
Name: Quad-core 2.0GHz
Description: CPU
Price: Item price: $135.00

Type: Memory
Name: 2 GB
Description: RAM
Price: Item price: $200.00

Type: OS
Name: Fire OS 7
Description: Based on Android
Price:
Type: Storage
Name: 32 GB
Description: Flash
Price: Item price: $115.00

Desktop - Low tier
Price: 524.96

Components:
Type: CPU
Name: Quad-core 2.0GHz
Description: CPU
Price: Item price: $135.00

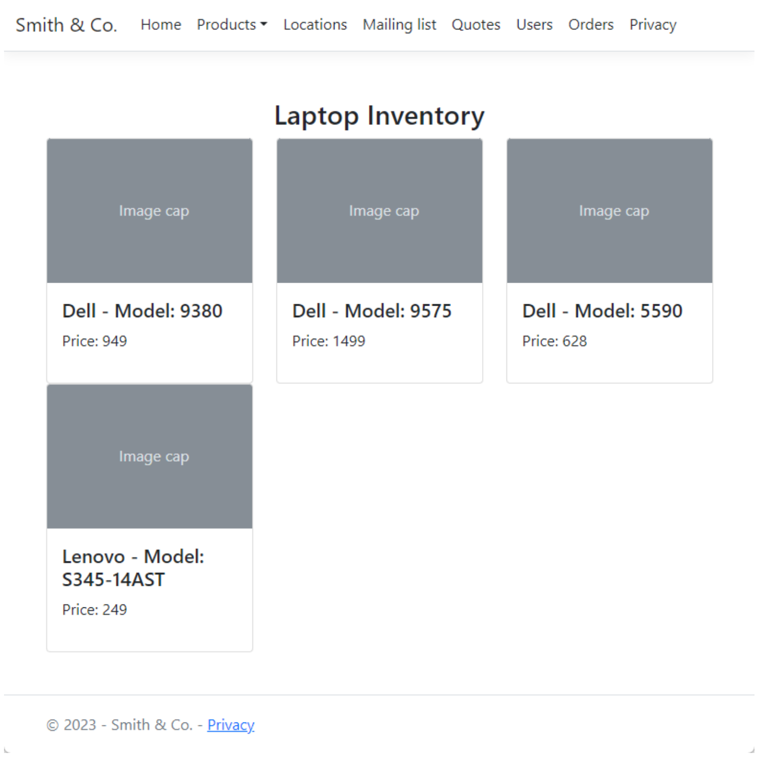
Type: Memory
Name: 2 GB
Description: RAM
Price: Item price: $200.00

Type: OS
Name: Fire OS 7
Description: Based on Android
Price:
Type: Storage
Name: 32 GB
Description: Flash
Price: Item price: $115.00

Figure 6 Desktops page.

Laptops

The laptops page shows all laptop products from an SQL file.



Laptop Inventory

Dell - Model: 9380
Price: 949

Dell - Model: 9575
Price: 1499

Dell - Model: 5590
Price: 628

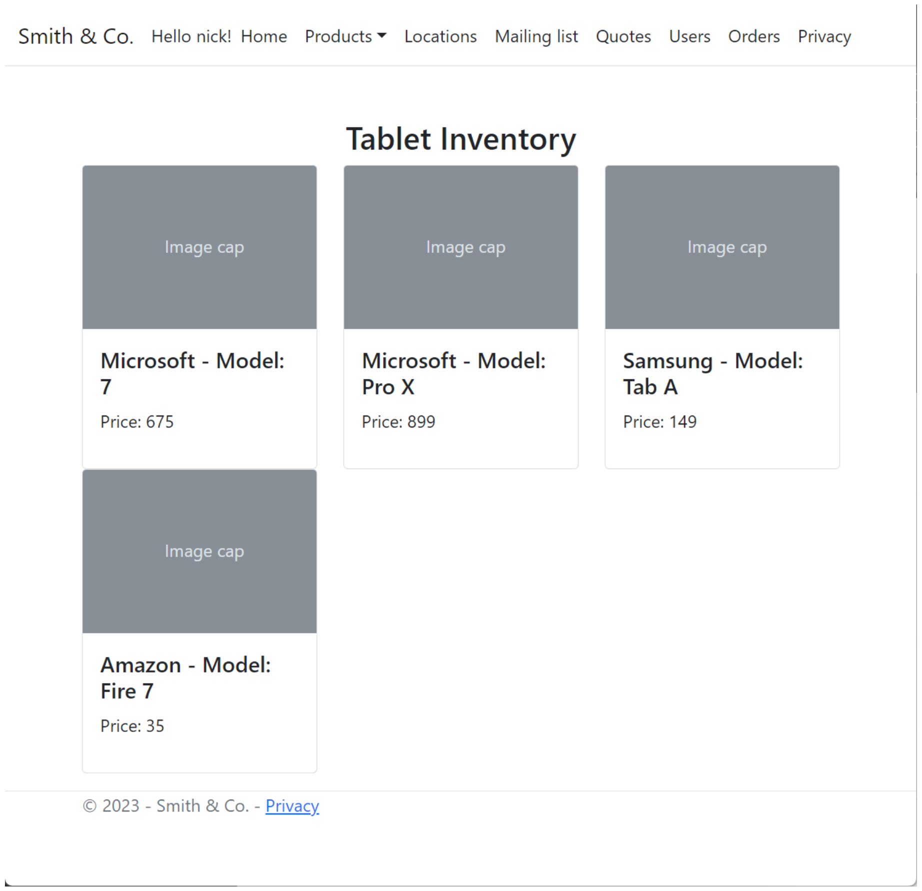
Lenovo - Model: S345-14AST
Price: 249

© 2023 - Smith & Co. - Privacy

Figure 7 Laptop inventory page.

Tablets

The tablet page shows all tablet products from an SQL file.



Tablet Inventory

Microsoft - Model: 7
Price: 675

Microsoft - Model: Pro X
Price: 899

Samsung - Model: Tab A
Price: 149

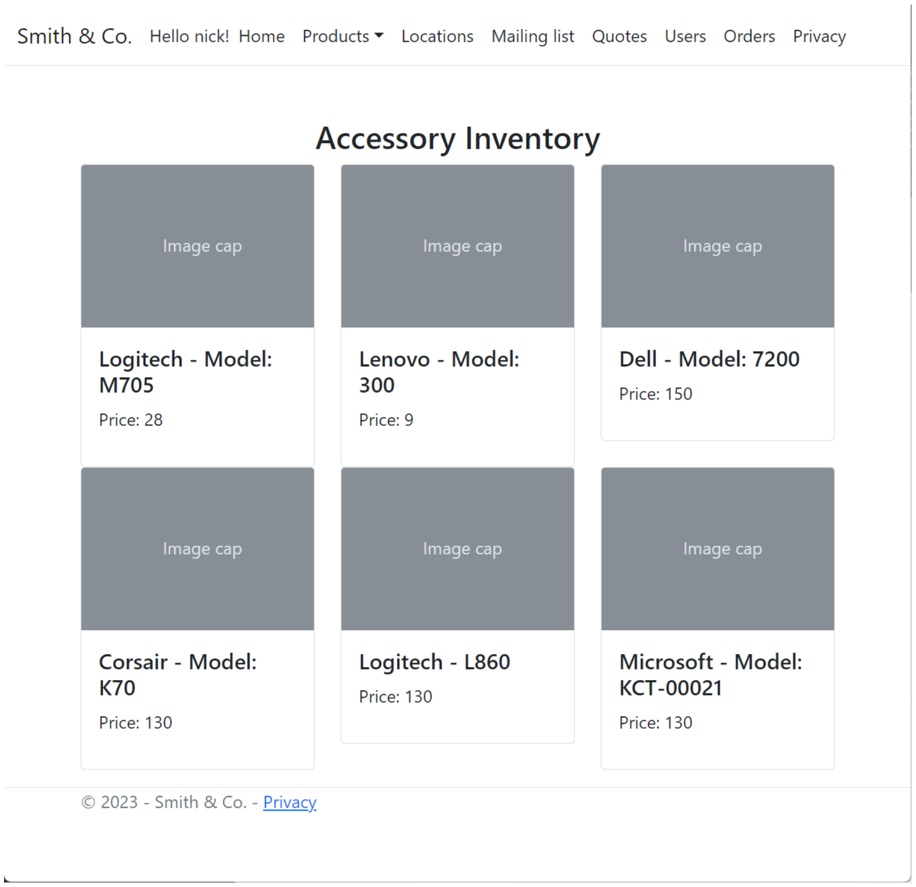
Amazon - Model: Fire 7
Price: 35

© 2023 - Smith & Co. - Privacy

Figure 8 Tablet inventory page.

Accessories

The accessories page shows all accessory products from an SQL file.



Accessory Inventory

Logitech - Model: M705
Price: 28

Lenovo - Model: 300
Price: 9

Dell - Model: 7200
Price: 150

Corsair - Model: K70
Price: 130

Logitech - L860
Price: 130

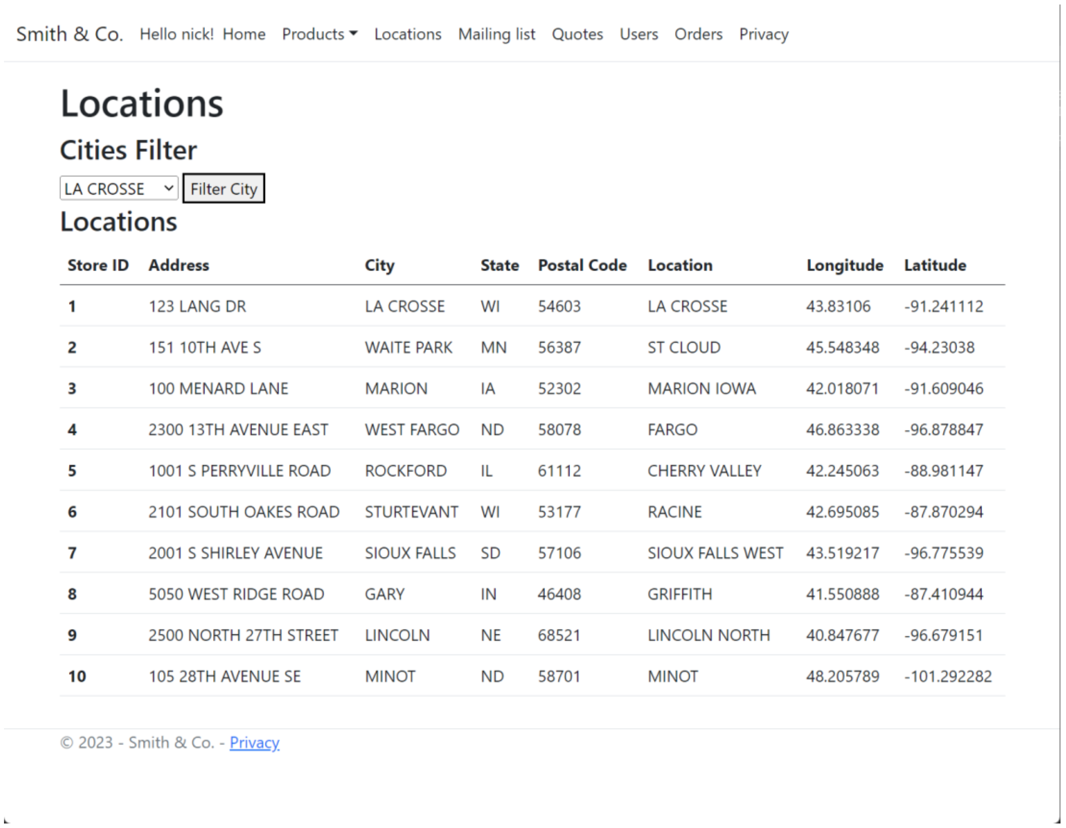
Microsoft - Model: KCT-00021
Price: 130

© 2023 - Smith & Co. - Privacy

Figure 9 Accessory inventory page.

Locations

The locations page is generated from an XML file called “store_locations.xml”. Razor code reads an array set in the program to generate the HTML code. Javascript is then used on the client side to allow for filtering the list by the city. This will show only locations with the specified city name. The mock-up of this page is as shown in Figure 10.



Locations

Cities Filter

Locations

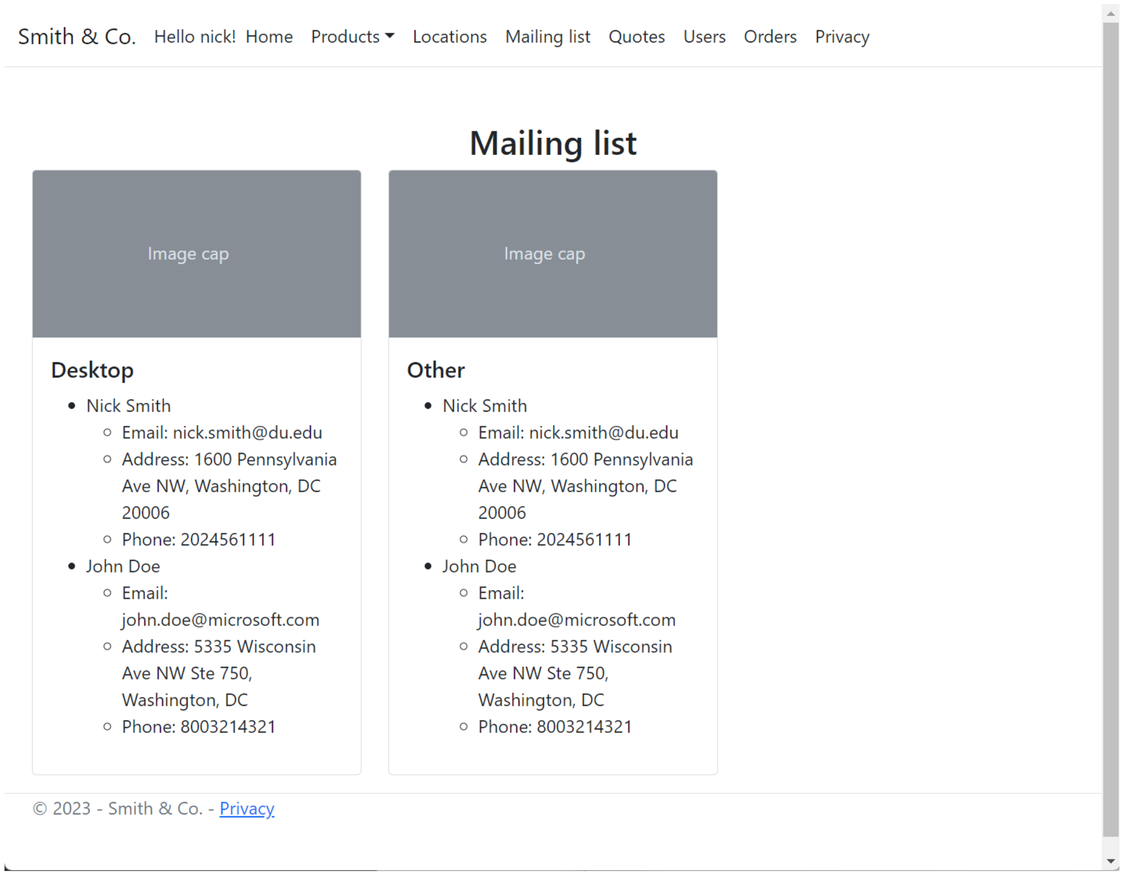
Store ID	Address	City	State	Postal Code	Location	Longitude	Latitude
1	123 LANG DR	LA CROSSE	WI	54603	LA CROSSE	43.83106	-91.241112
2	151 10TH AVE S	WAITE PARK	MN	56387	ST CLOUD	45.548348	-94.23038
3	100 MENARD LANE	MARION	IA	52302	MARION IOWA	42.018071	-91.609046
4	2300 13TH AVENUE EAST	WEST FARGO	ND	58078	FARGO	46.863338	-96.878847
5	1001 S PERRYVILLE ROAD	ROCKFORD	IL	61112	CHERRY VALLEY	42.245063	-88.981147
6	2101 SOUTH OAKES ROAD	STURTEVANT	WI	53177	RACINE	42.695085	-87.870294
7	2001 S SHIRLEY AVENUE	SIOUX FALLS	SD	57106	SIOUX FALLS WEST	43.519217	-96.775539
8	5050 WEST RIDGE ROAD	GARY	IN	46408	GRIFFITH	41.550888	-87.410944
9	2500 NORTH 27TH STREET	LINCOLN	NE	68521	LINCOLN NORTH	40.847677	-96.679151
10	105 28TH AVENUE SE	MINOT	ND	58701	MINOT	48.205789	-101.292282

© 2023 - Smith & Co. - Privacy

Figure 10 Locations page.

Mailing List

The mailing list page reads xml files that are generated by the client application. Each mailing list type is in a unique xml file. All the customers that have been added to these files are then output to the web application page as shown in Figure 11.



Mailing list

Desktop

- Nick Smith
  - Email: nick.smith@du.edu
  - Address: 1600 Pennsylvania Ave NW, Washington, DC 20006
  - Phone: 2024561111
- John Doe
  - Email: john.doe@microsoft.com
  - Address: 5335 Wisconsin Ave NW Ste 750, Washington, DC
  - Phone: 8003214321

Other

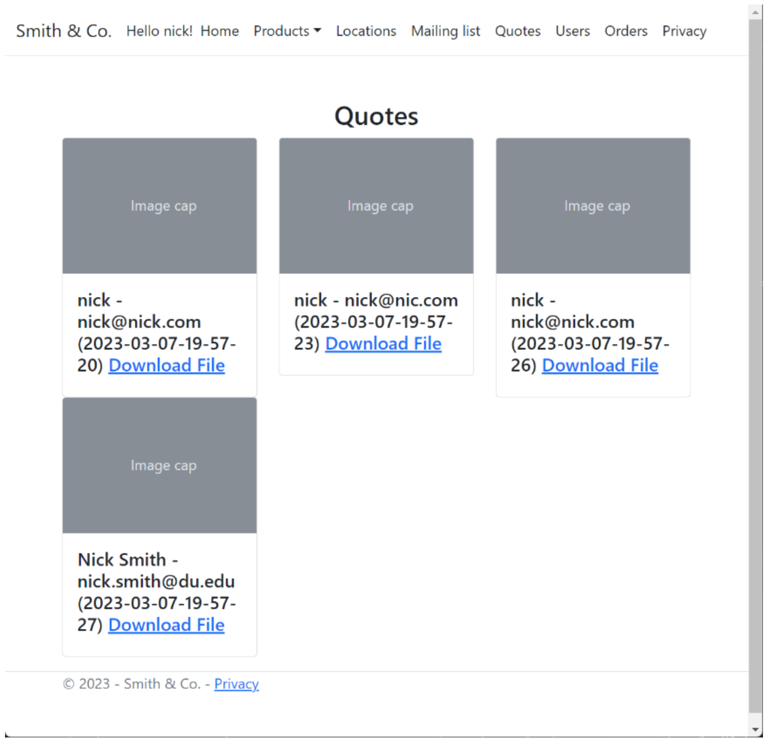
- Nick Smith
  - Email: nick.smith@du.edu
  - Address: 1600 Pennsylvania Ave NW, Washington, DC 20006
  - Phone: 2024561111
- John Doe
  - Email: john.doe@microsoft.com
  - Address: 5335 Wisconsin Ave NW Ste 750, Washington, DC
  - Phone: 8003214321

© 2023 - Smith & Co. - Privacy

Figure 11 Mailing list page.

Quotes

The quotes page reads a folder that contains all the quotes that have been generated by the client application. This then feeds the razor code to generate the HTML for the site. This list also allows for the user to download the file through the browser with the download file clickable text shown in blue, see Figure 12.



Quotes

nick - nick@nick.com (2023-03-07-19-57-20) Download File

nick - nick@nic.com (2023-03-07-19-57-23) Download File

nick - nick@nick.com (2023-03-07-19-57-26) Download File

Nick Smith - nick.smith@du.edu (2023-03-07-19-57-27) Download File

© 2023 - Smith & Co. - Privacy

Figure 12 Quotes page.

Users

The users page, see Figure 13, displays all the users for the application ecosystem. This reads from an XML file called users.xml; which is shared with the client application and is also used to store the passwords for the users. While the developer does need to be cautious to not display the hashed password, it is more secure than having the password save din playing text. Further development would ideally add an attribute class to allow for the password to be used within any class that requires it or something with this idea.

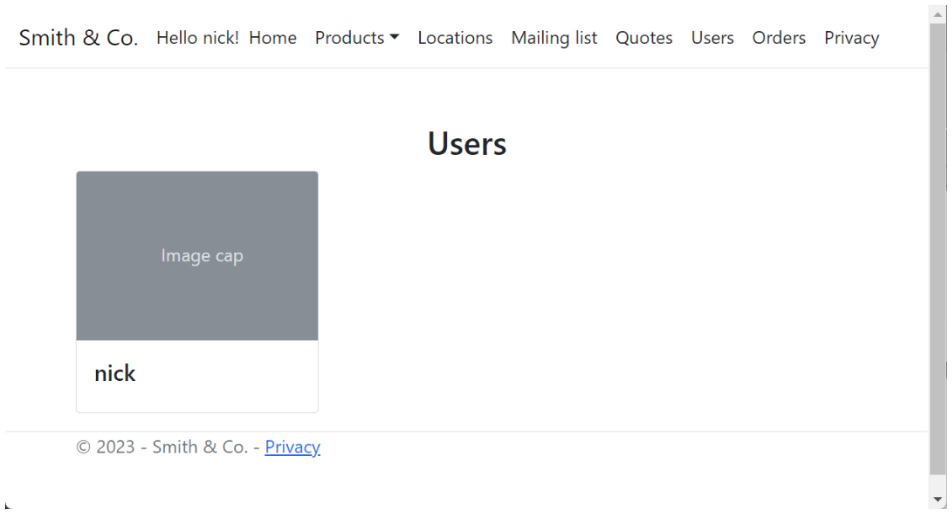
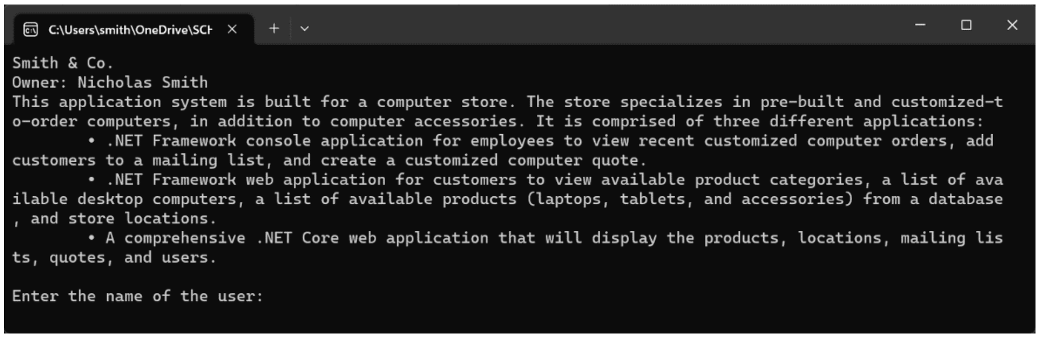


Figure 13 Users page.

.NET Framework Console Application

Console Framework

The console application starts with a prompt defining the application system similarly to the web application. The user is prompted to enter their username as shown in Figure 14. Once logged in the user is prompted with options to execute different tasks: view customized orders, add a customer to the mailing list create a customized computer quote and an option to close the application, see Figure 15. This prompt will re-appear until the user execute the close the application task.

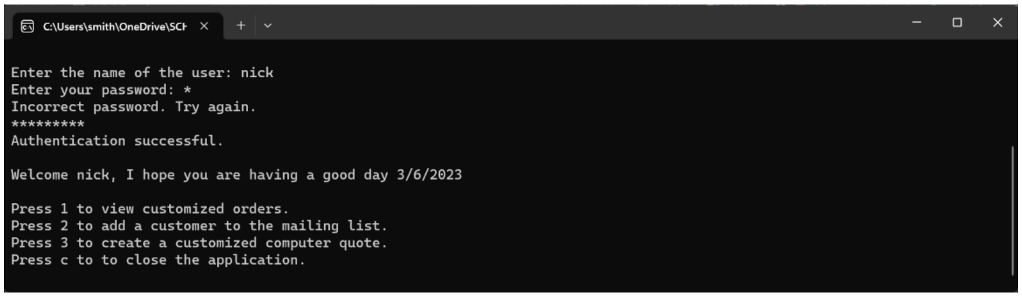


```
Smith & Co.
Owner: Nicholas Smith
This application system is built for a computer store. The store specializes in pre-built and customized-t
o-order computers, in addition to computer accessories. It is comprised of three different applications:
        • .NET Framework console application for employees to view recent customized computer orders, add
customers to a mailing list, and create a customized computer quote.
        • .NET Framework web application for customers to view available product categories, a list of ava
ilable desktop computers, a list of available products (laptops, tablets, and accessories) from a database
, and store locations.
        • A comprehensive .NET Core web application that will display the products, locations, mailing lis
ts, quotes, and users.

Enter the name of the user:
```

Figure 14 Console launch.

Authentication

This console application implements simple software security by prompting the user for a username and password. The password entered is compared to a hashed password that corresponds to the username. If successful the user is prompted with "Authentication successful. If the password is incorrect the user will be prompted to try again. See Figure 15.

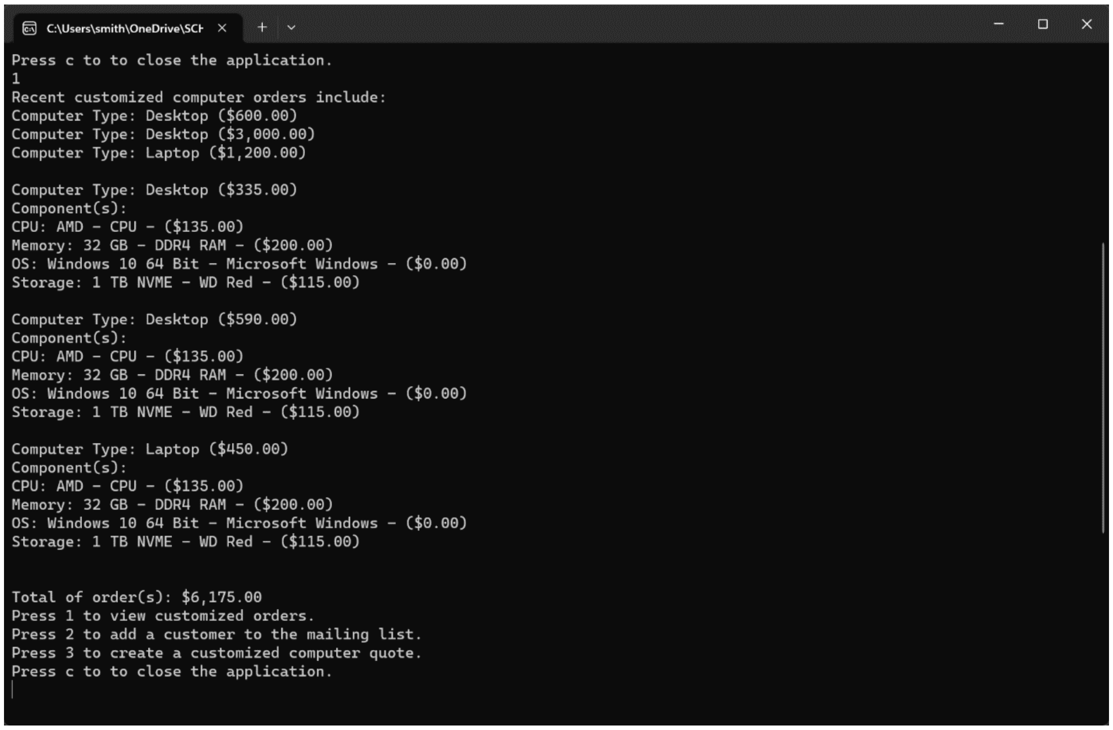


```
Enter the name of the user: nick
Enter your password: *
Incorrect password. Try again.
*********
Authentication successful.

Welcome nick, I hope you are having a good day 3/6/2023

Press 1 to view customized orders.
Press 2 to add a customer to the mailing list.
Press 3 to create a customized computer quote.
Press c to to close the application.
```

Figure 15 Console sign in.

Customized order

Customized order, see Figure 16, allows for the user to view existing products available to be ordered. This mock-up allows for both simple definitions and itemized computer systems. Ideally more options should be added to this task to filter products by specs and price.

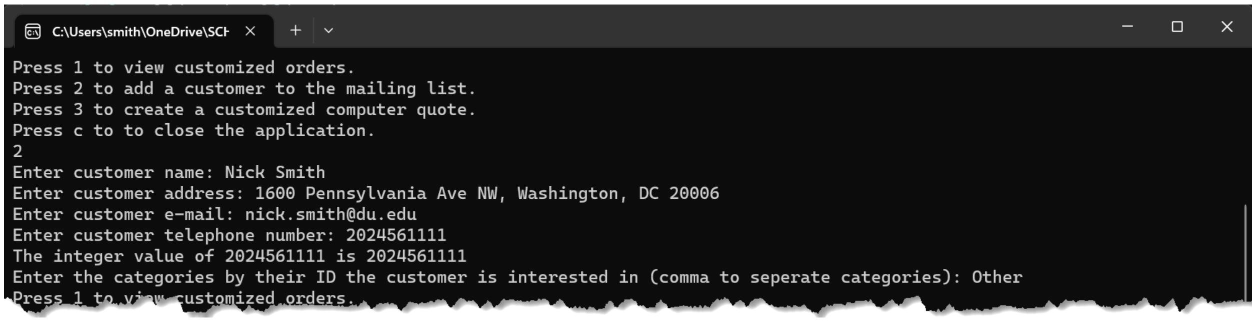


```
Press c to to close the application.
1
Recent customized computer orders include:
Computer Type: Desktop ($600.00)
Computer Type: Desktop ($3,000.00)
Computer Type: Laptop ($1,200.00)

Computer Type: Desktop ($335.00)
Component(s):
CPU: AMD - CPU - ($135.00)
Memory: 32 GB - DDR4 RAM - ($200.00)
OS: Windows 10 64 Bit - Microsoft Windows - ($0.00)
Storage: 1 TB NVME - WD Red - ($115.00)

Computer Type: Desktop ($590.00)
Component(s):
CPU: AMD - CPU - ($135.00)
Memory: 32 GB - DDR4 RAM - ($200.00)
OS: Windows 10 64 Bit - Microsoft Windows - ($0.00)
Storage: 1 TB NVME - WD Red - ($115.00)

Computer Type: Laptop ($450.00)
Component(s):
CPU: AMD - CPU - ($135.00)
Memory: 32 GB - DDR4 RAM - ($200.00)
OS: Windows 10 64 Bit - Microsoft Windows - ($0.00)
Storage: 1 TB NVME - WD Red - ($115.00)


Total of order(s): $6,175.00
Press 1 to view customized orders.
Press 2 to add a customer to the mailing list.
Press 3 to create a customized computer quote.
Press c to to close the application.
```

Figure 16 Console customized orders.

Mailing list

Mailing list allows for the user to enter information about a customer to add them to a category the customer is interest in. This information is saved to an XML file that is unique for each category. This information is live to be viewed by the web application. See Figure 17 for an example of data entry for adding a new customer to a mailing list.

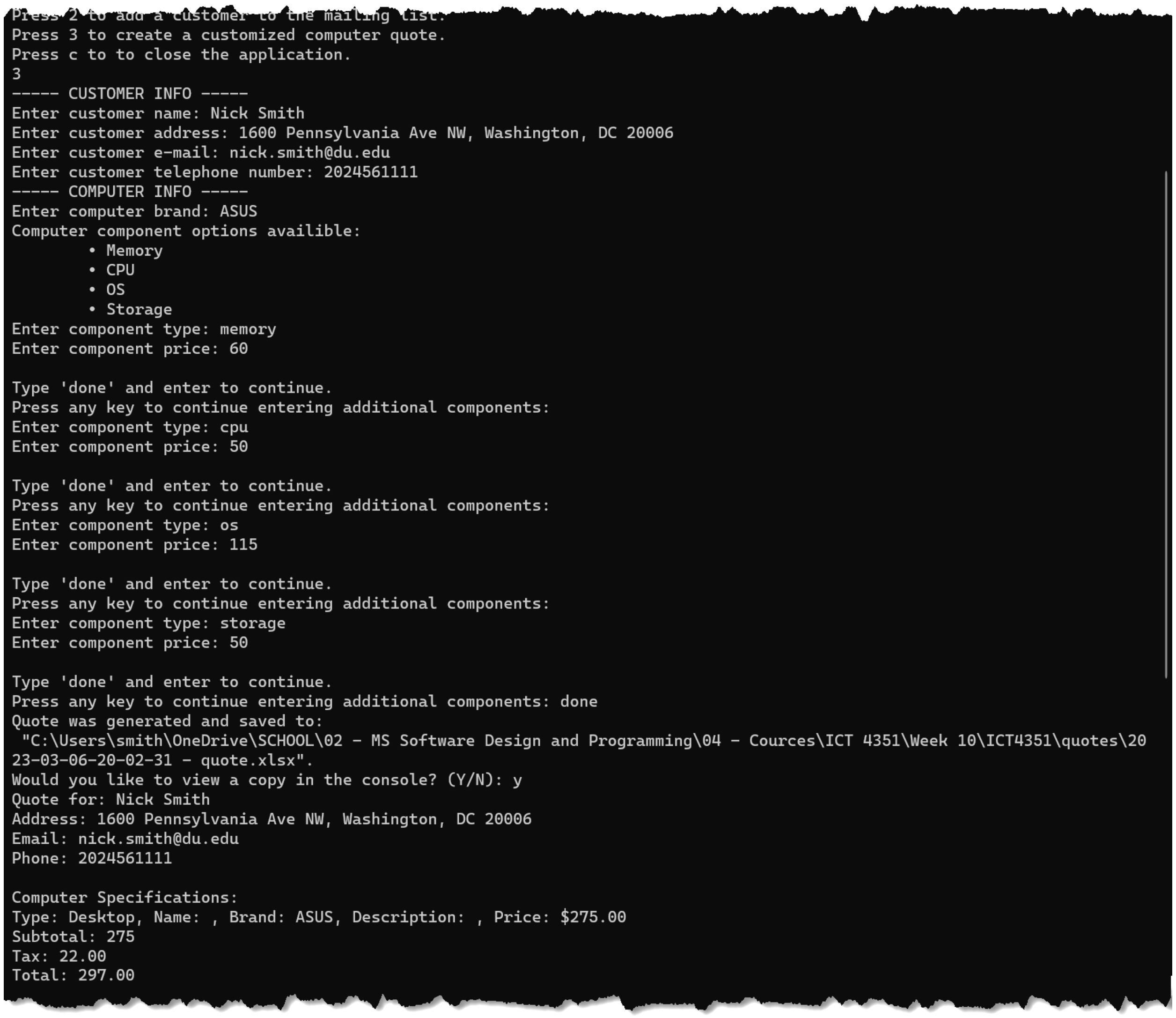


```
C:\Users\smith\OneDrive\SCH
Press 1 to view customized orders.
Press 2 to add a customer to the mailing list.
Press 3 to create a customized computer quote.
Press c to to close the application.
2
Enter customer name: Nick Smith
Enter customer address: 1600 Pennsylvania Ave NW, Washington, DC 20006
Enter customer e-mail: nick.smith@du.edu
Enter customer telephone number: 2024561111
The integer value of 2024561111 is 2024561111
Enter the categories by their ID the customer is interested in (comma to seperate categories): Other
Press 1 to view customized orders.
```

Figure 17 Console mailing list.

Create customized computer quote

The last option for this mock-up is the ability for the user to create a customized computer quote. See Figure 18 for an example user entry process for developing a new quote. As you can see once the quote information is successfully entered the user is prompted if they would like to view a copy of the quote in the console.



```
Press 2 to add a customer to the mailing list.
Press 3 to create a customized computer quote.
Press c to to close the application.
3
----- CUSTOMER INFO -----
Enter customer name: Nick Smith
Enter customer address: 1600 Pennsylvania Ave NW, Washington, DC 20006
Enter customer e-mail: nick.smith@du.edu
Enter customer telephone number: 2024561111
----- COMPUTER INFO -----
Enter computer brand: ASUS
Computer component options availible:
        • Memory
        • CPU
        • OS
        • Storage
Enter component type: memory
Enter component price: 60

Type 'done' and enter to continue.
Press any key to continue entering additional components:
Enter component type: cpu
Enter component price: 50

Type 'done' and enter to continue.
Press any key to continue entering additional components:
Enter component type: os
Enter component price: 115

Type 'done' and enter to continue.
Press any key to continue entering additional components:
Enter component type: storage
Enter component price: 50

Type 'done' and enter to continue.
Press any key to continue entering additional components: done
Quote was generated and saved to:
 "C:\Users\smith\OneDrive\SCHOOL\02 - MS Software Design and Programming\04 - Cources\ICT 4351\Week 10\ICT4351\quotes\20
23-03-06-20-02-31 - quote.xlsx".
Would you like to view a copy in the console? (Y/N): y
Quote for: Nick Smith
Address: 1600 Pennsylvania Ave NW, Washington, DC 20006
Email: nick.smith@du.edu
Phone: 2024561111

Computer Specifications:
Type: Desktop, Name: , Brand: ASUS, Description: , Price: $275.00
Subtotal: 275
Tax: 22.00
Total: 297.00
```

Figure 18 Console generate quote.

Conclusion

In conclusion, the computer store mock-up application created in this lab assignment is a comprehensive .NET Core web application that meets the requirements of displaying specific content based on user role. This is used in conjunction with a console application developed earlier in the design process.

The application includes a login form, user authentication, and authorized access pages for certain users only. The application comprises of five pages: Products, Locations, Mailing Lists, Quotes, and Users. The Products page lists all the products grouped by the type field. The Locations page displays all the stores from an XML file. The Mailing Lists page shows the customers added to various XML files generated by the console application. The Quotes page provides a downloadable list of all the quotes generated by the console application. Lastly, the Users page displays the names of the users from an XML file generated by the console application.

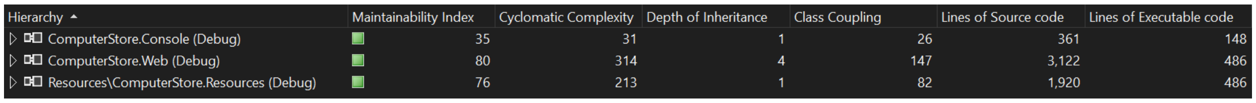
The application system is built with ASP .NET 6.0 for the website, .NET Framework 4.8 for the console application, and has a shared class library for the computer store resources that is targeted for .NET Standard 2.0. The mock-up was developed using Visual Studio 2022, and Git was used to document and track changes during the development process. Overall, this application demonstrates the capabilities of .NET Core and ASP .NET in creating secure and efficient web applications.

Lessons Learned

1. Utilizing a shared component approach helps in consistent standards and centralized efforts for feature updates and bug fixes, reducing development time.
2. Git can be used as a repository management software to document and track changes during the development process.
3. ASP.NET Core Web App (Model-View-Controller) allows for Razor pages to be used for generating the HTML that will be sent to the client computers.
4. Implementing user authentication with Microsoft.AspNetCore.Authorization restricts access to certain pages for authorized users only.
5. Error messages can be used to help users correct their mistakes.

Reference

Appendix A



Hierarchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Source code	Lines of Executable code
ComputerStore.Console (Debug)	🟩 35	31	1	26	361	148
ComputerStore.Web (Debug)	🟩 80	314	4	147	3,122	486
Resources\ComputerStore.Resources (Debug)	🟩 76	213	1	82	1,920	486

Figure 19 Code metrix.