**CS 504 – Software Engineering**

**HOP06 – Blazor Page Application – Login - Logout System**

11/08/2020 Developed by Kim Nguyen & Yu-Che Liu

Center for Information Assurance (CIAE) @City University of Seattle (CityU)

Icon

Description automatically generated

**Before You Start**

* Version numbers may not match with the most current version at the time of writing. If given the option to choose between stable release (long-term support) or most recent, please choose the stable release rather than beta-testing version.
* This tutorial targets Windows users and MacOS users.
* There might be subtle discrepancies along the steps. Please use your best judgement while going through this cookbook style tutorial to complete each step.
* For your working directory, use your course number. This tutorial may use a different course number as an example.
* The directory path shown in screenshots may be different from yours.
* If you are not sure what to do or confused with any steps:
  1. Consult the resources listed below.
  2. If you cannot solve the problem after a few tries, ask a TA for help.

**Learning Outcomes**

Students will be able to:

* Continue customizing Blazor Application
* Adding login and logout system to the application.

**Resources:**

* Microsoft | ASP .NET - <https://dotnet.microsoft.com/apps/aspnet>

**Blazor App – Create Login & Logout system**

**Please move the Blazor App folder from “Module 5” folder to “Module 6”.**

(Under the Hands-on Practice folder)

This week, we will be creating a login and logout system for our application. This week’s HOP result will look like following:

<https://youtu.be/ee8GXoqbRMY>

1. Open the Blazor App project in VSCode. Under Pages folder, create a new file called Login.razor

A picture containing meter, clock

Description automatically generated

1. Add the following code to the Login.razor

Text

Description automatically generatedText

Description automatically generated

1. Under Data folder, create a file named MemberData.cs file:

A picture containing meter

Description automatically generated

1. Type the following code into the MemberData.cs file:

Text

Description automatically generated

1. Under the Data folder, create a file named LoginService.cs file:

A picture containing meter

Description automatically generated

1. Type the following code into the LoginService.cs file:

Text

Description automatically generated

1. Under the root folder, in Startup.cs file, add the following line of code:

Text

Description automatically generated

1. Run the app:

dotnet watch run

**To see the login page, include /login into the url:**

*(Note: Your port number might be different from the screenshot)*



You should see:

Graphical user interface, application

Description automatically generated

Go ahead and experiment the page.

1. **Watch the following video to see the explanations of all the code we just created this week:**

<https://youtu.be/YdaidaUYdWY>

Next week, we will be adding Data base and have dynamic data.

1. **Required: Link the Login page we just created to the main menu. Expected result:**

**Graphical user interface, application

Description automatically generated**

**Push your work to GitHub**

Open the terminal from the VSCode by hitting the control + ~ key, make sure you are in the right path, for example: /Desktop/CS504/HOP06-YourGitHubUsername/Module 6

Type the following command:

git add . (to copy all changes you have made)

git commit -m “Submission for Module 6 – Your Name” (To add a message to your submission)

git push origin master (to upload your work to Github)