**CS 628 Full-Stack Development II**

**HOS04B: Views – Reusable Layout**

July 19, 2020

12/29/2020 Updated by Min

School of Technology & Computing (STC)

City University of Seattle (CityU)

**Before You Start**

* The directory path shown in screenshots may be different from yours.
* Some steps are not explained in the tutorial**.** If you are not sure what to do:
  1. Consult the resources listed below.
  2. If you cannot solve the problem after a few tries, ask a TA for help.

**Resources**

* Holmes, S. & Harber, C. (2019). [Getting MEAN with Mongo, Express, Angular, and Node, Second Edition](https://login.proxy.cityu.edu/sso/skillport?context=148121). Manning Publications. (ISBN 9781617294754)
  + Chapter 4 - Building a Static Site with Node and Express
    - 4.3 Creating Some Views
  + Appendix B: Installing and Preparing the Supporting Cast
    - Twitter Bootstrap
* Pug, <https://pugjs.org/api/getting-started.html>

**Learning Outcomes**

* Understand the template engine in Express - Pug
* Understand a front-end layout framework
* Apply the reusable layout to an Express web application.

**Activities**

* Preparing for a MEAN project environment
* **Understanding the basic structure of the reusable layout**
* **Building the Navigation**
* **Wrapping the Content**
* **Adding the Footer**
* **Adding Some Custom Styles**
* Source Code Management and Deployment

**Preparing for a MEAN project environment**

1. Copy the “module04a” directory to the “module04b” one.
2. Open the “module04b” folder in the VSCode.  
   Close the “Welcome” window.  
   Open the terminal (Ctrl+`).
3. Test whether the application works correctly.  
   >>>npm start  
     
   <http://localhost:3000>A screenshot of a cell phone

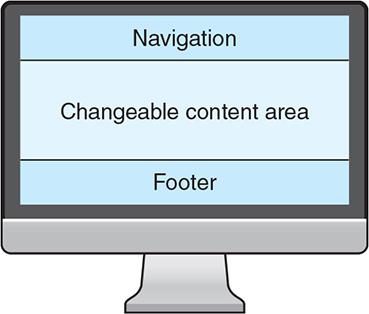
   Description automatically generated

##### **Understanding the basic structure of the reusable layout**

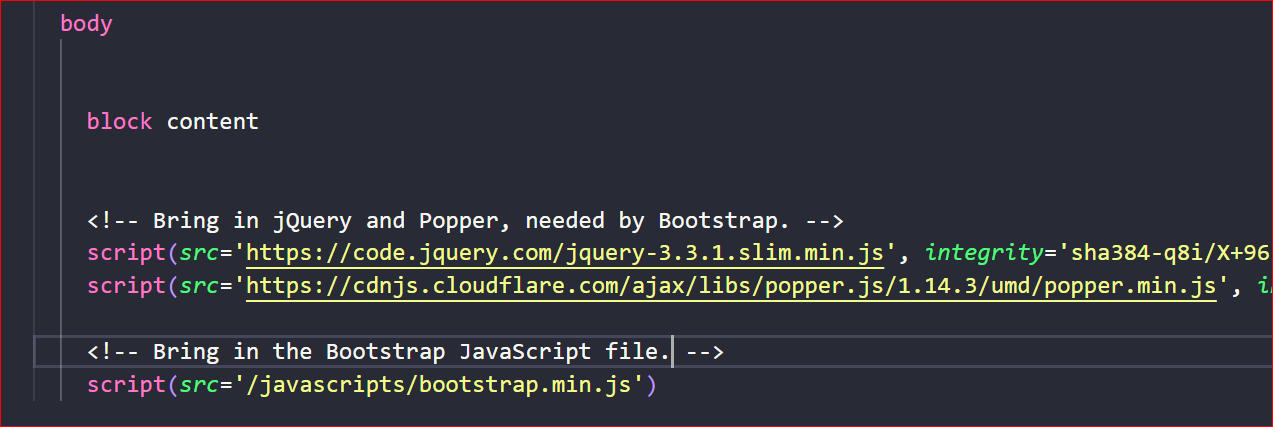
At the top of each page, you will want a navigation bar and logo.

You will have a content area in the middle.

At the bottom of the page, you will have a copyright notice in the footer.



1. Open the Pug HTML template file “layout.pug” under the “app\_server/views” views directory.  
   Analyze the basic structure of the reusable layout.
   1. Navigation – empty
   2. Content – “block content”
   3. Footer - empty



##### **Building the Navigation Bar**

Bootstrap offers a collection of elements and classes that you can use to create a sticky navigation bar that is fixed to the top and collapses the options into a drop-down menu on mobile devices.

In the navigation, the following is going to be created:

* The “Loc8r” logo linking to the homepage
* An “About” link on the left, pointing to the /about URL page

##### **Logo**

1. Copy and paste the below code for the “logo” linking to the homepage (“/” path) into “body” before “block content” in **layout.pug**.

  body

    nav*.navbar.fixed-top.navbar-expand-md.navbar-light*

*.container*

        a*.navbar-brand*(*href*='/') Loc8r

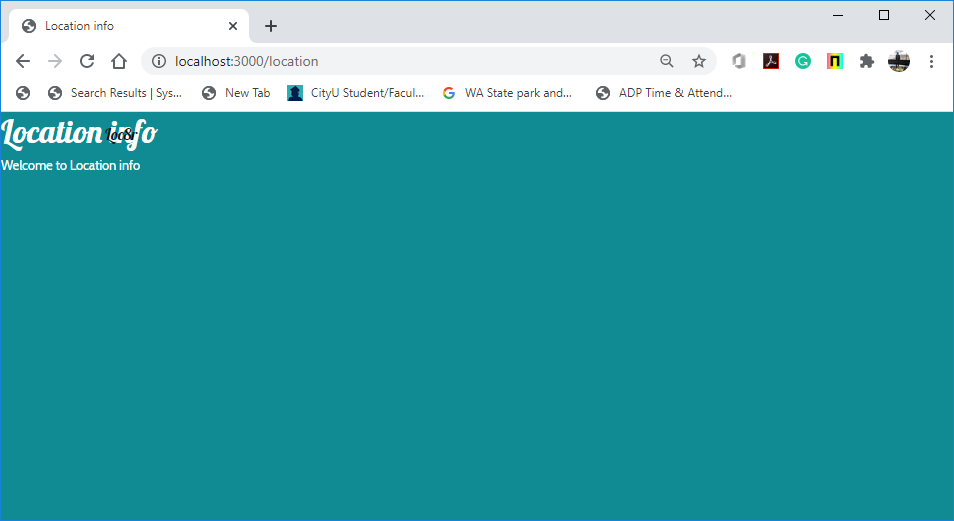
          button*.navbar-toggler*(*type*='button', *data-toggle*='collapse',*data-target*='#navbarMain')

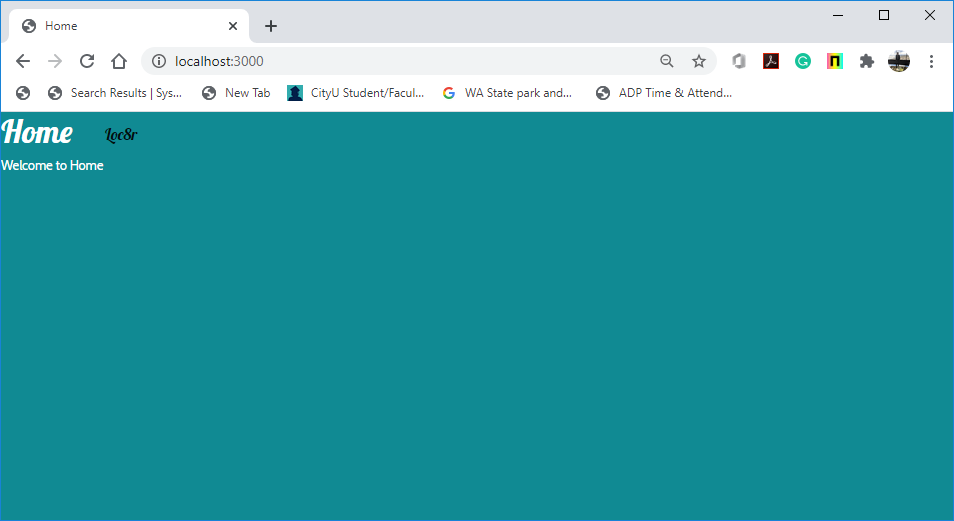
            span*.navbar-toggler-icon*

*.container.content*

block content

1. Run the application for testing.   
   >>>npm start
   1. In your browser, type the “localhost:3000/location/” URL path.
   2. You can see that the navigation overlays the content; this problem will be addressed during content layouts.
   3. In the “Location” web page above, click the “Loc8r” logo.
   4. You can link to the homepage





##### **About Menu**

1. Copy and paste the below code for the “About” linking to the “About” page (“/about” path) into the body before “block content”.

  body

    nav*.navbar.fixed-top.navbar-expand-md.navbar-light*

*.container*

        a*.navbar-brand*(*href*='/') Loc8r

          button*.navbar-toggler*(*type*='button', *data-toggle*='collapse',*data-target*='#navbarMain')

            span*.navbar-toggler-icon*

*#navbarMain.navbar-collapse.collapse*

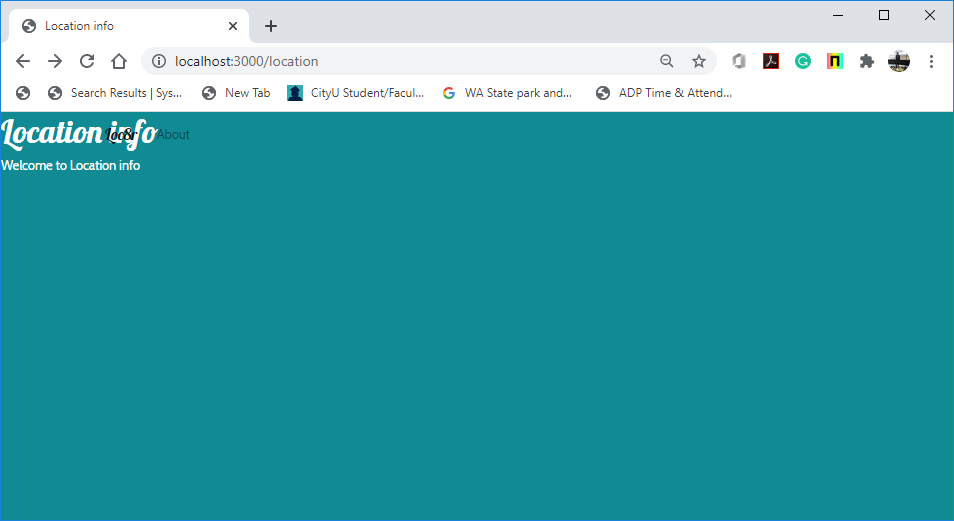
          ul*.navbar-nav.mr-auto*

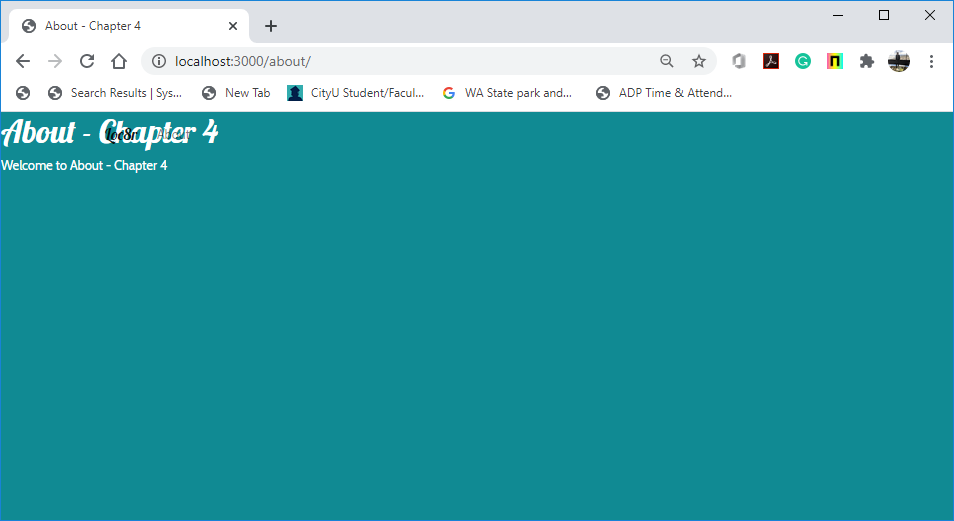
            li*.nav-item*

              a*.nav-link*(*href*='/about/') About

block content

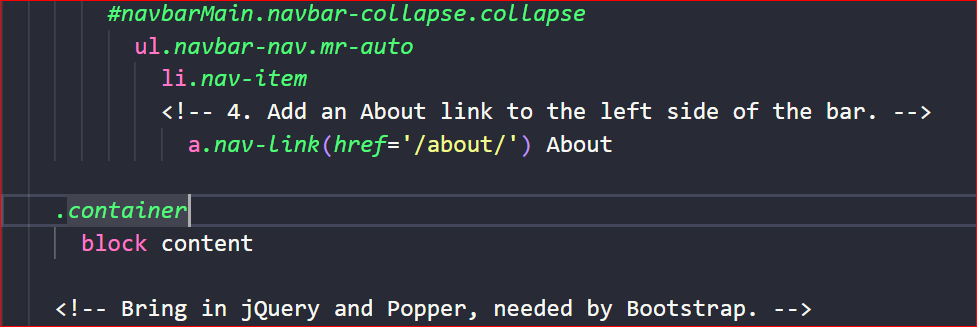
1. Run the application for testing.   
   >>>npm start  
   1. In your browser, type the “localhost:3000/location/” URL path.
   2. You can see that the navigation overlays the content; this problem will be addressed during content layouts.
   3. In the “Location” web page above, click the “About” menu.
   4. You can link to the “About” page.



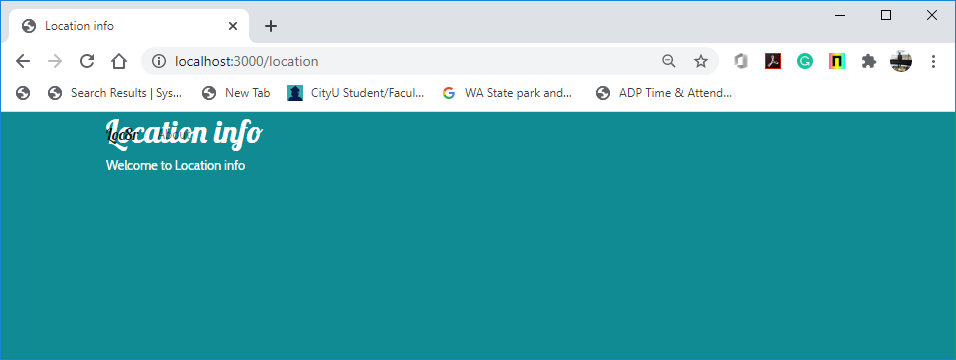


##### **Wrapping the Content**

1. Currently, the content block is anchored to the left margin and is unconstrained, meaning that it stretches the full width of any device.
   1. We want that the “div” with a class of container is centered in the window and constrained to sensible maximum widths on large displays.
   2. The contents of a container div remains aligned to the left as normal, though.
   3. Pug does not include any HTML tags.
   4. Correct indentation is critical for providing the expected outcome in Pug.
   5. Enter “.container” and indent “block content.”



1. Run the application for testing.   
   >>>npm start
   1. In your browser, type the “localhost:3000/location/” URL path.
   2. You can see that the navigation overlays the content; this problem will be addressed during content layouts.
   3. In the “Location” web page above, you can see the content was moved to the left.

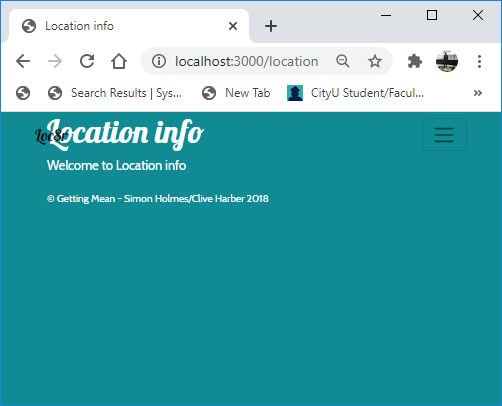


##### **Adding the Footer**

1. At the bottom of the page, you want to add a standard footer.
   1. Add the following code for your simple footer in “layout.pug.”
   2. This code is placed inside the container div that holds the content block, so when you add it, make sure that the footer line is at the same level of indentation as the block content line.

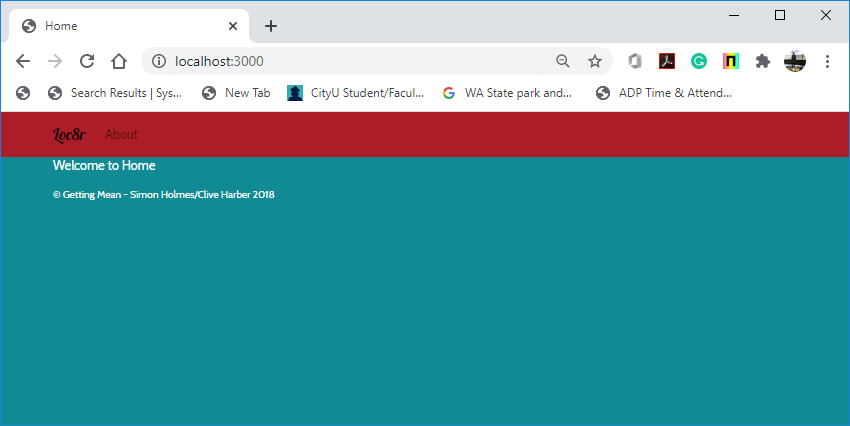
##### 

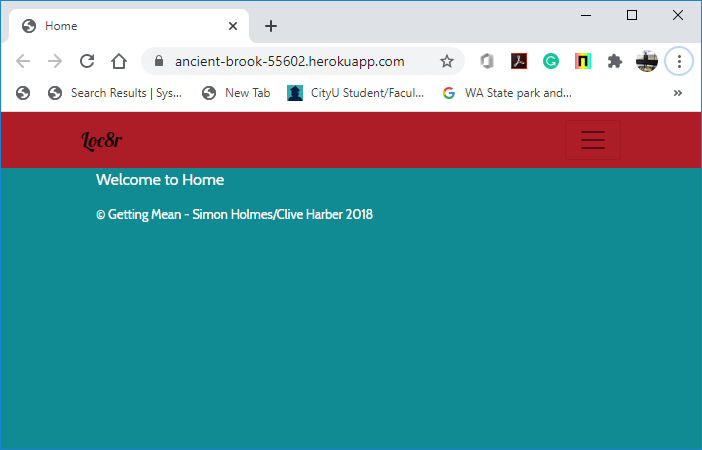
1. Run the application for testing.   
   >>>npm start
   1. In your browser, type the “localhost:3000/location/” URL path.
   2. You can see that the navigation overlays the content; this problem will be addressed during content layouts.
   3. In the “Location” web page below, you can see the footer at the bottom..



##### **Adding Some Custom Styles**

1. The Loc8r application in this book uses some custom styles that we created.   
   Update the “public/stylesheets/style.css” file with [“](https://viewer.books24x7.com/assetviewer.aspx?bookid=148121&chunkid=845249946&resumebookmarkid=d096c3d8-c7cf-ea11-990a-0050569562bd) Listing B.1: Custom styles to give Loc8r a more distinctive look,” which shows a good starting point and
2. Run the application for testing.   
   >>>npm start
   1. In your browser, type the “localhost:3000/location/” URL path.  
      You cannot see that the navigation overlays the content.
   2. In the “Location” web page above, click the “About” menu.
   3. You can link to the “About” page.
   4. In the “About” web page, click the logo.
   5. You can link to the home page.





**Source Code Management and Deployment**

**Push your work to GitHub**

Run the following commands to push your work to the GitHub repository:

Open the terminal from the VSCode by hitting the control + ~ key and type the following command:

>>> git add .

>>> git commit -m “Submission for Module 4 --yourname”

>>> git push origin master

**Push your work to Heroku**

>>> git push heroku master

If you cannot remember your branch name, run the command “git status” to check.