**CS 628 Full-Stack Development II**

**HOS04A: Routes & Controllers**

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
    - 4.1 Defining the Routes in Express
    - 4.2 Building Basic Controllers

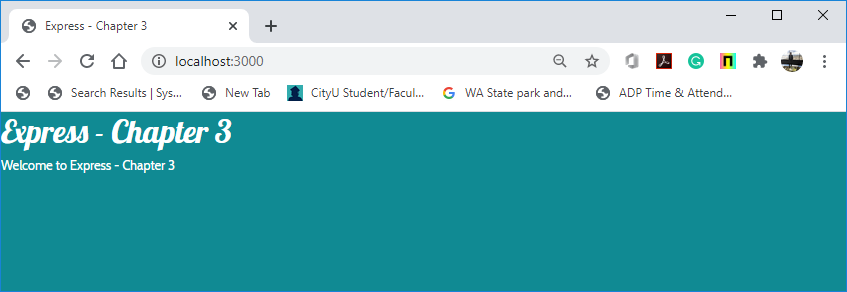
**Learning Outcomes**

* Understand routes
* Designing routes (or URL paths)
* Mapping routes and controllers

**Activities**

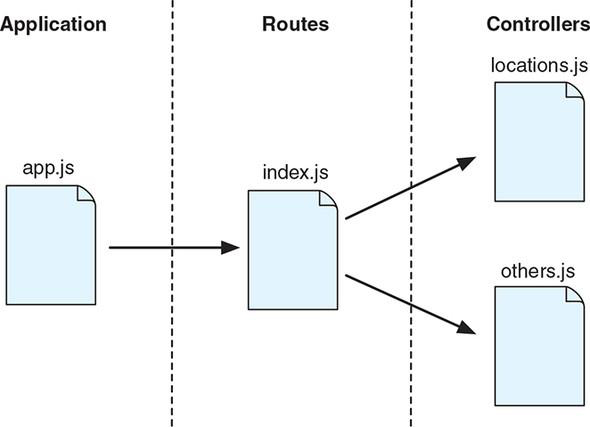
* Preparing for a MEAN project environment
* **Requiring the Controller files in a route file**
* **Setting up the routes**
* **Testing the routes**
* **Adding the Others controller**
* **Adding the Locations Controller**
* **Testing the Controllers and Routes**
* Source Code Management and Deployment

**Preparing for a MEAN project environment**

1. Copy the “module03c” directory to the “module04a” one.
2. Open the “module04a” folder in the VSCode.  
   Close the “Welcome” window.  
   Open the terminal (Ctrl+`).
3. Test whether the application works correctly.  
   >>>npm start  
     
   <http://localhost:3000>

##### **Requiring the Controller files in a route file**

We have to refer the two controller files (location.js and others.js) in the routes file index.js. Figure 4.3 shows the proposed file architecture for routes and controllers in your application.



Note: The controller files (location.js and others.js) are not yet created, it will be done shortly.

1. In the “index.js” under the routes, you will do the following things:
   1. Delete the “ctrlMain” variable, which requires the “main.js” under the controllers.
   2. Add the “ctrlLocations” variable, which requires the “locations.js” under the controllers.
   3. Add the “ctrlOthers” variable, which requires the “others.js” under the controllers.

Before  
A screenshot of a cell phone

Description automatically generated

After  
A picture containing sitting, screen, black, table

Description automatically generated

Now you have two variables (“ctrlLocations” and “ctrlOthers”) that you can reference in the route definitions, containing different collections of routes.

##### **Setting up the Routes**

From Table 4.1, you already know which paths you want to map, so you need to put everything together into the “routes/index.js” file.

The Locations is going to have 3 pages

* List page
* Details page
* Add Review page

The Others is going to have 1 page

* About page

|  |  |  |
| --- | --- | --- |
| Defining a URL path (i.e. route ) for each of the screens in the prototype | | |
| **Collection** | **Screen** | **URL path** |
| Locations | List page - List of locations (the homepage) | / |
| Locations | Details page - Location detail | /location |
| Locations | Add Review page - Location review form | /location/review/new |
| Others | About page - About Loc8r | /about |

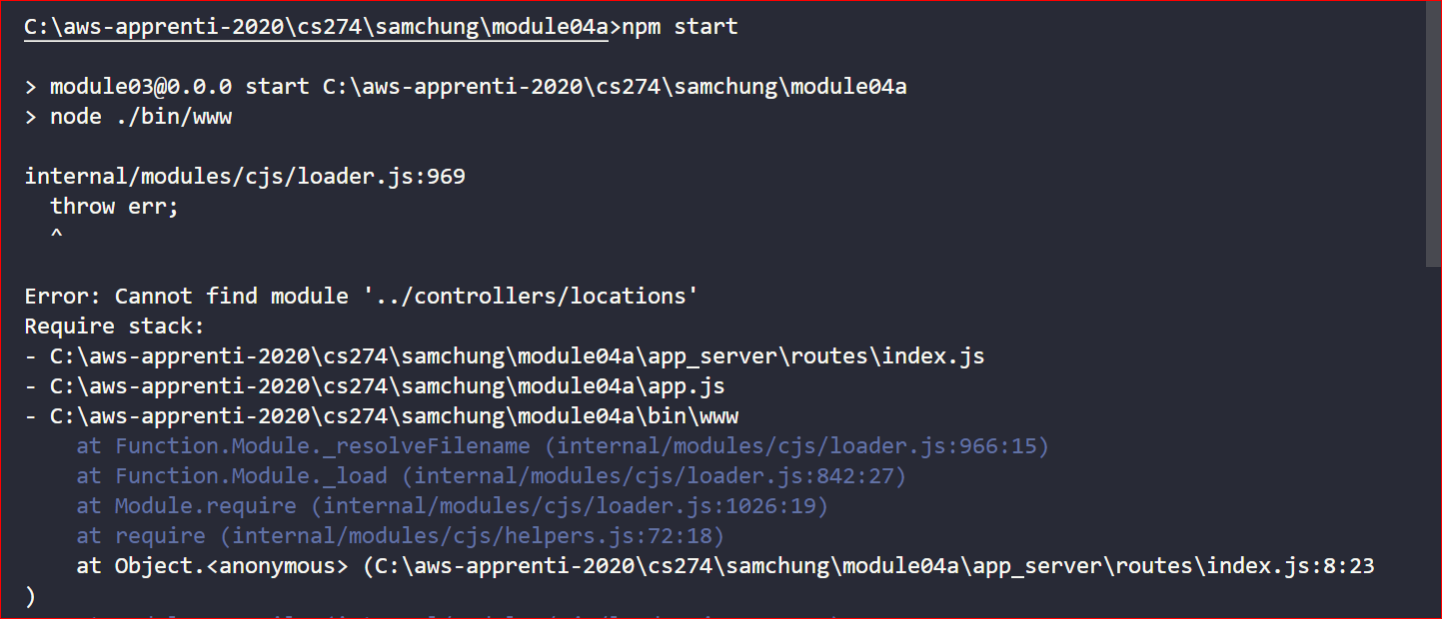
1. In the “index.js” routing file under the routes, you will map the defined URLs to some specific controllers.

* Defines location routes and maps them to controller functions
* Defines other routes.

A screenshot of a cell phone

Description automatically generated

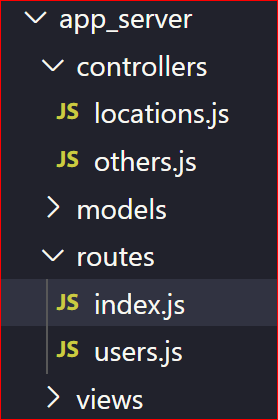
**Testing the Routes**

1. Run the application for testing.  
   >>>npm start  
     
   Since you haven’t created the mapped controller files, you will encounter errors.  
   

**Adding the Others controller**

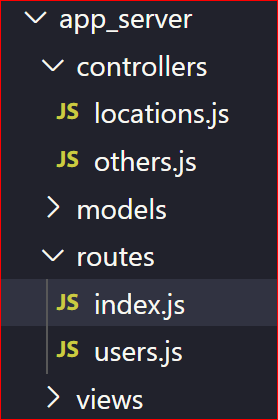
In the above steps, we created the routes for two files “others” and “locations.”

Now let’s create these files under the controllers folder.

1. Under the “controllers” directory, the “main.js” controller file is available.   
   We do not require it.   
   So, rename by right clicking on the “main.js,” change the name to “others.js”.  
   
2. We are going to create “About page” in “others”.   
   The controller name is going to be “about”.   
   In the “others.js” controller file, enter the following code. ~~update the existing code with the one below.~~  
     
     
   A screenshot of a cell phone

   Description automatically generated

**Adding the Locations Controller**

1. Under “app\_server/controllers,” create a file “locations.js.”  
   
2. Create three basic controller functions: homelist, locationInfo, and addReview.  
   A screenshot of a cell phone

   Description automatically generated

**My notes: This hos skips some steps. You also need to edit the app.js so index.router is called when certain urls are called. And in the router, you specify which urls correspond to which controllers.**

app.use('/', indexRouter);

app.use('/location', indexRouter);

app.use('/location/review/new', indexRouter);

app.use('/users', usersRouter);

**Testing the Controllers and Routes**

1. Run the application for testing.  
   >>>npm start  
     
   A screenshot of a cell phone

   Description automatically generated  
     
   A screenshot of a cell phone

   Description automatically generated  
     
   A screenshot of a cell phone

   Description automatically generated

A screenshot of a cell phone

Description automatically generated

**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.