

# CSCI 421 – Advanced Web Development

## Lab Introduction and Overview

For this class, you will be using hosted services for developing and running your web applications. The choice of using Amazon Web Services (AWS) has been made due to the fact that AWS is an industry standard, very low cost, and embraced by companies large and small. In this introduction you will be instructed on what you will need for this class as far as lab resources go and how to get started. You will even be shown a very good video tutorial that you are required to both watch and “follow along” in order to set up your own development environment and first MEAN full stack application.

### Instructions:

1. You must create (if needed) an Amazon Web Services (AWS) Account for using Amazon Lightsail. You can sign-up for an AWS account below:

<https://aws.amazon.com/lightsail/>

2. Login in to the AWS console and create a MEAN Stack Lightsail instance:

#### Instance location ?



You are creating this instance in **Ohio, Zone A** (us-east-2a)

[Change AWS Region and Availability Zone](#)

#### Pick your instance image ?

Select a platform



Select a blueprint



WordPress  
4.9.8



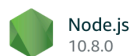
WordPress  
Multisite  
4.9.8



LAMP (PHP 5)  
5.6.37-2



LAMP (PHP 7)  
7.1.20-1



Node.js  
10.8.0



Joomla  
3.8.11



Magento  
2.2.5



MEAN  
4.0.1



Drupal  
8.5.6



GitLab CE  
11.1.4-1



Redmine  
3.4.6



Nginx  
1.14.0-1

3. Create a GitHub account (if needed) as this will be used later in this course:

<https://github.com/>

4. Create your first MEAN Stack application:

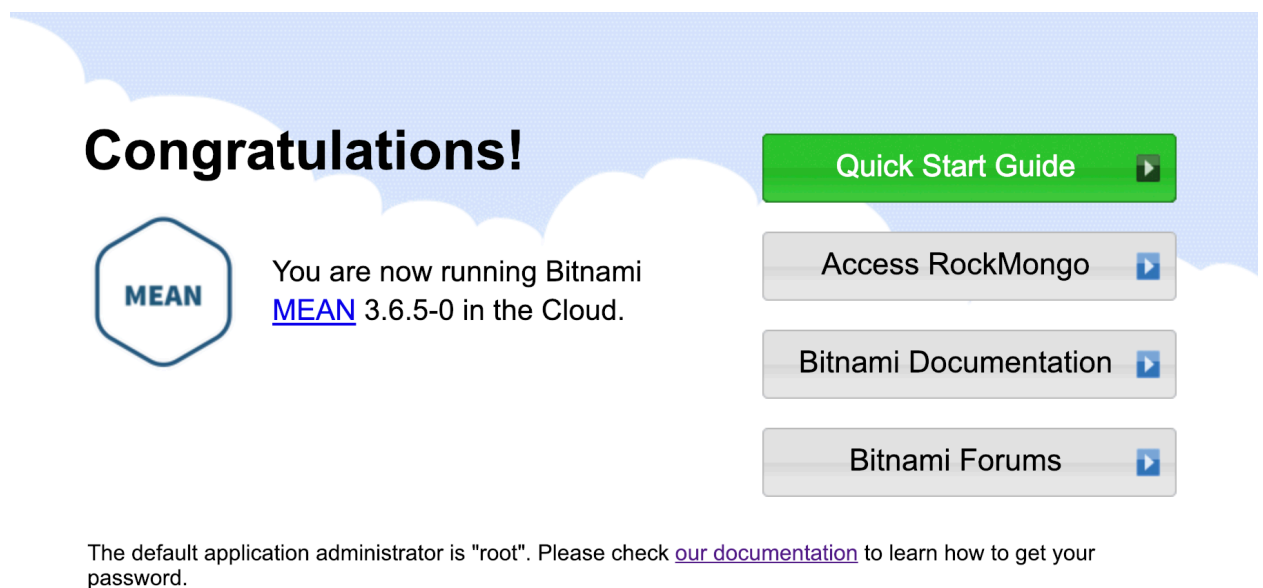
An excellent tutorial here, watch it and follow it step-by-step:

<https://youtu.be/iohBEVf4uIQ>

5. When you get the MEAN instance up and running (see steps i. and ii. from the instructor's notes below) and have port 8080 set up properly you should be able to navigate to the application using a browser and navigating to:

http:<ipaddress>:8080.

There you should see the following:



6. IMPORTANT: Get the "ToDo" application from the tutorial working (see steps iii. through v. from the instructor's notes below). The app can be loaded from a browser by navigating to the default website via the following URL:

http://<ipaddress>

Once you have navigated to the ToDo application you should see the website as is presented in the image on the next page:

## No tasks to display. Add one above.

*Served by host: ip-172-26-12-226*

7. Lastly, make sure that the ToDo application works, that you can add, edit and delete tasks, and when all is working and you have set up the node instance to be persistent (run when you close your connection...see the notes below) then send the instructor your ipaddress of your ToDo application via an email: [thomas.rogers@millersville.edu](mailto:thomas.rogers@millersville.edu)

### **Instructor's notes** from the tutorial (may be of help):

#### i. Creating Lightsail MEAN instance:

- Create instance using AWS
- Download default.pem file under accounts, restrict perms: `chmod 400 default.pem`
- SSH to instance via: `ssh -i default.pem bitnami@<ip_address>`
- Install emacs if you wish: `sudo apt install emacs`

#### ii. Changing default Admin web app to use port 8080 (instead of 80):

- Change default Listen port by editing `/stack/apache2/conf/httpd.conf`
- Change bitnami Listen port by editing: `/stack/apache2/conf/bitnami/bitnami.conf`
- Restart apache to effect changes: `sudo stack/ctlscript.sh restart apache`
- Open up firewall on Lightsail instance: (...) "Manage", then "Networking", scroll down to "Firewall", click "+ Add Another", enter: Custom, TCP, 8080
- Should be able to navigate a browser to your default app given the IP address of your MEAN instance: `http://<ipaddress>:8080`

#### iii. Creating an Application Database for Todo application:

- Must have mongo db password: `cat bitnami_application_password`
- Copy password to clipboard, paste in command: `mongo admin --username root -p <password>`

- Show databases: `mongo> show databases`
- Create database: `mongo> use tasks`
- Create user tasks: `mongo> db.createUser(`  
     ... {  
     ... user: "tasks",  
     ... pwd: "tasks",  
     ... roles: ["dbOwner"]  
     ... }  
     ... )

**NOTE:** Skip step iv below as RockMongo has been deprecated and is no longer available. Proceed to step v.

iv. Setting Up for RockMongo (CLI for mongo DB):

- In another terminal window: `ssh -N -L 8888:127.0.0.1:8080 -i default.pem bitnami@<ip_address>`  
 (Note: above command leaves window open)
- In browser tab navigate to: `127.0.0.1:8888/rockmongo`
- Login with same credentials as above.

v. Installing ToDo application:

- `> git clone https://github.com/mikegcoleman/todo.git`
- `> cd todo`
- `> npm install --production`
- `> emacs .env`

**Note:** The following are the two lines for placing in the .env file, note the DB\_URL line is a single line so feel free to cut/paste:

`PORT=80`

`DB_URL=mongodb://tasks:tasks@localhost:27017/?authMechanism=SCRAM-SHA-1&authSource=tasks`

vi. Running ToDo application web server (when in todo folder):

`>sudo node ./bin/www`

Or, to run in background (no need to keep connection open):

`>nohup sudo node ./bin/www &`

Note: above command starts node and it remains active)

To find node processes (and get PID) and then kill:

`>ps aux | grep node`

`>sudo kill -9 <PID>`

- Once the todo web server is running you should be able to navigate to the IP address of your MEAN instance to see the todo app: `http://<ipaddress>`