

MEAN.machine

**Business Career Center Application
Version 1.0 Documentation**

Names: Benjamin Behrend, Donna Tabique,
Jaime Wallitschek, Kawika Bader, Zhengping Duan

Date: (08/03/2015)

Contents

Installation for MAC.....	3
1. Yeoman.....	3
1.1 Yeoman Installation.....	3
1.2 Server Setup.....	3
1.3 Additional Information.....	4
2. MongoDB.....	4
2.1 MongoDB Windows Setup.....	4
2.2 MongoDB Mac Setup.....	5
2.3 MongoDB Database Creation.....	5
3. Nodejs.....	6
3.1 Installation.....	6
Amazon Web Service (AWS) Installation.....	6
1. Setup Ubuntu Server.....	6
2. Git Clone the Repository.....	6
3. Run the Application as a Service.....	6

Installation for MAC

1. Yeoman

1.1 Yeoman Installation

1. Before installing Yeoman, download Nodejs and Mongodb onto your system. For Mongodb instructions, see Section 2 for Mongodb and Section 3 for Nodejs.

2. Put the Mongodb "/data/db" directory into your application's directory.

3. Open Terminal or Command Line.

4. Change the current directory to your application's directory by typing
`cd appDirectoryName`
Where *appDirectoryName* is the name of your application directory folder.

5. To install Yeoman on your system, type:
`npm install -g yo`

6. Install the MEANJS generator by typing:
`npm install -g generator-meanjs`

7. Type: `yo`

8. Yeoman will then prompt the user to answer the following questions:

```
? 'Allo <user>! What would you like to do? Meanjs
Make sure you are in the directory you want to scaffold into.
This generator can also be run with: yo meanjs
You're using the official MEAN.JS generator.
? What would you like to call your application? asuCareerCenter
? How would you describe your application? Full-Stack JavaScript
with MongoDB, Express, AngularJS, and Node.jss)
? How would you describe your application in comma seperated key
words? MongoDB, Express, AngularJS, Node.jss)
? What is your company/author name? MEAN.machine
? Would you like to generate the article example CRUD module? Yes
? Which AngularJS modules would you like to include? ngCookies,
ngAnimate, ngTouch, ngSanitize
```

9. This will create and generate the file structure and basic app files, such as a Home Page and a Header.

1.2 Server Setup

1. In the Terminal/Command Prompt window, install grunt by typing:
`npm install -g grunt-cli`

2. Type: `grunt`

```
If you are getting the following error:
MEAN.JS application started on port 3000
Could not connect to MongoDB!
Error: failed to connect to [localhost:27017]
```

- Open another Terminal/Command Prompt window and type *mongod* to establish a connection.
3. If you do not want to use grunt, then type *node server.js* to run the server.

1.3 Additional Information

1.3.1 Proxies

1. If your network connection has proxies, you may run into trouble setting up Nodejs or MongoDB.
2. To work around the proxies for Nodejs, type the following into your Terminal/Command prompt Window:

```
npm config set proxy http://proxy.company.com:8080  
npm config set https-proxy http://proxy.company.com:8080
```

2. MongoDB

This section covers Windows and Macintosh installations. For older versions and other platforms, please see documentation on <http://docs.mongodb.org/manual/> and select your preferred platform under the Installation tab.

2.1 MongoDB Windows Setup

2.1.1 Installation

1. Install MongoDB from <https://www.mongodb.org/>
2. Follow the instructions to finish installation of MongoDB.
3. Open Windows Explorer and navigate from Computer
 \\Local Disk:\Program Files\MongoDB\Server\3.0\bin\

Mongod.exe = actual mongo database; program that runs in background to allow your computer to run as database.

Mongo.exe = application / command line interface with Mongod.exe database

4. In Window Explorer navigate to Computer and your root on the Local Disk. Then create a New folder called "data" and inside this folder create another called "db"

2.1.2 Running Mongodb

1. Open Start Menu and type into the search "cmd.exe" to open the windows command line.
2. Change directories to the bin folder of the mongo installation:
 `cd Local Disk:\Program Files\MongoDB\Server\3.0\bin\`
3. Now that you are inside the "bin" folder which houses the Mongo files, type "mongod" into the command line and press "Enter".

4. A list of lines should run down the screen until the last row says "waiting for connections on port 27017"

This means that setup was successful and your application is now running in the background of your computer

2.2 MongoDB Mac Setup

2.2.1 Installation using Homebrew

Homebrew is a package manager and is a simpler way to install MongoDB. To install Homebrew, visit <http://brew.sh> (specifically for installation options), or type the following command in your Terminal window:

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

2. To install MongoDB, type the following:

```
brew install mongodb
```

2.2.2. Manual Installation

1. To manually install MongoDB on your Mac OS, visit <http://docs.mongodb.org/manual/tutorial/install-mongodb-on-os-x/> for instructions.

2.2.3 Running MongoDB

1. A data directory must be created as the root administrator to have the correct permissions. Create a data directory by typing the following into your Terminal window:

```
sudo mkdir -p /data/db
```

2. Set permissions for directory by typing:

```
sudo chmod 777 /data/db
```

3. To run type: *mongod*

2.3 MongoDB Database Creation

1. After completing setup and your mongod is running in the background, you may begin running codes to work with databases

2. In a new (second) command line window that is focused at the Mongo files folder, enter mongo to connect to enter the "test" database

- a. Typing db should result in a printing of "test" to the screen
- b. Note that data is stored by categories not tables

3. Nodejs

3.1 Installation

1. For Macintosh and Windows installation, visit the following link:
<https://nodejs.org/download/>
2. Choose your preferred version and run the installer.
3. To check if installation worked, open Terminal/Command Prompt and type: `node -v`

Amazon Web Service (AWS) Installation

1. Setup Ubuntu Server

This assumes the installer has knowledge of AWS services: EC2, VPC, Internet Gateways, Subnet groups, Network ACL's, Route tables, and Key-Pairs.

Follow the instructions at the following link. It is not necessary to install Redis.

Stop when you get to the "Clone Your App and Install npm and Bower Packages"

<https://www.codefellows.org/blog/how-to-host-a-nodejs-app-on-an-ec2-ubuntu-server>

2. Git Clone the Repository

1. Login to the server.
2. Ensure you are located in `/home/ubuntu`
3. Run `git clone` <https://github.com/asu-cis440-summer/career-center-MEAN.machine.git>
4. Change directory to `/home/ubuntu/career-center-MEAN.machine/bcc`
5. Run `grunt`
6. Point your browser to the ip address of the instance on port 3000
Example: <https://x.x.x.x:3000>

Where x.x.x.x is the IP Address of your server instance. This will be located in the AWS EC2 console.

3. Run the Application as a Service

Forever is an NPM package that will run the server forever, or until it crashes/reboots.

1. Login to your instance
2. Change directory to `/home/ubuntu/`
3. Run `sudo npm -g install forever`
4. Update your `$PATH`
`export PATH=$PATH:/usr/local/bin`
`export NODE_PATH=$NODE_PATH:/usr/local/lib/node_modules`
5. Change directory to your application
`/home/ubuntu/career-center-MEAN.machine/bcc`
6. Run forever
`forever start server.js`

You should see the following message:

```
warn:    --minUptime not set. Defaulting to: 1000ms
warn:    --spinSleepTime not set. Your script will exit if it does
not stay up for at least 1000ms
info:    Forever processing file: server.js
```