

# Homework #9 (MEAN) - Group + Individual

## Part 1

### Task 1

This part is for you to practice developing locally on your machine. You have two options:

- Use vagrant (with virtualbox).
- Install a VM on VirtualBox.
  - Any flavor is fine. Ubuntu 14.04 or 16.04 is recommended.
  - Mint Mate 17.3 (<https://linuxmint.com/edition.php?id=206>) is recommend. It is based on Ubuntu 14.04 and it comes with VirtualBox tools installed, so you don't need to struggle with this.
- **NOTE: For the most part, OS should not make too much of a difference as long as you are consistent with node versions. Recommended node version is 6.11.4, which is the current LTS version.**

### Task 2

Sign up for a Bitbucket account.

- <https://bitbucket.org/>
- We want you to practice working in a team using a repository (rather than on a production machine).
- You can create a team of up to 5 users (the current max group size) for free
  - <https://bitbucket.org/product/pricing?tab=host-in-the-cloud>
- You also get 'Unlimited private repos'.
  - We want your group work to be private just for you and your team. GitHub does not allow this by default.
- **NOTE: Every team member should have at least 5 'significant' commits (Fixing a typo is not a 'significant' commit). In other words, it is generally recommended that you commit every time you add a new feature or fix a but. We will check this.**

## Part 2

This section is to learn to use the MEAN stack by creating a 'blog type' application.

**NOTE: Development should\* take place on your local machine, however, similar steps will be used when deploying to production.**

## Task 1

Install MongoDB: <https://www.digitalocean.com/community/tutorials/how-to-install-mongodb-on-ubuntu-14-04>

**UPDATE:** Use the link below and follow based on your Ubuntu version:

<https://docs.mongodb.com/manual/tutorial/install-mongodb-on-ubuntu/>

## Task 2

- Not sure that this works very well in Windows, but should work on Linux or Mac.
- Setup a virtual environment using nodeenv.
  - Nodeenv allows you to isolate your installation of NodeJS and all it's application in a folder.
- <https://github.com/ekalinin/nodeenv>
- Use the LTS version of nodejs
  - <https://nodejs.org/en/>

## Task 3

- Install the MEAN stack and launch it: <http://meanjs.org/docs/0.4.x/>
- Import your code into Bitbucket.

## Task 4

MEANJS comes with an 'articles' application. You are to extend this with the following.

- Model for articles
  - id: unique identifier for the articles.
  - slug-field: Human readable unique identifier for the articles.
    - **NOTE: This field should be unique.**
  - content: Text of the article.
  - tags: **[array]** A list of tags that the user enters when creating the article.
  - author: Who created the article.
  - views: **[int]** Number of times the article has been viewed.
  - comments: **[int]** Number of comments for this article.
  - created: **[datetime]** Timestamp when article was created.
  - updated: **[datetime]** Timestamp when article was updated.
- Model for comments
  - id: unique identifier for the comment.
  - article: id of the article this comment belongs to.
  - author: author of the article.
  - comment: Text of the comment.
  - created: **[datetime]** Timestamp when the comment was created.

## Task 5

Create/Modify the following pages.

- create
  - This page allows a user to create an article.
  - Page should only be shown if they are logged in.
  - Only appropriate fields should be shown.
- update
  - This page allows a user to update an article (i.e. user must be logged in to see this).
  - **NOTE: Only the original creator should be able to update it.**
  - Only the appropriate fields should be editable.
- list (**landing page for the site**)
  - For this, you probably want to add 'lodash' to your stack to use functions like '\_chunk'
    - <https://lodash.com/docs>
  - Divide your page into 2 (content + sidebar on the right).
  - Shows a list of all articles in a grid view (3 rows) on the content side.
    - Each cell includes the title, views, few characters from the body, author and date created.
    - Articles should be ordered by most recent article first.
  - A searchbar that searches the articles (by title).
    - **NOTE: Search should be done client side.**
  - Sidebar should show unique tags from all the articles.
    - Clicking on a tag should show only articles with that tag.
    - **NOTE: This should be done client side.**
  - A button at the top to reset all articles.
  - A button at the top (if they are logged in), to create a article.
- view
  - Shows the article.
  - **UPDATE: You can remove references to delete, if you want. [\*\*\*Discussion in class].**
  - Shows all the information about the article.
  - Shows the comments at the bottom (most recent first).
    - Form entry to add a comment at the top of comments (bottom of article) (if they are logged in).
- delete
  - Take out any references for deleting an article.

## Task 6

- Brand your site.
  - Add a 'cool' name for your site.
  - Remove links and such that are not needed.
  - Make your site presentable.

## Task 7

Deploy your application on your machine.

**NOTE: We will discuss this on Wednesday's class.**

Deployment instructions:

```
1 # Prerequisites
2 sudo apt-get install linux-headers-
3 sudo apt-get install build-essentials
4
5 # Use root for below commands.
6
7 # Switch to /opt/ directory.
8 cd /opt/
9
10 # Create directory called meanjs and cd into it.
11 mkdir meanjs
12 cd meanjs
13
14 # Install nodeenv if you have not already and create a node 6.11.4 environ
15 # ment.
16 pip install nodeenv
17 nodeenv --node=6.11.4 env
18
19 # Activate the nevironment.
20 source env/bin/activate
21
22 # Install meanjs (prerequisites): http://meanjs.org/docs/0.4.x/
23
24 # Clone your application.
25 git clone url-of-repo mean
26 cd mean
27 npm install
28
29 # Install nodemon and forever globally.
```

```
29 npm install -g nodemon
30 npm install -g forever
31
32 # [NOTE: Before upload, disable liverelaod.js]
33 vim modules/core/server/views/layout.server.view.html
34
35 # Comment out this line: <script type="text/javascript" src="{{host}}:3572
36 9/livereload.js"></script>
37
38 # Open up port 3000 on your firewall.
39
40 # Start the node server.
41 nodemon --debug server.js
42
43 # Go to your site at port 3000 (http://ta2.dhcp.nd.edu:3000)
44 # If that works, then you can start your site permanently with:
45 forever start server.js
46
47 # Restart/Stop commands.
48 forever stop server.js
49 forever restart server.js
```

## Task 8

Send your URL with your team's netids to [gmadey@nd.edu](mailto:gmadey@nd.edu) and [qzhi@nd.edu](mailto:qzhi@nd.edu), with subject line CSE 40613 - group-number (i.e, your machine's hostname) - HW9