Table of Contents

[GETTING STARTED AND INSTALLING 1](#_Toc494833127)

[SETTING UP THE SERVER SIDE 2](#_Toc494833128)

[SETTING UP THE CLIENT SIDE 2](#_Toc494833129)

[USING/NOT USING BOOTSTRAP 3](#_Toc494833130)

# GETTING STARTED AND INSTALLING

* **DUPLICATE TEMPLATE:** Go to /desktop/dojo\_projects/mean/full\_mean, copy/duplicate the folder/project: “full\_mean\_template\_with\_login\_reg”, rename it whatever you want
* **JSON FILES:** go to the terminal and install the necessary npm/bower components, but first you create a package.json and bower.json file so that if you delete your node\_modules or bower\_components file you can just type “npm install” or “bower install” to have everything re-installed. The commands are listed below:
* **NPM MODULES INSTALLATION:** Start by typing “**npm init - y**” to create a package.json file, hit enter repeatedly. Next you install the required npm modules, the most common modules and their commands are listed below, you may or may not need all of these, make sure to type “--save” in each one so that that module is put into the package.json file, see below:

**npm install --save express**

**npm install --save express-session** (only needed for session use)

**npm install --save body-parser**

**npm install --save mongoose**  (only needed for database word)

**npm install --save bcrypt**  (only needed for password hashing)

**npm install --save connect-mongo** (only used for session use)

**npm install --save ejs**  (only use ejs for other html pages besides partials)

**npm install --save socket.io**  (only used for sockets)

* **ADJUST SERVER.JS FILE ACCORDING TO YOUR MODULES:** Go to your server.js file and comment/uncomment certain lines according to the modules you need. For example, if you’re using sessions, uncomment the session and MongoStore requires lines at the top as well as the code block below it (and change the “NAME\_HERE” part of the url).
* **BOWER COMPONENTS INSTALLATION:** NOTE - for all of these commands, I currently need to add “sudo” before them and “--allow-root” after them due to how things were installed, if not necessary, the simply take that part out. Start with “**sudo bower init -y --allow-root**” to create your bower.json file. Next, install and save with “--save” the bower components you’ll need, I’ve listed the most common one’s I use below, you may or may not use them all:

**sudo bower install --save angular --allow-root**

**sudo bower install --save angular-route --allow-root** (only used with angular-routes)

**sudo bower install --save angular-cookies --allow-root** (only used with cookies, typically used with sessions)

**sudo bower install --save angular-messages --allow-root** (only used with angular messages)

* **CONFIGURE INDEX.HTML ACCORDING TO YOUR BOWER COMPONENTS:** Include the below scripts to your index.html head, only if you installed the corresponding bower component:

**<script src ='angular/angular.js'></script>**

**<script src="angular-route/angular-route.js"></script>**

**<script src = 'angular-cookies/angular-cookies.js'></script>**

**<script src = 'angular-messages/angular-messages.js'></script>**

# SETTING UP THE SERVER SIDE

* **MONGOOSE.JS:** If, you’re using a database, go to config/mongoose.js and change the name of your database in the dbURI part “DB\_NAME\_HERE”
* **MODELS:** If you’re using a login/register functionality or users, adjust your user.js file and model accordingly. If you’re not, then delete that file. Create a new model file by duplicating and renaming the template.
* **CONTROLLER AND ROUTES:** If you’re using a login/register functionality or users, adjust your routes.js and users.js file accordingly, if not, then you can delete the routes and that controller file. Create a new controller file by duplicating and renaming the template.

# SETTING UP THE CLIENT SIDE

* **APP.JS AND PARTIALS:** After making sure you’re index.html file is good, go to your app.js file. If you’re not using angular routes or cookies, then remove those modules, otherwise add any other bower components in your angular module. Also, if you’re using a login/register functionality or users, adjust your routes and the corresponding partials, if not, then you can delete the routes and those partial files.
* **CONTROLLERS:** If you’re using a login/register functionality or users, adjust the loginRegController accordingly, if not, you can delete that file. Create a new controller by duplicating and renaming the template. If you’re not using the $http, $routeParams, $location, and $cookies services then delete them accordingly, otherwise add any new services/factories.
* **FACTORIES:** If you’re using a login/register functionality or users, adjust the loginRegFactory file accordingly (should also change the name of the cookies, to something more specific, otherwise it may see another cookie with the name ‘id’), if not then you can delete that file. Create a new factory by duplicating and renaming the template. If you’re not using the $http, $routeParams, $location, and $cookies services then delete them accordingly, otherwise add any new services.

# USING/NOT USING BOOTSTRAP

* **WHAT THE TEMPLATE COMES WITH:** The template comes with the 3 folders in “client/static”, bootstrap, fonts, and js. These files contain the necessary info for bootstrap. The index.html comes with two stylesheet links at the top of the page for boostrap, and two scripts at the bottom of the body for the boostrap js
* **IF YOU’RE NOT USING BOOTSTRAP:** If you’re not using boostrap, then delete the folders: bootstrap, fonts, and js from “/client/static”. Then go to index.html and delete the top two stylesheet links to “/static/bootstrap/….”, and the bottom two scripts at in the body.
* **IF YOU’RE USING BOOTSTRAP:** You can leave everything the way it is, it’s setup for bootstrap already.

END