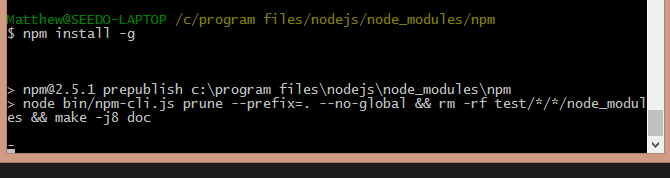
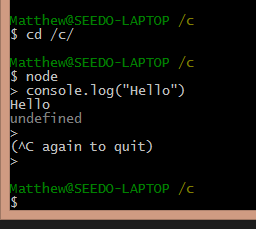
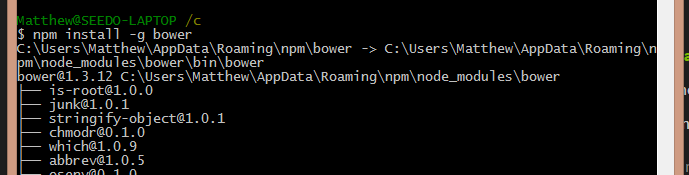
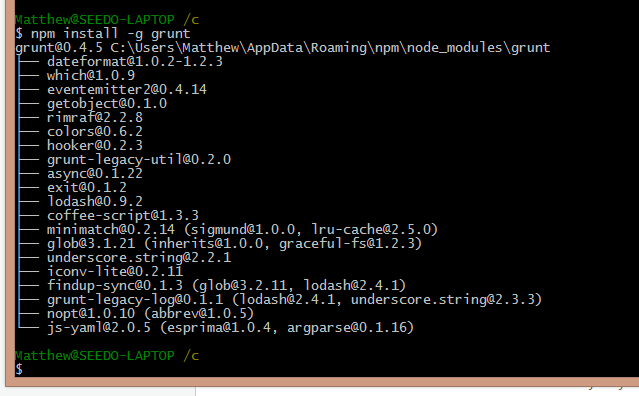
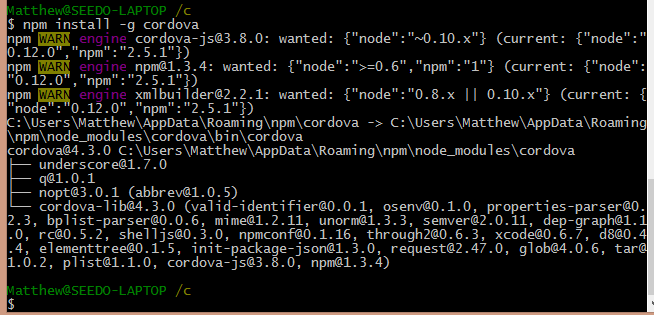
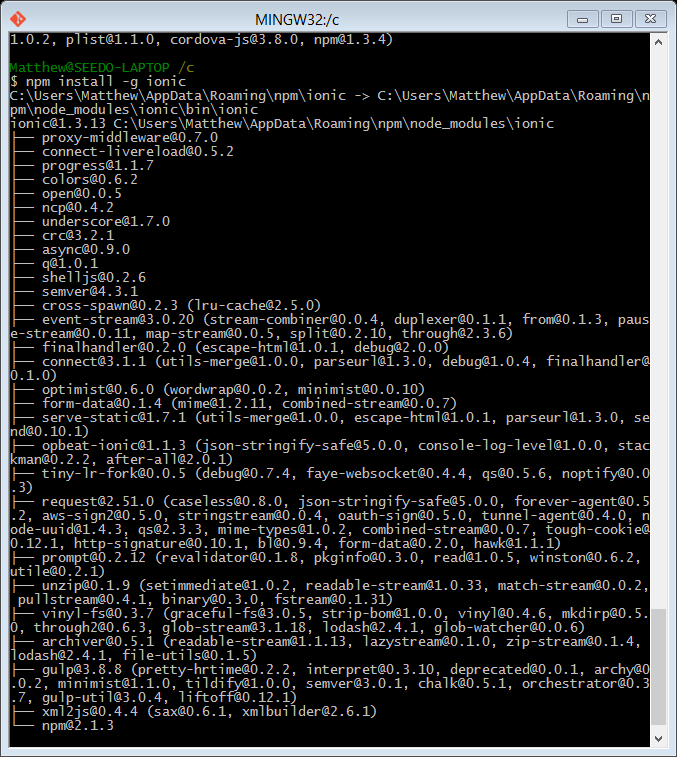
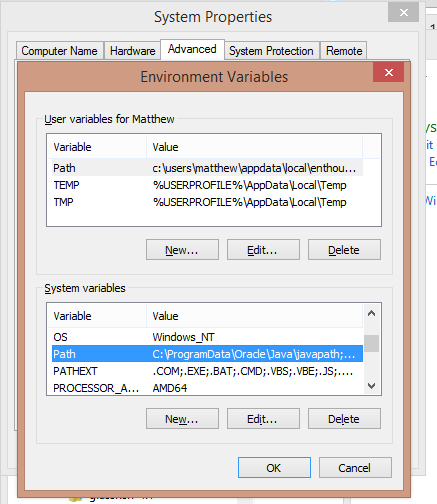
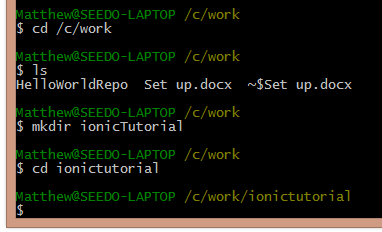
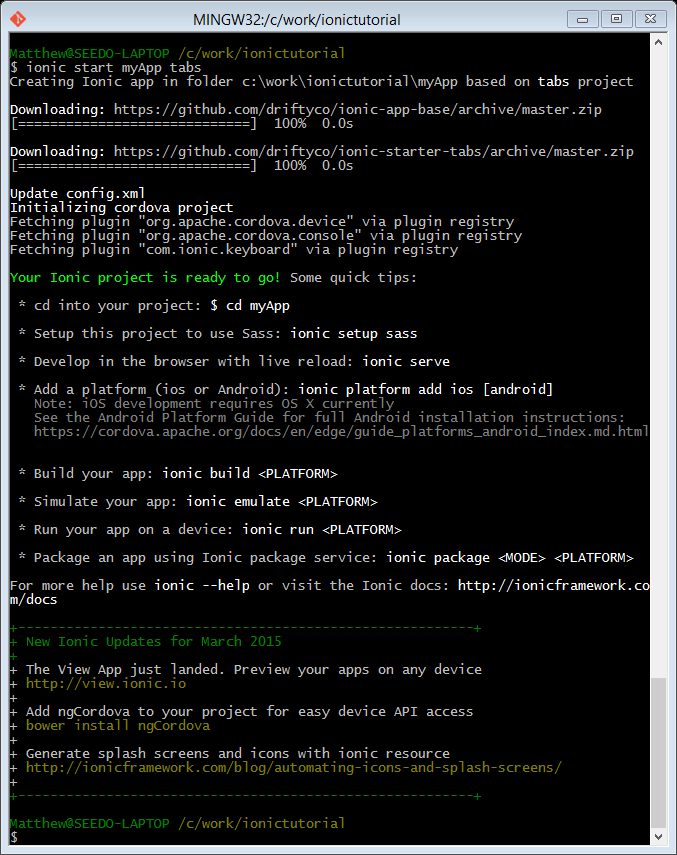
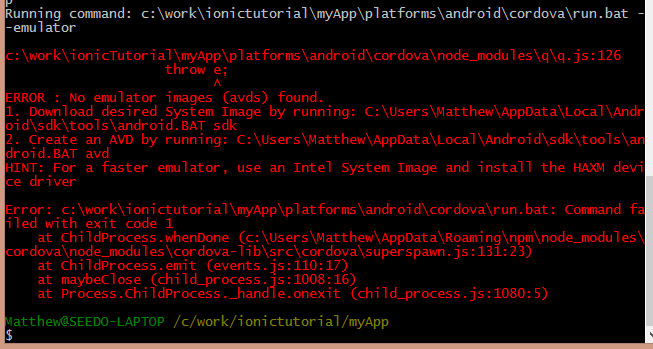
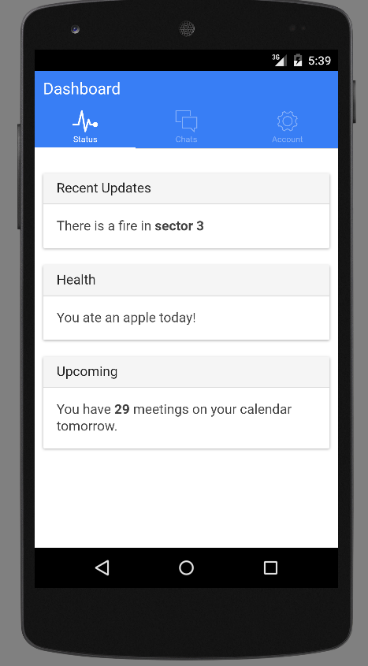
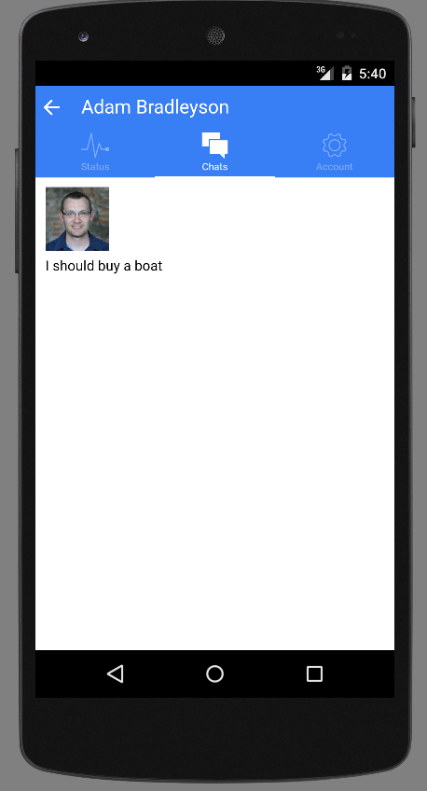
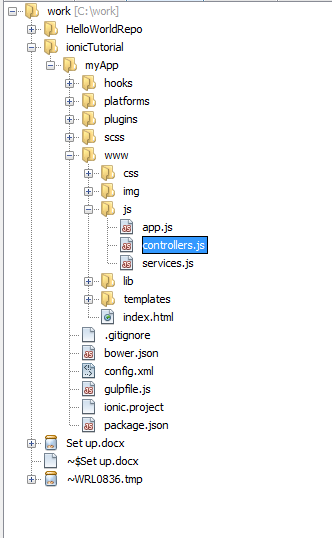
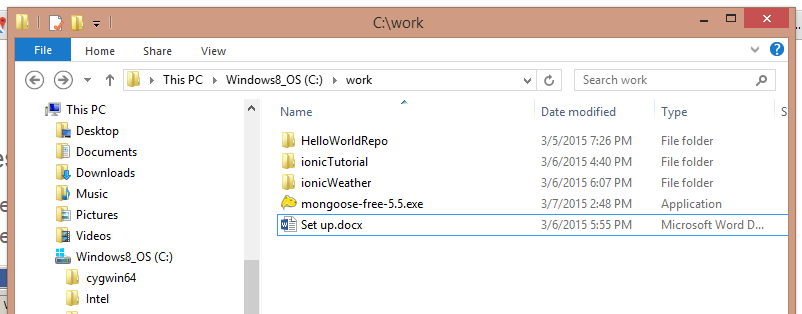
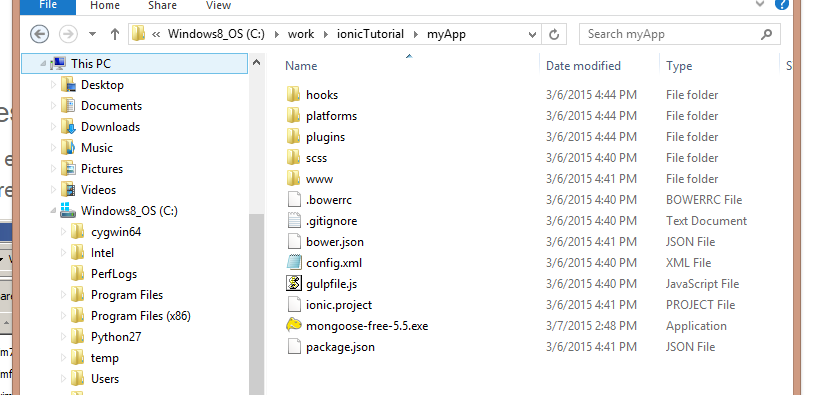
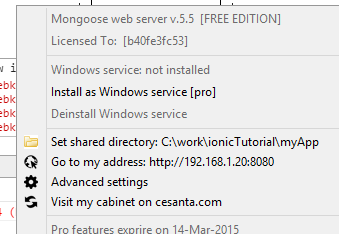
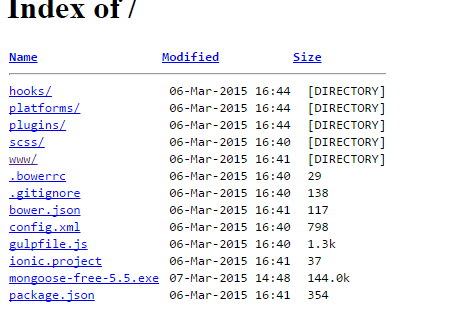
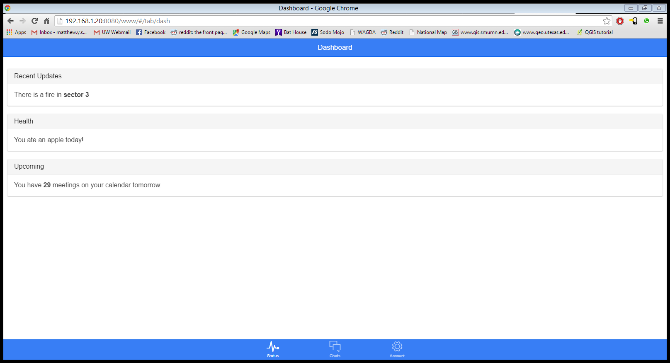
Set up

**So, you want to develop apps with Angular.js and Cordova/phonegap….here’s what you need to do**

1. Download Netbeans 8.0.x Java EE edition
   1. <https://netbeans.org/downloads/>
2. Create a GitHub account
   1. Facebook for programming
   2. We can share source code
   3. <https://github.com/>
   4. Look me up: <https://github.com/setom>
3. Download Git Bash (command line git tool)
   1. Don’t get the GUI client, I’ve never used it
   2. <http://git-scm.com/downloads>
4. Download the Android SDK
   1. This is needed to build apps
   2. I think we only need this to get access to the AVD manager
   3. Use this tutorial: <http://spring.io/guides/gs/android/>
      1. NB: Android studio is huge, and I don’t know if we’re using it…Ideally we could just get the AVD Manager…I’ll investigate
         1. Maybe we can develop in android studio instead?
5. Set up JSNode, Bower
   1. Download JSNode
      1. <https://nodejs.org/>
   2. Open Git Bash
   3. Navigate to the Node folder through Git Bash (command line)
   4. Type “npm install -g”
   5. 
   6. This should install the dependencies globally
   7. After the install, you can verify it was properly done by going to “cd /c/”
   8. “node”
      1. This should open a command line node editor
   9. “console.log(“Hello”)
      1. This should output Hello and Undefined
   10. Ctrl+C TWICE to exit
   11. 
6. Download and install bower library management
   1. From Git Bash Type “npm install –g bower”
   2. 
   3. This will install the bower dependencies
7. Download and install the grunt task runner
   1. From Git Bash type “npm install –g grunt”
   2. 
   3. This will install grunt globally
8. Download and install the cordova tool
   1. Cordova is used to “native-ify” apps
   2. The cordova wrap will populate a web browser on the mobile device that will make apps appear to run as native, when they are actually html 5 wrapped
   3. From Git Bash run “npm install –g cordova”
   4. 
   5. You may get warnings, don’t worry for now
9. Download and install the ionic tool
   1. Ionic is a framework that is styled to build native looking apps
   2. From Git Bash run “npm install –g ionic”
   3. 
   4. This should install ionic globally
10. Download Apache-Ant
    1. I have no idea what this is
    2. It’s from here: [http://learn.ionicframework.com/videos/windows-android/#](http://learn.ionicframework.com/videos/windows-android/)
    3. Download and unzip it to somewhere you’ll remember (I put it in program files)
    4. Navigate to the /bin folder
    5. Open Control Panel and search ‘Environment Variables’
    6. Edit the PATH
    7. Add the path to the /bin folder (check the semi colons)
       1. Ex: C:\Program Files\Apache Ant\apache-ant-1.9.4-bin\apache-ant-1.9.4\bin
    8. OK – OK – OK
11. **At this point we should be ready to build our first app!**
12. Make a folder under /c/ for all of our projects, having it in one spot will make source control simpler
    1. I made mine under /c/work
13. From Git Bash, navigate to cd /c/work
    1. Mkdir ionicTutorial
    2. This will create a new directory for our ionic app
    3. Then navigate into the directory
    4. 
14. Lets make our app in here:
    1. Basic directions taken from <http://ionicframework.com/getting-started/>
    2. Ionic start myApp tabs
       1. This will call ionic, start a new project, name it myApp and use the default TAB style
    3. It should download the github master TABS archetype
    4. 
15. **Lets run that archetype app!**
16. Type “ls” to see the contents of the folder
    1. It should have a new folder myApp
    2. Navigate into it
17. From inside the new myApp folder, run the ionic build commands
    1. NOTE: unless you’re on a mac, you can only build android
    2. “ionic platform add android”
       1. Add android as a platform we will be using, will install android dependencies
    3. “ionic build android”
       1. Take all the source files and build me an android app
    4. “ionic emulate android”
       1. Emulate my app!
18. **SHIT WTF HAPPENED???**
    1. 
    2. It looks like there aren’t any Android Virtual Devices to use for the emulator (AVDs)
    3. There are two options here:
       1. Make one using the android studio AVD manager
          1. <http://developer.android.com/tools/devices/managing-avds.html>
          2. Nah.
       2. Make one from the command line
          1. <http://developer.android.com/tools/devices/managing-avds-cmdline.html>
          2. Do it, make a generic android emulator
          3. (NB: when I started androd studio for the first time, it looks like it auto makes one for me)
          4. (NB: I had problems with the HAXM accelerator…workaround was to create an AVD using ARM rather than x86)
          5. NB: The emulator is slow as fuck
             1. Potential workaround with Mongoose or TomCat7
             2. Try to only open it once and keep it open while you code rather than reboot it all the time
19. SHIT WORKS YO:
    1. 
    2. You should also be able to click around the app and navigate it
       1. NB its slow as fuck
    3. 
20. **Lets take a look at our code now:**
21. Open Netbeans 8.0.x
    1. Open the files view using Window-Favorites
    2. Right click in the new window, and add a new favourite (/c/work)
    3. You should be able to see our projects now:
    4. 
    5. The app.js, controllers.js and services.js are going to be our angular.js code files
    6. Later we can add views (HTML) that we can navigate through
22. Adding Mongoose to run web apps
    1. This will help us to avoid running the emulator every GD time
    2. <http://cesanta.com/docs/FileSharing.shtml>
       1. Go to the site and download Mongoose (free)
    3. Once it’s downloaded put a copy in your work folder
       1. Each time you make a new app, you will need to put a copy of the mongoose.exe in it, so keep a master copy
    4. 
    5. Now, Put a copy of it in your ionicTutorial app that we made earlier
    6. 
    7. Double click mongoose and it will start a servelet on your pc
       1. Open the Mongoose menu in the bottom corner, select go to address 192.xxx.xx.x.x…xx
       2. It will pop open a servelet for your app!
       3. 
       4. Navigate to your app through www
       5. 
       6. BAM: APP
       7. 

**Tutorials:**

For next steps I suggest hitting up some of these tutorials:

Shaping up with Angular.js

* <http://campus.codeschool.com/courses/shaping-up-with-angular-js/>
* This is reallllly hokey…but it teaches you the basics in a fun(ish) way

Phonecat tutorial

* <https://docs.angularjs.org/tutorial>
* This one is better, but it’s harder…so maybe do this one second.

Shaka-brah!