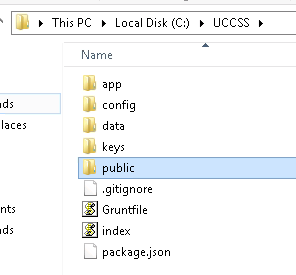
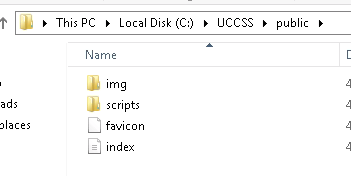
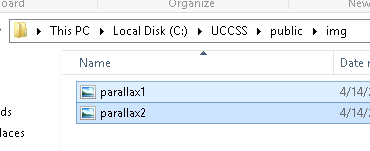
Download the repository from <https://github.com/rosshigh/uccss> and extract it. Create a folder called UCCSS and copy the contents of the repository Server folder into it. Create a folder called public in the UCCSS folder.



Drill into the repository Client2/aurelia-app folder and copy the img and scripts folders and the index.html and favicon.ico files into the Client2/aurelia-app folder into the public folder of the application.

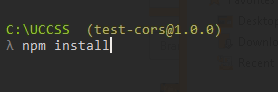


The two images in the img folder show on the landing screen and won’t work for you but, for now, just to get the site working you can use these. If you replace these, use the same names. We can talk about customizing the landing screen at some point. I’m not sure I want to leave the page like this. I was just playing around when I created it.



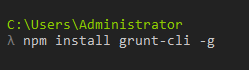
You run MongoDB and the app from a console. You can use Windows cmd or you can use another console application. I use Cmder (<https://github.com/cmderdev/cmder/releases/tag/v1.3.2>).

Open a console and navigate to the UCCSS/Server folder. Enter the command shown:

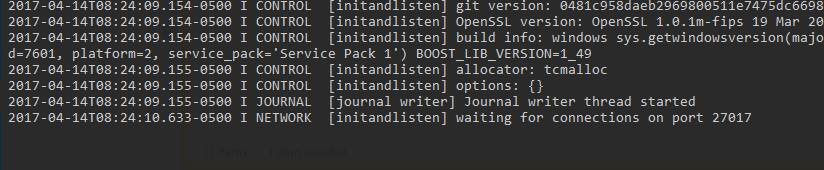


This will install all the dependencies for the server side of the application (I hope).

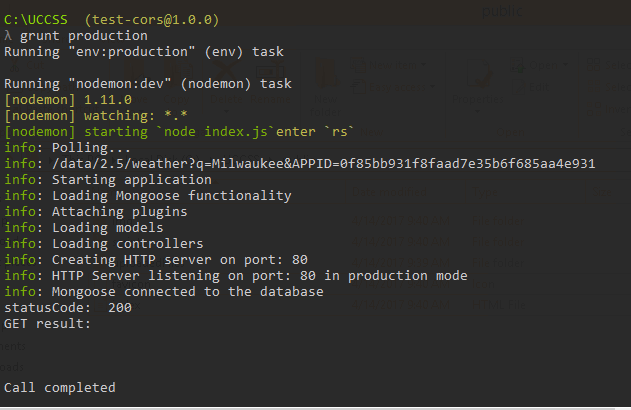
When that completes install Grunt (<https://gruntjs.com/getting-started>) with this command:



MongoDB creates its files in a folder at C:\data\db by default so create those folders if they don’t exist. Open a second console and enter the command **mongod**. If you get the message that mongodb is waiting for connections, then MongoDB is running.

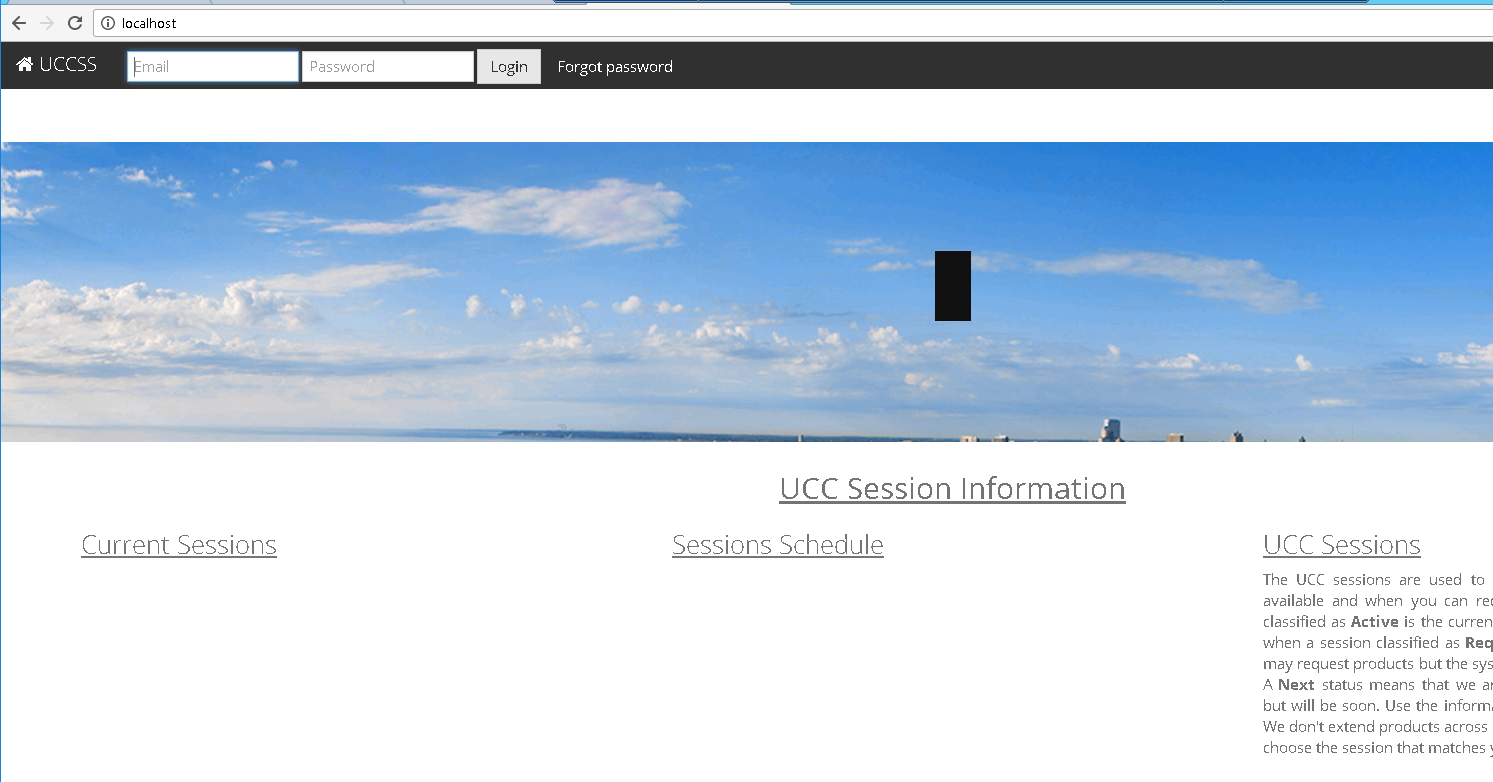


To run the application, use the command **grunt production** in the UCCSS folder.



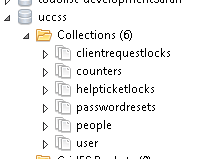
That’s my appid for openweathermap.org. You should register and get your own eventually but we can do that later.

Open a browser and enter the url of the server. If you are on the same machine as the application, enter <http://localhost>.

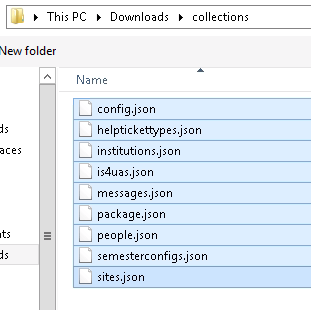


Next, you need to import some data into the database. There is a mongodb console but I use MongoChef (<https://studio3t.com/download/)>. Install MongoChef and start it. Click Connect and then create a new connection. Accept the default values and save the connection.

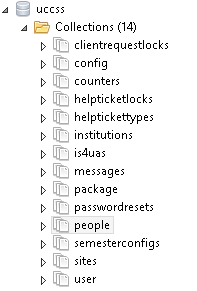
Once it connects, you’ll see there are several sample databases but there should also be a database called uccss. The collections you see may be different. The beauty of MongoDB is that if you attempt to access a database or collection and it doesn’t exist, it creates it for you.



Right-click the uccss database and select **Import collections…** Select JSON and click Next. Click the plus sign and then navigate to the data/collections folder in the repository. Select all the files:

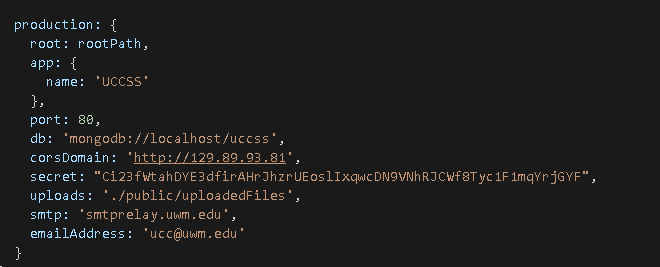


Click Open and then finish the wizard.



I only included Chico State and Gino. You can log on with the same account Gino uses for the test system here.

One last thing to do is to use a editor to open the config.js file in the config folder and edit the production parameters. The only thing you should edit is the smtp and emailAddress.



Let’s see if we can get this far.