**About This Directory:**

The djangoFrontend directory contains the support and content files needed to host a website using Python’s Django framework. As I create frontend content for the Board DB Compare site, I will add to this directory.

I am working loosely off this sentdex tutorial: <https://pythonprogramming.net/django-web-development-with-python-intro/>

**Dependencies:**

* Django (run the below command to get it)

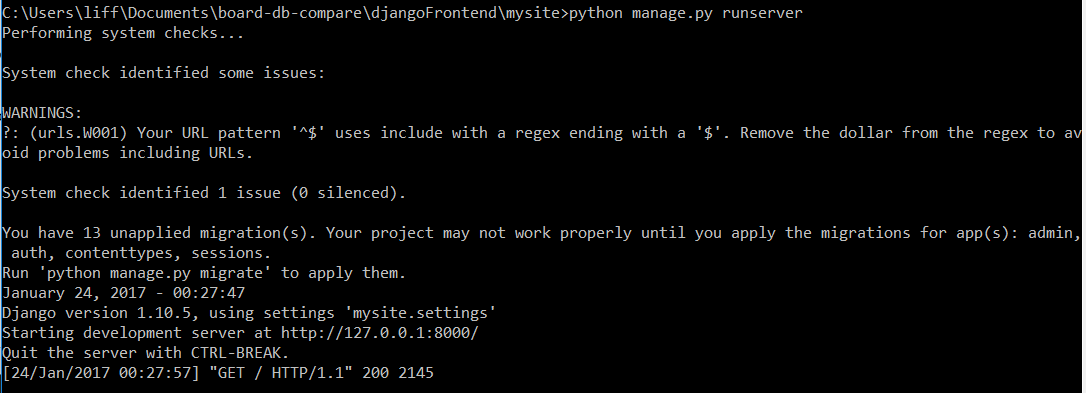


For CSS I am using Bootstrap (subject to change, but I really like it). Info about CSS classes available for use here: <http://getbootstrap.com/css/>

**Django Development Server:**

You can run the Django development server and test the current version of the site at the root url <http://127.0.0.1:8000/>

First, navigate to the mysite folder and run manage.py’s runserver function. You’ll get something like this for now (migration error will be fixed soon):



Refreshing the page in a browser resulted in the get request in the last line. Any action on the server can be monitored here.

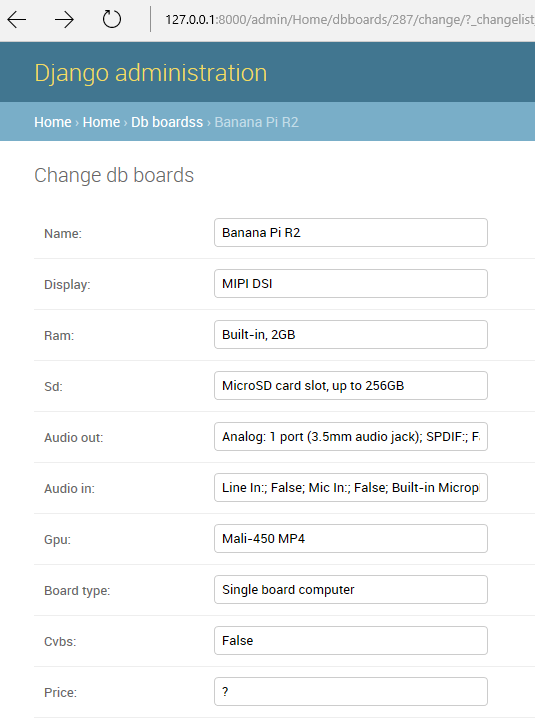
To close the development server, press CTRL+C.

So that’s about all I have so far. Content soon to come, stay posted ☺

**Django Admin Site:**

Django offers an administrator site that is useful for managing backend site data that is stored in SQLite database(s).

You can view and manipulate the data that is currently stored in whatever models you currently have registered with the admin site (see the mysite admin.py file for an example of how to register a model). Below is a screenshot of part of the edit page for a record in the dbBoards database:



While you have the development server running you can access the admin site by going to <http://127.0.0.1:8000/admin/>. The username is “admin” and the password is “Administrator.”

**Populating the dbBoards Database:**

To update the list of boards and their attributes, which is stored in the dbBoard database model, navigate to the mysite folder and run manage.py’s populate\_db command, then wait for the boardDB.getAttributes() method to run and the database to be populated with the result (it takes about a minute):

