Documentation for CRIMSON ALERTS, A CS50 final project by Juliet Nelson.

Accessing the site:

This URL will take you to the sign up site for Crimson Alerts: https://frozen-peak-9783.herokuapp.com/ The CSS for this is adapted from the Harvard Crimson in order to integrate with their site, since this project is ultimately designed for use by The Crimson Staff.

Signing up:

Once you are on the site enter the demographic information requested then click submit. The form should either prompt you to fix mistakes or take you to the confirmation page indicating that you have sucessfully signed up for Crimson Alerts! You should receive a text message from the number "(267)875-2746" confirming your sign up and welcoming you to Crimson Alerts. To enter another subscriber simply reload the page.

Unsubscribing:

To unsubscribe without using admin privilidges respond STOP to the Crimson Alerts number ((267)875-2746). You can also delete yourself from the database, but that takes administrative privilidges.

Administration:

To access Administration of Crimson Alerts visit https://frozen-peak-9783.herokuapp.com/admin and use the admin username:janelson12 and password: juliet

Adding Breaking News:

Breaking news can be added by an administrator by clicking the news link in django admin and then the add news button on the top right hand corner. One will then be prompted for the headline and URL. The Alert Sent field should be left unchecked (it will automatically be filled in when an alert is sent).

Sending Alerts:

Heroku is set to run the send alerts app every 10 minutes. Any new breaking news added in the interum will be sent automatically.

Installation:

NOTE: These instructions adapted from internal Crimson instructions for thecrimson.com.

Software requirements:

- Python 2.7
- pip (Python package installer)
- A bunch of Python packages listed in requirements.txt
- MySQL

The Crimson recommended installing Python on Mac OS with Homebrew (http://brew.sh) since the default version installed on Mac doesn't have pip. Then you can just install all the required Python packages with

```
pip install -r requirements.txt
```

Then you have to create a MySQL database. I called mine "final50".

Then to get this thing up and running you have to set three environment variables, DATABASE_URL, TWILIO_ACCOUT_SID, and TWILIO_AUTH_TOKEN. The DATABASE_URL is parsed by dj-database-url so make sure it's something dj-database-url understands (https://github.com/kennethreitz/dj-database-url). The TWILIO environment variables should be set according to your Twilio account for sending SMS. I included mine in the file ".env" so you guys don't have to create an account—just don't go crazy cause each one costs $$$!

Django automatically creates the tables so you just have to run

```
./manage.py migrate
```

to create a blank database. I didn't bother including a SQL file of my database because this is so easy.

To start the server just do

```
./manage.py runserver
```

and it'll start a server at http://localhost:8000. django has all the documentation here:https://docs.djangoproject.com/en/1.7/ref/django-admin/

If you want to log into admin (http://localhost:8000/admin) just create a "super" user with

```
./manage.py createsuperuser
```

and then just fill out all the stuff it wants.