Homework 6: Deployment

Due date: Feb 29, 2016

This homework is the fourth (and final) in a series of homeworks in which you will build an increasingly sophisticated nano-blogging site. This site will now be a featureful, interactive web application including user registration and authentication, email integration for user verification, photo upload, and quasi-real-time updates.

For this assignment, you will extend your Homework 5 solution to add email validation of user accounts and you will deploy your application to the cloud.

The learning goals for this assignment are to:

- Demonstrate mastery of the learning goals from the previous homeworks, including both technical features of web applications and the development process.
- Gain familiarity with sending e-mail programmatically from your web application and generated links in the e-mail that return to your web application.
- Gain familiarity with a cloud deployment platform.
- Gain basic familiarity with database systems administration.

Specification

This section describes the enhancements you will make to your social network application. To begin your assignment, recall that you should start by copying over your project from the last homework over to the hw5 folder (see Turning in your work for the directory structure).

1. When registering a new account:
    - require that an email address to be provided,
    - send, via email, a link that can be clicked on to validate the email address.
2. Deploy the social network application on either Heroku or AWS EC2.

Requirements

Your application must also follow these requirements:

- You must meet all requirements specified in the previous assignments.
- You must fix all errors that the mentioned in your hw4 feedback (grades for hw4 will be posted well before this assignment is due).
- You must store your data in a relational database that is not SQLite.
- If you deploy on Heroku, you must store your images in Amazon S3.
- If you deploy on AWS, you may not use the development server. (You must deploy with Apache HTTP Server or something similar.)
- You may not store passwords or access keys in your application code (or in your repos).
- Cite all external resources used and any additional notes you would like to convey to your grader in the README.md file.

Assignment hints

1. If you are deploying on EC2, the recommended method is to run Django under Apache HTTP Server. Django documentation is:
    - https://docs.djangoproject.com/en/1.8/howto/deployment/wsgi/modwsgi/
2. If you are deploying on Heroku, the tutorial is:
    - https://devcenter.heroku.com/articles/getting-started-with-python
3. Data files (pictures) created in Heroku will disappear when your server becomes inactive. To store your pictures permanently, you should put your images AWS S3. The Heroku guide is here:
    - https://devcenter.heroku.com/articles/s3-upload-python
4. You should not include passwords and other access keys in your application code and you should not put them in your GitHub (or Heroku) repositories.
    - On EC2, put your access keys in files that will not be stored on a server. Be sure such files are listed in your .gitignore file.
    - On Heroku, set environment variables for passwords and access keys (as is shown in the S3 documentation).

Grading criteria

For substantial credit your solution must clearly demonstrate the learning goals for this assignment, which are described above in the introduction.

Specification fulfillment

Your submission must follow all specifications and requirements introduced in the previous sections.

Coverage of technologies

You must demonstrate use of the technologies of this assignment, particularly:

- deployment of a Django application
- application email configuration

application database configuration

Turning in your work

You must provide the URL for your deployed application in the `hw6/README.md` file in your GitHub student repository. The TAs will not be able to visit your site if you don't do this.

If your deployment code is not being developed in your student repo, you must copy your code into your student repo and then submit it. The directory structure will look somewhat like the following (some files/directories omitted):

```
[YOUR-ANDREW-ID]/hw6/
    |-- README.md
    |-- webapps/
        |-- settings.py
        |-- urls.py
    |-- socialnetwork/
        |-- static/
        |-- templates/
        |-- models.py
        |-- views.py
    |-- manage.py
```