Yufan Xu

Exercise 2.8: Deploying a Django Project

Learning Goals

- Enhance user experience and look and feel of your web application using CSS and JS
- Deploy your Django web application on a web server
- Curate project deliverables for your portfolio

Reflection Questions

- Explain how you can use CSS and JavaScript in your Django web application.
  - CSS can be used to add styling to my application, making it more visually appealing and easy to navigate. Some examples include a navigation bar at the top of the page, a background image or color, and formatting of the images, buttons, and text. JavaScript can be used to add some interactive elements for user engagement, like maybe embedding instruction videos for recipes, adding an option for users to leave comments under recipes, and maybe "favoriting" or "following" another user to get updates for new recipes (ideas that I most likely won't go into for this project now but are ideas to think about in the future).
- In your own words, explain the steps you'd need to take to deploy your Django web application.
  - First, I would need to upload the code to a GitHub repository, taking care to remove the settings.py file (containing parameters that may pose a security risk). Then, I would need to configure my application for deployment using a service provider like Heroku. This includes creating a few new files (like to declare the process type and entry point of the app), updating some files (settings.py, requirements.txt, database configuration), and installing some packages to create a complete environment for the production environment on the server (a HTTP server,psycopg2). Then with everything ready, I can deploy it to Heroku (logging into the client, creating the application, uploading the application, setting up the database tables, creating a superuser, and lastly updating ALLOWED_HOSTS to allow the server access to the Django application.
- (Optional) Connect with a few Django web developers through LinkedIn or any other network. Ask them for their tips on creating a portfolio to showcase Python programming and Django skills. Think about which tips could help you improve your portfolio.
- You've now finished Achievement 2 and, with it, the whole course! Take a moment to reflect on your learning:
  - What went well during this Achievement?
    - I developed a good basic understanding of how the Django framework is structured and the use of urls and directing (as well as redirecting) users to different pages based on the views and templates.
  - What's something you're proud of?
    - I was able to add navigation to allow for easier access to the different views on the web application. I am also proud of my data visualization charts. I had ideas

for what I wanted to display in the charts that were a little different from the way the course/example was given, so it was difficult to pull the correct data (in the correct format) from the queryset and dataframe to generate the charts. I didn't want to settle with an "easy" option if it didn't represent something I considered useful for the application, and I am proud that I didn't give up on my idea.

- ➢ What was the most challenging aspect of this Achievement?
    - ■ The most challenging aspect of this was actually in the last stage - deployment. As I prepared the app for deployment, many issues occurred and it felt hopeless when the instructions in the course notes were not working out. I ended up deviating from the course notes and researching a new hosting platform that offered a free option for connecting with a Postgres database. Then there were issues with the media files, so I had to again research options to serve those files. It was challenging because it was really self learning.
    - ■ Another challenging aspect was in generating the data visualization charts, as the data I wanted to use wasn't easily available from the queryset (were not attributes that had its own column already). So it was hard to retrieve those values from methods within the model and then manipulate them to get the data I needed.
- ➢ Did this Achievement meet your expectations? Did it give you the confidence to start working with your new Django skills?
    - ■ The achievement met my expectations of seeing how a Django web application is structured, how the different files and pieces interact with each other, and how to incorporate user-uploaded media, authentication, and data visualization. Allowing users to upload images and protecting certain views from unauthorized users were 2 pieces that were surprisingly simpler than I had imagined it to be. I feel like there is so much more I need to learn, but this gave me a basic understanding that will lay the foundation.

Well done—you've now completed the Learning Journal for the whole course.