

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

1. Explain how you can use CSS and JavaScript in your Django web application.

CSS and JavaScript can be used in a Django web application by using a static folder that is referred to in the HTML templates as follows: `{% static 'recipes/css/styles.css' %}`

2. In your own words, explain the steps you'd need to take to deploy your Django web application.

Sensitive information must be cleared, a .gitignore file needs to be created to keep from pushing development environment info to production. Next a Procfile must be created, Gunicorn installed, and the application needs to be configured to the standards of the hosting service. From there, the settings.py file has to be adjusted to handle static files as they go to production. Whitenoise must also be installed. Finally, a requirements.txt file must be created to replicate the development environment for production. All application data then gets pushed to GitHub and then pushed to the hosting service.

3. (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.
4. You've now finished Achievement 2 and, with it, the whole course! Take a moment to reflect on your learning:
 - a. What went well during this Achievement?
 - b. What's something you're proud of?
 - c. What was the most challenging aspect of this Achievement?
 - d. Did this Achievement meet your expectations? Did it give you the confidence to start working with your new Django skills?

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