

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.*

In a Django web application, you can use CSS to style your web pages and JavaScript to add interactivity. Simply create CSS and JavaScript files in the static directory of your project, link them in your HTML templates, and write your styles and code accordingly.

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

Deploying a Django web application using Heroku:

1. Prepare Github Repository and Upload Code:
 - a. Create a new repository on GitHub to host your code.
 - b. Push your Django application code to the GitHub repository.
2. Update Django Application for Hosting:
 - a. Modify your Django application's settings to ensure compatibility with Heroku.
 - b. Configure the database settings to use Heroku's database service.
 - c. Update the static file and media file handling settings.
 - d. Ensure your application has a proper secret key and debug mode is turned off.
3. Deploy Using Heroku:
 - a. Sign up for a Heroku account at heroku.com.
 - b. Install the Heroku CLI (Command Line Interface) on your machine.
 - c. Open the terminal or command prompt and log in to Heroku using the `heroku login` command.
 - d. Create a new Heroku app using the `heroku create` command.

- e. Link your Heroku app to the GitHub repository using the `heroku git:remote -a <app-name>` command.
- f. Push your code to Heroku by running `git push heroku main`.
- g. Run any necessary database migrations using the `heroku run python manage.py migrate` command.
- h. Set up any additional resources required by your application, such as a Heroku Postgres database or cloud storage for static files.

3. *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?*

During this Achievement, I could grasp the concepts of Django and its Model-View-Template (MVT) architecture. Setting up the project structure and getting familiar with Django terminologies went smoothly. I also explored built-in features like user authentication and authorization, which helped enhance my understanding.

b. *What's something you're proud of?*

I can be proud of my progress in creating models, views, and templates for my Django applications. Additionally, integrating Python's matplotlib and pandas libraries for data analysis and visualization within my web application showcases my ability to leverage external tools effectively.

c. *What was the most challenging aspect of this Achievement?*

The most challenging part of this Achievement might have been deploying my Django web application using Heroku. Configuring the deployment environment and ensuring a successful deployment can be complex and require careful attention to various settings and configurations.

d. *Did this Achievement meet your expectations? Did it give you the confidence to start working with your new Django skills?*

Completing the deployment process using Heroku makes me more confident in deploying my Django projects and sharing them with others.