Exercise 2.2: Django Project Set Up

Learning Goals

- Describe the basic structure of a Django project
- Summarize the difference between projects and apps
- Create a Django project and run it locally
- Create a superuser for a Django web application

Reflection Questions

- Suppose you're in an interview. The interviewer gives you their company's website as an example, asking you to convert the website and its different parts into Django terms. How would you proceed? For this question, you can think about your dream company and look at their website for reference.

I would convert the website to a Django project by using its built-in application structures to build several app modules, configuration files, and a database.

I would name the project to reflect your company's name, and since this is a car rental company, I add several modules with specific functionality catered to your business needs.

Each section of the website be its own app such as:

- A client app for customer details
- A rentals app to track car rentals
- A vehicle app to list cars available
- An insurance app for insurance options.

I'd set up a database with models for users, vehicles, and rental history, and views and templates would be used on each website page and URLs would take care of the navigation.

- In your own words, describe the steps you would take to deploy a basic Django application locally on your system.

Here are the steps I would take to deploy a Django application locally:

1. I would create a folder and give it a name
2. I'd create a virtual environment and give it a name
    a. $mkvirtualenv project-name
3. Then I'd go to terminal and install Django
    a. pip install django
4. Next, I'd create a Django project
    a. django-admin startproject myapp
5. Then I'd start the server
    a. python manage.py runserver

6. Next, I’d create the app
    a. python manage.py startapp myapp
7. Then I’d set up the database
    a. python manage.py migrate
8. Next I would rename the root folder
    a. Get out of root folder and run this: mv project src
9. Then i’d create an admin superuser
    a. python manage.py createsuperuser
10. Next, I’d stop the server
    a. Control + C
11. Then I’d restart the server
    a. python manage.py runserver

- Do some research about the Django admin site and write down how you’d use it during your web application development.

I would use Django’s built-in admin panel to manage my website’s data. I would use it to do the following:

- Create a superuser to login
- Manage users by adding them, editing user passwords and access rights
- Edit the website’s content
- Work with the database directly
- Interact with Django models to perform CRUD operations without need to add custom code
- To generate an admin interface