

Exercise 2.1: Getting Started with Django

Learning Goals

- Explain MVT architecture and compare it with MVC
- Summarize Django's benefits and drawbacks
- Install and get started with Django

Reflection Questions

1. Suppose you're a web developer in a company and need to decide if you'll use vanilla (plain) Python for a project, or a framework like Django instead. What are the advantages and drawbacks of each?

Vanilla Python would be better suited for projects that are on a smaller scale and don't need all the built-in features that Django provides. Another advantage of vanilla Python is that you have more control over how your application works. However, because you have more control and don't have the features Django brings, your development time will be longer/slower.

Django on the other hand, isn't well-suited for smaller projects because of all the features it comes with because it makes it so server intensive. It also has specific ways of doing things the "Django Way," so you lose some control over how your application works. However, this leads to faster development times and lets you, as a developer, focus on other things.

2. In your own words, what is the most significant advantage of Model View Template (MVT) architecture over Model View Controller (MVC) architecture?

The most significant advantage that MVT has over MVC is that the Model-View is loosely or not coupled, which leads to a more modular architecture and therefore can make for a more scalable application.

3. Now that you've had an introduction to the Django framework, write down three goals you have for yourself and your learning process during this Achievement.

I want to be able to finish this achievement with a solid understanding of how Django works and how it compares to developing a web application with something like React. Because we used React so much in the Immersion course, I'm really interested in seeing how developing works with something new.