Yufan Xu

# 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

- ❖ 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?

    - ➢ The advantages of using a framework like Django include quicker development times, as it comes with many useful features and prebuilt functionality, and its scalability. If the project can make use of a good amount of the features and services offered by Django, and it is a large project (or projected to continue to grow in the future), Django will be a great option. On the other hand, a framework like Django is highly structured, which leads to some loss of control regarding how the application will run. So if we need or want control over the details of the project, vanilla Python might be a better option. Also, if the project is small, doesn't need a database, or will not make use of the features included, a framework like Django may not be needed either.

- ❖ 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 of MVT architecture over MVC architecture lies in the way user interactions are managed. With MVC, the developer needs to write the controller code that handles the business logic, which then displays the data in the View. With MVT, the View handles the business logic, whereas the Template takes care of the presentation of the data retrieved from the Model (database) and handles the controller output the developer would've had to write in MVC. The View and Model are loosely coupled in MVT so it is easy to modify and add components, making it suitable for applications of all sizes, whereas MVC isn't as suitable for small applications.

- ❖ 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. You can reflect on the following questions if it helps: 1) What do you want to learn about Django? 2) What do you want to get out of this Achievement? 3)Where or what do you see yourself working on after you complete this Achievement?

    - ➢ 1. Learn about the built-in features Django offer and how to apply them to a project

    - ➢ 2. Learn how to rebuild a project or transition an existing project into the Django framework

    - ➢ 3. Gain a better understanding (details) of how the coding/development process differs using Django versus vanilla Python to aid in determining which is more suitable for projects