

Django is based on MVT (Model-View-Template) architecture. MVT is a software design pattern for developing a web application.

MVT is explained below as the following:

- a. **Model:** The model is going to act as the interface of your data. It is responsible for maintaining data. It is the logical data structure behind the entire application and is represented by a database (generally relational databases such as MySQL, Postgres).
- b. **View:** A view can be based on a database, an application, or any other source of information, and it is typically separate from the code that receives and responds to HTTP requests. When you send an HTTP request to an application, the code that handles that request might be on the same physical machine as the application code. This might not be the case for a high volume of requests. For example, a web application that runs on a cloud infrastructure might have a single virtual host running on a single physical machine. In this case, the code that handles requests might not be on the same physical machine as the code that receives the requests.
- c. **Template:** An HTML template text file defines the structure or layout of a file (for example, an HTML page) with placeholder text representing actual content. A model can be used to dynamically populate an HTML page with data from a view, producing a view. A template may be used to define the structure of any kind of file, not just HTML.

Django also provides URLs mapping with its built-in features.

URLs: View functions can be used to handle HTTP requests in a more efficient manner by processing each resource individually. URLs, on the other hand, can be processed collectively via a single function. View functions may handle each resource individually by processing URLs individually. View functions may also receive data from a URL mapper that matches certain literal strings or characters in a URL and passes them on as data.