**Project Architecture**

* Model–View–Controller (MVC) is a software design pattern that separates internal representations of information.
* Django is based on a modified version of MCV architecture - MVT (Model-View-Template).
* Django web applications typically group the code that handles each of these steps into separate files, as seen in Figure 1:



Figure 1: Modified Django MVT Architecture

* **URLs:**While it is possible to process requests from every single URL via a single function, it is much more maintainable to write a separate view function to handle each resource. A URL mapper is used to redirect HTTP requests to the appropriate view based on the request URL. The URL mapper can also match patterns of strings or digits that appear in a URL and pass these to a view function as data.
* **View:** A view is a request handler function, which receives HTTP requests and returns HTTP responses. Views access the data needed to satisfy requests via *models*, and delegate the formatting of the response to *templates*.
* **Models:** Models are Python objects that define the structure of an application's data, and provide mechanisms to manage (add, modify, delete) and query records in the database.
* **Templates:** A template is a text file defining the structure or layout of a file (such as an HTML page), with placeholders used to represent actual content. A *view* can dynamically create an HTML page using an HTML template, populating it with data from a *model*.
* In this data-driven website, the web application waits for HTTP requests from the web browser.
* When a request is received the application works out what is needed based on the URL and possibly information in POST data or GET data.
* Depending on what is required it may then read or write information from a database or perform other tasks required to satisfy the request. The application will then return a response to the web browser, often dynamically creating an HTML page for the browser to display by inserting the retrieved data into placeholders in an HTML template.