

DJANGO REPORT

- What does this technology (library/framework/service) accomplish for you?
 - Django is an open-source framework used to make database-driven web development easier. It follows an MVC architecture. It is written in Python throughout and uses reusability, classes, and methods.
 - Django creates a layer of reusability and ease on top of TCP socket servers, HTTP protocols, database protocols, models, etc.
 - It also assists in content administration, SQL injections, security
 - It is very powerful and easily scalable due to its modularity and reusability
 - Django was created to extract a generic Web development framework that lets users build Web applications more quickly.
- How does this technology accomplish what it does?
 - Django was built using python. It uses the several inbuilt libraries, primarily the **socketserver** library
 - The socketserver library is a python library that is used for networking servers
 - It creates a TCP socket server that listens on a port for different kind of messages called events
 - Django is able to use a form of functional programming on top of event handling to handle HTTP requests and responses
 - <https://docs.djangoproject.com/en/3.0/topics/http/>
 - <https://github.com/django/django/blob/master/django/http/request.py>
 - <https://github.com/django/django/blob/master/django/http/response.py>
 - It uses a class called **multiparser** for parsing chunks of data sent across the TCP server
 - <https://github.com/django/django/blob/master/django/http/multipartparser.py>
 - The server is run from a file called manage.py when using Django
 - Manage.py calls the runserver protocol
 - <https://github.com/django/django/blob/master/django/core/servers/basehttp.py>
 - This is where the actual server is created and run
 - For paths, it maintains a list called urlpatterns which respond to a route with a certain action/method

- In our case, URLs are stored in a file called `urls.py` and it calls functions from `views.py` which in turn invoke the HTTP calls in the Django library
 - The URLs handler in Django is able to parse and handle URLs (e.g. validating a path).
- <https://github.com/django/django/blob/master/django/urls/conf.py>
 - This file above is where the function `path()` lives, which we use in `urls.py`. It, at the least, takes a path and a “callable” a.k.a callback function
- To render files and return them through HTTP, we take a look at `views.py` in our project
 - We invoke two imports, `render` and `HttpResponse`
 - `HttpResponse` comes from `response.py` linked above
 - `Render` comes from Django's `shortcuts.py`
 - <https://github.com/django/django/blob/master/django/shortcuts.py>
 - This renders a file into a string and uses the `HttpResponse` to send a response back to the client
 - When a page is requested, Django creates that `HttpRequest` object after which everything else takes place
- What license(s) or terms of service apply to this technology?
 - Django is licensed under a 3-clause BSD License.
 - This is an open-source license granting broad permissions to modify and redistribute Django.
 - It also uses the license for python which also allows the distribution under a permissive open source license
 - <https://github.com/django/django/blob/master/LICENSE>
 - This license allows for redistribution including commercial use.
 - For the purpose of our project, it is open for use
 - The license, however, states that” *Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer*”

