

Client server architecture: A model in which the client and server communicate over a network. From lecture 4, the client asks questions on behalf of users, and the server waits for and responds to questions. Server serves many clients.

In our Django application from GE02, the client is the web browser where users interact with the application, and the server would be the Django framework running on a server we created.

URL: Uniform Resource Locators are addresses used to access resources on the internet. Consists of HTTPS, a domain name, and a path to the resource

In the Django application, we were able to define the URL path in url.py. The URL patterns map to views, indicating which view should be called for a particular URL. I am still a bit unsure, but after following the directions provided in the GE, this part was not working. Only when I added the extra paths listed below, did it work.

```
from django.urls import path
from . import views

urlpatterns = [
    #path function defines a url pattern
    #' ' is empty to represent based path to app
    # views.index is the function defined in views.py
    # name='index' parameter is to dynamically create url
    # example in html <a href="{% url 'index' %}">Home</a>.
    path('', views.index, name='index'),
    path('', views.logout, name='logout'),
    path('', views.login, name='login'),
]
```

Urls.py from portfolio_app

```
from django.contrib import admin
from django.urls import path, include
from django.contrib.auth import views as auth_views

urlpatterns = [
    path('admin/', admin.site.urls),
    #connect path to portfolio_app urls
    path('', include('portfolio_app.urls')),
    path('login/', auth_views.LoginView.as_view(), name='login'),
    path('logout/', auth_views.LogoutView.as_view(), name='logout'),
]
```

Urls.py from django_project

HTTP requests: protocol used for transmitting data over the internet. HTTP requests are made by the clients and the server receives them to complete the request. Django uses request and response objects to pass state through the system.

In the Django project from GE02, when a user(me) accesses the local host, it triggers the https request. The url patterns in the screenshot above define which view function should handle the request.

```
portfolio > portfolio_app > views.py
1 from django.shortcuts import render
2 from django.http import HttpResponseRedirect
3 # Create your views here.
4 def index(request):
5     # Render index.html
6     return render(request, 'portfolio_app/index.html')
7
8
9 def login():
10     return render("Login")
11
12 def logout():
13     return render("Logout")
```

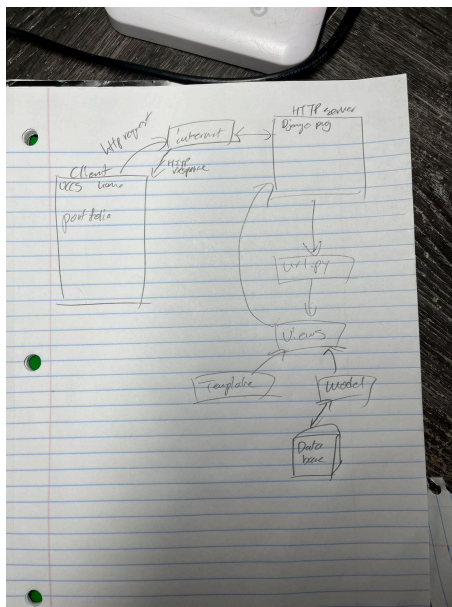
Views.py

Django MVT: Django's Model, View, Template architecture consists of the following.

1. Model is acting as the interface for the data. It is responsible for maintaining the data.
2. View is the user interface. This is what is visible to the end user on the browser.
3. Template manages the presentation layer. It is responsible for presenting the data and rendering the HTML.
4. Manage.py is used to interact with the project (starting server, ending server, databases, etc.)

Breaking down the GE:

1. For the GE02 django project, we did not set up any migration or ORM, there were no databases or queries. For future projects if there are going to be databases or ORM, it would communicate with views.py to represent the data correctly.
2. For views, we created paths with URL and views in order for the project to handle the html template
3. We have created an html template with embedded django template. The template.html and index.html file we created dictated how the data from views.py should be presented.
4. By typing in "python manage.py runserver" command, we were able to run the local server and see the changes and connections that were made in the project.



Source:

Lecture 4

https://canvas.uccs.edu/courses/152508/pages/l04-ch03-introsaaas?module_item_id=2429634

Https request and response

<https://docs.djangoproject.com/en/4.2/ref/request-response/>

Django MVT

<https://www.geeksforgeeks.org/django-project-mvt-structure/>

<https://www.javatpoint.com/django-mvt>