Repaso -Django

1.- Crear Proyecto en Pychar, cuidando que se cree con un entorno virtual venv

2.- Abrir la terminal pychar e instalar Django: ***pip install django***

3.- Crear un proyecto con Django: ***django-admin startproject mysite .*** (de esta manera se craer le proyecto dentro del proyecto creado en el punto 1)

Una vez creado el proyecto en Django analizaremos los archivos y para que se usan

settings.py: archivo de configuración central de todo el projecto Django

urls.py: archivo de rutas o urls principal del proyecto, acá van las rutas de tu landing y de cada app creada

4.- Una vez creado tu proyecto puedes agregar todas las vistas necesarias para hacer que tu landing page funcione adecuadamente, siguiendo los siguiente conceptos

A.- Templates, para crear tus templates debe ir a settings.py, modificar el arreglo TEMPLATES, para modificar la linea de DIRS y debe quedar de la siguiente manera

```
TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [BASE_DIR / 'templates'],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages'
            ],
        },
    },
]
```

B.- Crear en el mismo nivel de la app mysite una carpeta llamada templates

C.- Crear dentro de templates un archivo llamado base.html, acá vas a declarar todo el html principal que servirá de como template padre para aplicar la herencia, (acá colocar tus archivos de bootstrap y todos aquellos archivos generales a tu landing, recuerda que todo lo va acá se cargará en todos los templates que hereden de el.

No olvides declara todos los bloques necesarios para que puedas implementar la herencia.

```
{% extends "base.html" %}

{% block title %} Home {% endblock %}

{% block content %}
    <h1>About</h1>
{% endblock %}
```

D.- Crear todos los templates necesario para las rutas que vas a declarar: home, contact, about, projects etc etc

```
{% extends "base.html" %}

{% block title %} Home {% endblock %}

{% block content %}
    <h1>About</h1>
{% endblock %}
```

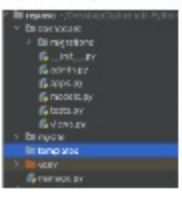
E.- Ir al archivo views.py dentro de tu app mysite, y crear todas las vitas necesarias para que puedan ser llamadas desde la urls.py

```
from django.shortcuts import render

A Richar
def home(request):
    return render(request, 'home.html')


A Richar
def contact(request):
    return render(request, 'contact.html',)


A Richar
def about(request):
    return render(request, 'about.html')
```

F.- Ir al archivo urls.py dentro de tu app mysite, y crear todas las rutas necesarias y que hagan coherencia con tus templates o vistas de tu landing

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

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.home, name='home'),
    path('contact/', views.contact, name='contact'),
    path('about/', views.about, name='about'),
]
```

5.- Para comenzar a agregar apps a tu sistema web, lo debes hacer con el siguiente comando: ***python manage.py startapp dashboard***, esto creará un app a nivel de mysite con la siguiente estructura



admin.py: permite declarar los modelos y sus configuraciones para que sean administrados desde el panel de administrador

models.py: permite declarar todas la configuraciones de tus modelos

views.py: permite declara todas funciones que reciben las peticiones desde el front y que llaman desde el urls.py

6.- Una vez creado tu app puedes agregar todas las vistas necesarias para hacer que tu app funcione adecuadamente, siguiendo los siguiente pasos

A.- Ir a mysite/settings.py, y agregar la app recién creada al arreglo INSTALLED_APPS

```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'dashboard',
    'smartphone',
]
```

B.- Crear dentro de tu app una carpeta llamada templates y dentro de esta, otra carpeta con el mismo nombre de tu app

C.- Crear dentro de la carpeta templates que esta a nivel de tu proyecto una carpeta llamada general y alli crear tu archivo base.html para que sirva de template padre para la herencia de tus apps

D.- Crear todos los templates necesario para las rutas que vas a declarar

E.- Ir al archivo views.py dentro de tu app, y crear todas las vitas necesarias para que puedan ser llamadas desde la urls.py

F.- Crear en tu app el archivo urls.py en donde deberás crear todas las rutas necesarias y que hagan coherencia con tus templates o vistas de tu app

G.- Ir al archivo mysite/urls.py, debes agregar las url declaradas en tu app dashboard

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

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.home, name='home'),
    path('contact/', views.contact, name='contact'),
    path('about/', views.about, name='about'),
    path('dashboard/', include('dashboard.urls')),
    path('smartphone/', include('smartphone.urls'))
]
```

7.- Para agregar postgres a tu proyecto debes ir al archivo mysite/settings.py y agregar el driver respectivo, mas la credenciales de tu base de datos

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql',
        'NAME': 'factory',
        'USER': 'postgres',
        'PASSWORD': '[illegible]',
        'HOST': 'localhost',
        'PORT': '5432',
    }
}
```

8.- Para realizar tu migraciones incluyendo las iniciales siempre debes ejecutar estos comandos:

python manage.py makemigration *(esto buscara todos lo modelos modificados o agregados y creará los archivos de migración)*
python manage.py migrate *(esto ejecutará las migraciones del comando anterior y llevara los cambios a la base de datos)*

9.- Para agregar autenticación a nuestro proyecto debemos ir al mysite/settings.py y agregar lo siguiente

```
LOGIN_REDIRECT_URL = 'authenticated'
LOGOUT_REDIRECT_URL = 'login'
LOGIN_URL = '/login/'
```

LOGIN_REDIRECT_URL, url a donde sera redireccionado la aplicación cuando alguien haga login efectivamente.
LOGOUT_REDIRECT_URL, url a donde será redireccionado la aplicación cuando se haga un logout efectivamente.
LOGIN_URL, url donde esta el inicio de sesión

10.- Ir a mysite/settings.py y agregar las rutas de autenticación login y logout

```
urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.home, name='home'),
    path('contact/', views.contact, name='contact'),
    path('about/', views.about, name='about'),
    path('login/', auth_views.LoginView.as_view(template_name='dashboard/login.html'), name='login'),
    path('dashboard/', include('dashboard.urls')),
    path('smartphone/', include('smartphone.urls'))
]
```

11.- para proteger rutas y que estas solo puedan ser accedidas cuando un persona esta logueada, se debe ocupar el decorador @login_required

```
A Richar
@login_required
def home(request):
    return render(request, 'dashboard/home.html')
```