

## EXERCISES QUE TRABAJAREMOS EN EL CUE

- EXERCISE 1: CREANDO PÁGINA DE CONTENIDO DINÁMICO.
- EXERCISE 2: INSTALANDO BOOTSTRAP EN DJANGO.
- EXERCISE 3: HERENCIA DE PLANTILLAS Y AGREGANDO UN NAVBAR DE BOOTSTRAP.

## EXERCISE 1: CREANDO PÁGINA DE CONTENIDO DINÁMICO.

El objetivo del presente ejercicio es plantear una guía paso a paso para desplegar las páginas con contenido dinámico, adecuar el proyecto con Bootstrap y la herencia de las plantillas.

Requisitos previos:

- Tener conocimiento de una terminal o línea de comando e instalación de paquetes de software en el sistema operativo, en este caso en Linux Ubuntu.
- Tener previamente instalado la versión de Python 3 y el virtualenv.
- Hacer uso de la herramienta Visual Studio Code.

Previamente creamos el entorno virtual `project_django`, y dentro el proyecto `site_web_django` y una aplicación boards. Para abrir el proyecto nos ubicamos en el directorio `site_web_django` y lo hacemos con VSC:

```
1 $ cd site_web_django
2
3 $ code .
```

Activamos el proyecto con workon:

```
1 $ workon projects_django
```

Ejecutamos la aplicación:

```
1 $ python manage.py runserver
```

Observamos en la terminal que se está ejecutando el servidor en `http://127.0.0.1:8000/` :

```
1 Watching for file changes with StatReloader
2 Performing system checks...
3
4 System check identified no issues (0 silenced).
5 July 10, 2022 - 20:35:21
6 Django version 4.0.5, using settings 'site_web_django.settings'
7 Starting development server at http://127.0.0.1:8000/
8 Quit the server with CONTROL-C.
```

Tenemos un contenido estático que muestra lo siguiente:



Procedemos a crear el primer ejemplo de contenido dinámico. En este caso, vamos a realizar una consulta al servidor con la finalidad de que nos devuelva la fecha y hora actual. Editamos archivo `boards/views.py`, y agregamos una nueva función `obtenerFecha`, la cual contiene el siguiente código:

#### BOARDS/VIEWS.PY

```
1 from django.views.generic import TemplateView
2 from django.shortcuts import render
3 import datetime
4 class IndexPageView(TemplateView):
5     template_name = "index.html"
6
7 def obtenerFecha(request):
8     fechaActual = datetime.datetime.now()
9     context = {'fecha' : fechaActual}
10    return render(request, 'fecha.html', context)
```

Creamos el archivo fecha.html en el directorio templates con el siguiente código:

## TEMPLATES/FECHA.HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>Document</title>
8 </head>
9 <body>
10     <h1>La Fecha actual es: {{ fecha }} </h1>
11 </body>
12 </html>
13
```

Procedemos a agregar la vista en la `urls.py` del boards:

## BOARDS/URLS.PY

```
1 from django.urls import path
2 from .views import IndexPageView, obtenerFecha
3 urlpatterns = [
4     path('', IndexPageView.as_view(), name='index'),
5     path('fecha/<name>', obtenerFecha, name='fecha'),
```

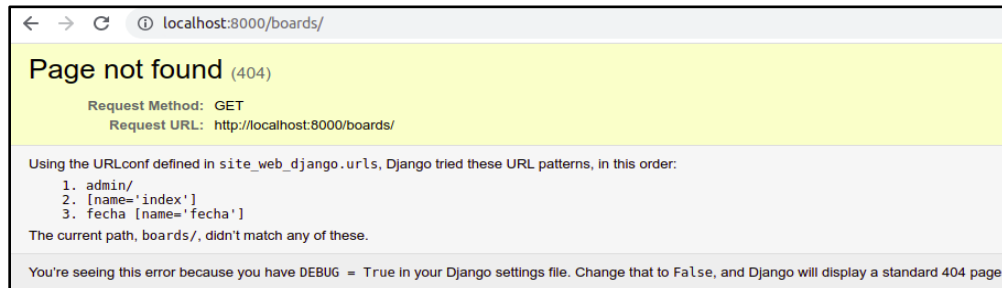
Se ejecuta el servidor de desarrollo de Django en la terminal:

```
1 $ python manage.py runserver
```

Accedemos en el navegador, y observamos la página principal con contenido estático  
<http://localhost:8000/>



Al colocar <http://localhost:8000/boards/>, observamos:



Los enlaces validos son:

<http://localhost:8000/>, <http://localhost:8000/admin/> y <http://localhost:8000/fecha>

En el enlace <http://localhost:8000/fecha> obtenemos datos que son dinámicos:



Luego, vamos a proceder a pasar datos por medio los parámetros de la URL. Esto significa que al solicitar la fecha, añadimos el parámetro nombre de tal manera nos muestre un mensaje según la variable nombre, y nos muestre un mensaje diciendo, por ejemplo: Jorge el día de hoy es July 12, 2022, 12:46 a.m. Procedemos a agregar el parámetro en el método obtenerFecha a la vista, en este caso name, y editamos el archivo con el siguiente código:

## BOARDS/VIEWS.PY

```
1 def obtenerFecha(request, name):  
2     fechaActual = datetime.datetime.now()  
3     context = {'fecha' : fechaActual, 'name' : name}  
4     return render(request, 'fecha.html', context)
```

En la URLs modificamos la ruta:

## BOARDS/URLS.PY

```
1 from django.urls import path
2
3 from .views import IndexPageView, obtenerFecha
4
5 urlpatterns = [
6     path('', IndexPageView.as_view(), name='index'),
7     path('fecha/<name>', obtenerFecha, name='fecha'),
8 ]
```

Modificamos el archivo fecha.html del template:

## TEMPLATE/FECHA.HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>Document</title>
8 </head>
9 <body>
10     <h1>{{ name }} hoy es: {{ fecha }} </h1>
11 </body>
12 </html>
13 ]
```

Procedemos a ejecutar el servidor de desarrollo de Django.

Terminal:

```
1 $ python manage.py runserver
```

Verificamos la dirección <http://localhost:8000/fecha>, y observamos:



La ruta URL correcta es:

<http://localhost:8000/fecha/<name>>, por ejemplo, <http://localhost:8000/fecha/Jorge>



## Jorge hoy es: July 12, 2022, 1 a.m.

### EXERCISE 2: INSTALANDO BOOTSTRAP EN DJANGO.

La forma preferida de instalar el marco Bootstrap para su aplicación Django es usar el paquete de **Python** **django-bootstrap-v5**, el cual se puede instalar usando PIP. Debido a que es diferente de la configuración anterior de la versión 4 de Bootstrap, hay un par de cambios que se deberán tener en cuenta:

#### LA NUEVA SINTAXIS

No necesita paquetes adicionales como **Django Crispy Forms** para diseñar sus formularios.

Procedemos a instalar Bootstrap con PIP en el proyecto que tenemos creado, site\_web\_django. abrimos la terminal y procedemos:

```
1 $ pip install django django-bootstrap-v5
```

Abra el archivo principal **settings.py** del proyecto, y agregue **django-bootstrap-v5** a sus aplicaciones instaladas.

## SETTINGS.PY

```
1 INSTALLED_APPS = [  
2     'django.contrib.admin',  
3     'django.contrib.auth',  
4     'django.contrib.contenttypes',  
5     'django.contrib.sessions',  
6     'django.contrib.messages',  
7     'django.contrib.staticfiles',  
8     'boards.apps.BoardsConfig',  
9     'bootstrap5',  
10 ]
```

Dentro del directorio de plantillas, cree un nuevo archivo y asígnele el nombre **base.html**. Puede usar la Terminal para crear la carpeta y el archivo, o usar el IDE. Abra el archivo **base.html** que creó anteriormente, y agregue las siguientes líneas en la parte superior del archivo, dentro de la etiqueta **<head>**.

```
1 {% load bootstrap5 %}  
2 {% bootstrap_css %}  
3 {% bootstrap_javascript %}  
4 {% bootstrap_messages %}
```

Cree un bloque de plantilla que usará para las plantillas de su proyecto: agregue las siguientes líneas a las etiquetas **<body>**

```
1 {% block content %}  
2 {% endblock content %}
```

Puede nombrar el bloque de plantilla con el nombre que desee.

Finalmente, el archivo **base.html** tiene el siguiente código:

## TEMPLATES/BASE.HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7
8     {% load bootstrap5 %}
9
10    {% bootstrap_css %}
11
12    {% bootstrap_javascript %}
13
14    {% bootstrap_messages %}
15
16    <title>Agregando Bootstrap</title>
17 </head>
18
19 <body>
20
21    {% block content %}
22
23    {% endblock content %}
24
25 </body>
26
27 </html>
28
```

El proyecto Django está listo para usar Bootstrap como marco de interfaz para diseñar sus páginas. Se puede comenzar a usar etiquetas Bootstrap para diseñar las páginas de su aplicación Django. Como referencia, en: <https://getbootstrap.com/docs/5.2/getting-started/introduction/>

Seguidamente como demostración, verificaremos la herencia de las plantillas al agregar un nuevo componente de Bootstrap.

### EXERCISE 3: HERENCIA DE PLANTILLAS Y AGREGANDO UN NAVBAR DE BOOTSTRAP.

Para realizar esta prueba, procedemos a modificar a crear un archivo llamado menu.html que hará uso de **Navbar** de Bootstrap. Para ello creamos el archivo, y agregamos el siguiente código tomado de la documentación oficial de Bootstrap: <https://getbootstrap.com/docs/5.2/components/navbar/>



## TEMPLATES/MENU.HTML

```
1 <nav class="navbar navbar-expand-lg bg-light">
2   <div class="container-fluid">
3     <a class="navbar-brand" href="#">Navbar</a>
4     <button class="navbar-toggler" type="button" data-bs-
5 toggle="collapse" data-bs-target="#navbarSupportedContent" aria-
6 controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
7 navigation">
8     <span class="navbar-toggler-icon"></span>
9   </button>
10  <div class="collapse navbar-collapse" id="navbarSupportedContent">
11    <ul class="navbar-nav me-auto mb-2 mb-lg-0">
12      <li class="nav-item">
13        <a class="nav-link active" aria-current="page"
14 href="#">Home</a>
15      </li>
16      <li class="nav-item">
17        <a class="nav-link" href="#">Link</a>
18      </li>
19      <li class="nav-item dropdown">
20        <a class="nav-link dropdown-toggle" href="#"
21 id="navbarDropdown" role="button" data-bs-toggle="dropdown" aria-
22 expanded="false">
23          Dropdown
24        </a>
25        <ul class="dropdown-menu" aria-labelledby="navbarDropdown">
26          <li><a class="dropdown-item" href="#">Action</a></li>
27          <li><a class="dropdown-item" href="#">Another
28 action</a></li>
29          <li><hr class="dropdown-divider"></li>
30          <li><a class="dropdown-item" href="#">Something else
31 here</a></li>
32        </ul>
33      </li>
34      <li class="nav-item">
35        <a class="nav-link disabled">Disabled</a>
36      </li>
37    </ul>
38    <form class="d-flex" role="search">
39      <input class="form-control me-2" type="search"
40 placeholder="Search" aria-label="Search">
41      <button class="btn btn-outline-success"
42 type="submit">Search</button>
43    </form>
44  </div>
45 </div>
46 </nav>
```

Seguidamente, vamos a crear un método `menuView` en la vista del boards.

## BOARDS/VIEWS.PY

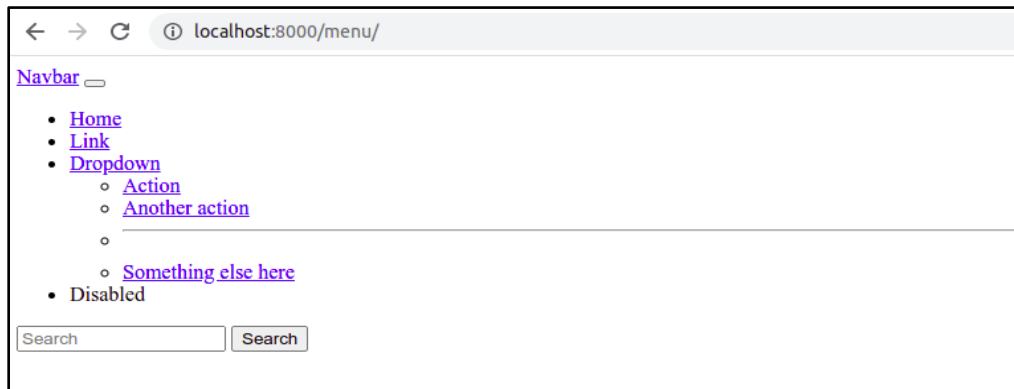
```
1 .....  
2 def menuView(request):  
3     template_name = 'menu.html'  
4     return render(request, template_name)
```

Agregamos en la URL el menú:

## BOARDS/URLS.PY

```
1 from django.urls import path  
2  
3 from .views import IndexPageView, obtenerFecha, menuView  
4  
5 urlpatterns = [  
6     path('', IndexPageView.as_view(), name='index'),  
7     path('fecha/<name>', obtenerFecha, name='fecha'),  
8     path('menu/', menuView, name='menu'),  
9 ]
```

Y verificamos en el navegador:



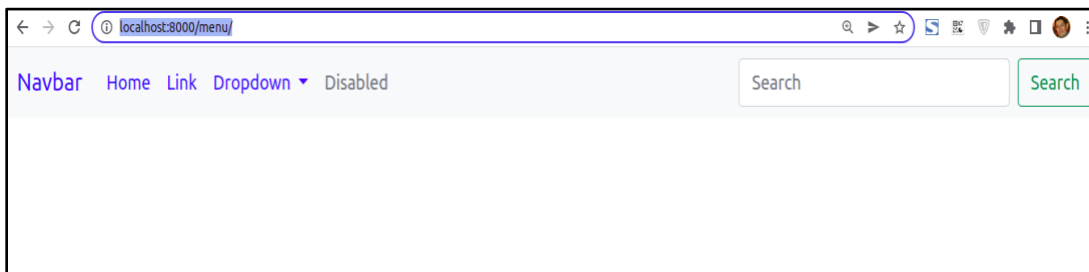
Luego, procedemos a agregarle estilo. Para ello se aplicará la forma de herencia en plantillas por medio del método `extends`, en el archivo `menu.html` agregamos la lectura de bootstrap 5 y la herencia, el código final queda de la siguiente manera:

## TEMPLATES/MENU.HTML

```
1 {% extends 'base.html' %}
2
3 {% load bootstrap5 %}
4
5 {% block content %}
6
7 <nav class="navbar navbar-expand-lg bg-light">
8     <div class="container-fluid">
9         <a class="navbar-brand" href="#">Navbar</a>
10        <button class="navbar-toggler" type="button" data-bs-
11toggle="collapse" data-bs-target="#navbarSupportedContent" aria-
12controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
13navigation">
14        <span class="navbar-toggler-icon"></span>
15        </button>
16        <div class="collapse navbar-collapse" id="navbarSupportedContent">
17            <ul class="navbar-nav me-auto mb-2 mb-lg-0">
18                <li class="nav-item">
19                    <a class="nav-link active" aria-current="page"
20href="#">Home</a>
21                </li>
22                <li class="nav-item">
23                    <a class="nav-link" href="#">Link</a>
24                </li>
25                <li class="nav-item dropdown">
26                    <a class="nav-link dropdown-toggle" href="#"
27id="navbarDropdown" role="button" data-bs-toggle="dropdown" aria-
28expanded="false">
29                        Dropdown
30                    </a>
31                    <ul class="dropdown-menu" aria-labelledby="navbarDropdown">
32                        <li><a class="dropdown-item" href="#">Action</a></li>
33                        <li><a class="dropdown-item" href="#">Another
34action</a></li>
35                        <li><hr class="dropdown-divider"></li>
36                        <li><a class="dropdown-item" href="#">Something else
37here</a></li>
38                    </ul>
39                </li>
40                <li class="nav-item">
41                    <a class="nav-link disabled">Disabled</a>
42                </li>
43            </ul>
44            <form class="d-flex" role="search">
45                <input class="form-control me-2" type="search"
46placeholder="Search" aria-label="Search">
47                <button class="btn btn-outline-success"
48type="submit">Search</button>
49            </form>
```

```
50     </div>
51   </div>
52 </nav>
53
54 {% endblock %}
```

Al acceder al navegador, observamos: <http://localhost:8000/menu/>



Y es responsivo:

