

Informe de Laboratorio 06

Tema: Django - Usando una plantilla para ver Destinos Turísticos

Nota

Estudiante	Escuela	Asignatura
Fernando Miguel Garambel Marín fgarambel@unsa.edu.pe	Escuela Profesional de Ingeniería de Sistemas	Laboratorio de Programación Web 2 Semestre: III Código: 1701212

Laboratorio	Tema	Duración
06	Django - Usando una plantilla para ver Destinos Turísticos	04 horas

Semestre académico	Fecha de inicio	Fecha de entrega
2024 - A	Del 24 de mayo 2024	Al 5 de junio 2024

1. Objetivos

- Implementar una aplicación en Django utilizando una plantilla profesional.
- Utilizar una tabla de Destinos turísticos para leer y completar la página web
- Utilizar los tags “if” y “for” en los archivos html para leer todos los registros de una tabla desde una base de datos.

2. Actividades

- Crear un proyecto en Django
- Siga los pasos del video para poder implementar la aplicación de Destinos turísticos
- Use git y haga los commits necesarios para manejar correctamente la aplicación.

3. Ejercicio Propuestos

- Deberán replicar la actividad del video que se encuentra en el AV de Teoría (Django Tutorial for Beginners - Telusko(<https://youtu.be/OTmQQjs10eg>) donde se obtiene una plantilla de una aplicación de Destinos turísticos y adecuarla a un proyecto en blanco Django.

- Luego trabajar con un modelo de tabla DestinosTuristicos donde se guarden nombreCiudad, descripcionCiudad, imagenCiudad, precioTour, ofertaTour (booleano). Estos destinos turísticos deberán ser agregados en una vista dinámica utilizando tags for e if.
- Para ello crear una carpeta dentro del proyecto github colaborativo con el docente, e informar el link donde se encuentra.
- Crear formularios de Añadir Destinos Turísticos, Modificar, Listar y Eliminar Destinos.

4. Equipos, materiales y temas utilizados

- Sistema operativo de 64 bits, procesador basado en x64.
- Latex.
- git version 2.41.0.windows.1
- Cuenta en GitHub con el correo institucional.

5. URL Github, Video

- URL del Repositorio GitHub para clonar o recuperar.
- https://github.com/FernandoGarambelM/Destinos_turisticos.git
- URL para el video flipgrid
- Video

6. Replicando la actividad del video

- Primero ingresamos al entorno virtual

Listing 1: Activar el ambiente virtual

```
env\Scripts\activate
```

Listing 2: Clonar el repositorio

```
git clone https://github.com/navinreddy20/django-telusko-codes
```

- Luego instalamos postgresql y pgadmin
- Luego instalamos al entorno virtual psycopg2-binary y Pillow

Listing 3: Instalando psycopg2-binary y Pillow

```
pip install psycopg2-binary  
pip install Pillow
```

- Después de instalar, creamos un superusuario con el siguiente script

Listing 4: Script para crear un superusuario

```
#!/bin/sh  
  
# Variables de entorno para el superusuario  
DJANGO_SUPERUSER_USERNAME=admin  
DJANGO_SUPERUSER_EMAIL=admin@example.com  
DJANGO_SUPERUSER_PASSWORD=123456  
  
# Ejecutar el comando createsuperuser sin interaccion  
python manage.py shell -c "from django.contrib.auth import get_user_model; User =  
get_user_model(); User.objects.create_superuser('$DJANGO_SUPERUSER_USERNAME',  
'$DJANGO_SUPERUSER_EMAIL', '$DJANGO_SUPERUSER_PASSWORD') if not  
User.objects.filter(username='$DJANGO_SUPERUSER_USERNAME').exists() else  
print('Superusuario ya existe.')"
```

- Realizamos migraciones para enlazar la base de datos

Listing 5: Código para realizar migraciones

```
python manage.py makemigrations  
python manage.py migrate
```

- Después de seguir estos pasos la plantilla ya funciona por completo

7. Adecuar la plantilla a un proyecto en blanco de Django

- Creamos el modelo destination con los datos requeridos

Listing 6: Código de models.py

```
1 from django.db import models
2
3 # Create your models here.
4
5 class Destination(models.Model):
6
7     name = models.CharField(max_length=100)
8     img = models.ImageField(upload_to='pics')
9     desc = models.TextField()
10    price = models.IntegerField()
11    offer = models.BooleanField(default=False)
```

- Luego creamos los forms

Listing 7: Código de forms.py

```
1 from django import forms
2 from .models import Destination
3
4 class DestinosTuristicosForm(forms.ModelForm):
5     class Meta:
6         model = Destination
7         fields = ['name', 'img', 'desc', 'price', 'offer']
```

- Luego creamos views.py

Listing 8: Código de views.py

```
1 from django.shortcuts import render, redirect, get_object_or_404
2 from .models import Destination
3 from .forms import DestinosTuristicosForm
4 # Create your views here.
5
6 def index(request):
7
8     dests = Destination.objects.all()
9
10    return render(request, "index.html", {'dests': dests})
11
12 def gestionar_destinos(request, id=None):
13     if request.method == 'POST':
14         if id:
15             destino = get_object_or_404(Destination, id=id)
16             form = DestinosTuristicosForm(request.POST, request.FILES, instance=destino)
17             if 'modificar' in request.POST:
18                 if form.is_valid():
19                     form.save()
20             elif 'eliminar' in request.POST:
```

```

21         destino.delete()
22     else:
23         form = DestinosTuristicosForm(request.POST, request.FILES)
24         if form.is_valid():
25             form.save()
26         return redirect('gestionar_destinos')
27
28     else:
29         if id:
30             destino = get_object_or_404(Destination, id=id)
31             form = DestinosTuristicosForm(instance=destino)
32         else:
33             form = DestinosTuristicosForm()
34
35     destinos = Destination.objects.all()
36     return render(request, 'gestionar_destinos.html', {
37         'form': form,
38         'destinos': destinos,
39         'id': id,
40     })

```

- Luego agregamos urls.py para que Django pueda leerlos

Listing 9: Código de urls.py

```

1 from django.urls import path
2
3 from . import views
4
5 urlpatterns = [
6     path("", views.index, name="index"),
7     path('gestionar/', views.gestionar_destinos, name="gestionar_destinos"),
8     path('gestionar/<int:id>', views.gestionar_destinos, name="modificar_destino"),
9 ]

```

- Usamos la plantilla de travello, específicamente la siguiente parte para listar los destinos turísticos con if y for

Listing 10: Pedazo de html donde se usa for e if

```

{% for dest in dests %}
    <!-- Destination -->
    <div class="destination item">
        <div class="destination_image">
            

            {% if dest.offer %}
            <div class="spec_offer text-center"><a href="#">Special
                Offer</a></div>
            {% endif %}
        </div>
        <div class="destination_content">
            <div class="destination_title"><a
                href="destinations.html">{{dest.name}}</a>

```

```
</div>
<div class="destination_subtitle">
  <p>{{dest.desc}}</p>
</div>
<div class="destination_price">From ${{dest.price}}</div>
</div>
</div>

{% endfor %}
```

Cursos

ID	Nombre	Descripción
1	Programacion Web 2	Curso de segundo año,primer semestre

Notas

ID	Alumno	Curso	Nota
1	Luis Guillermo Luque	Programacion Web 2	15

8. Referencias

- <https://docs.djangoproject.com/es/3.2/>
- <https://docs.djangoproject.com/es/3.2/ref/models/fields/#field-types>