

## Informe de Laboratorio 06

### Tema: Django

**Nota**

Estudiante	Escuela	Asignatura
Ajra Huacso Jeans Anthony jajra@unsa.edu.pe	Escuela Profesional de Ingeniería de Sistemas	Laboratorio de Programación WEB 2 Semestre: I Código: 20232208

Laboratorio	Tema	Duración
06	Django	04 horas

Semestre académico	Fecha de inicio	Fecha de entrega
2024 - A	Del 21 de mayo del 2024	Al 26 de Mayo del 2024

### 1. EJERCICIOS PROPUESTOS:

Implementa un Sistema en Django que maneje una tabla de Alumnos, una de Cursos y una de NotasAlumnosPorCurso y que permita ingresar a nuevos alumnos, nuevos cursos y finalmente permita ingresar las notas por curso.

#### Observaciones:

Haga un commit para cada paso de esta aplicación propuesta de Notas por Curso por Alumno..

1. Cada commit debe ser realizado con un mensaje descriptivo que estuvo siguiendo. Si los mensajes no son claros respecto al código "comiteado" tendrá menos calificación.
2. Use branches para hacer sus pruebas, como el caso de crear aplicaciones adicionales a la aplicación inicial, creada en clase.
3. Revise la documentación de Django e incluya nuevos tipos de datos para sus nuevas aplicaciones, en sus propias ramas o subramas

## Configuración de models.py

```
1 from django.db import models
2
3 class Alumno(models.Model):
4     nombre = models.CharField(max_length=200)
5
6     def __str__(self):
7         return self.nombre
8
9 class Curso(models.Model):
10     nombre = models.CharField(max_length=200)
11
12     def __str__(self):
13         return self.nombre
14
15 class Nota(models.Model):
16     alumno = models.ForeignKey(Alumno, on_delete=models.CASCADE)
17     curso = models.ForeignKey(Curso, on_delete=models.CASCADE)
18     nota = models.DecimalField(max_digits=5, decimal_places=2)
19
20     def __str__(self):
21         return f"Nota de {self.alumno.nombre} en {self.curso.nombre}: {self.nota}"
```

## Configuración de forms.py

```
1 from django import forms
2 from .models import *
3
4 class AlumnoForm(forms.ModelForm):
5     nombre = forms.CharField(max_length=200, label='Nombre del Alumno')
6
7     class Meta:
8         model = Alumno
9         fields = ['nombre']
10
11     def save(self, commit=True):
12         alumno = super().save(commit=False)
13         if commit:
14             alumno.save()
15         return alumno
16
17 class CursoForm(forms.ModelForm):
18     nombre = forms.CharField(max_length=200, label='Nombre del Curso')
19
20     class Meta:
21         model = Curso
22         fields = ['nombre']
23
24     def save(self, commit=True):
25         curso = super().save(commit=False)
26         if commit:
27             curso.save()
28         return curso
29
```

```
30 class NotaForm(forms.ModelForm):
31     alumno = forms.ModelChoiceField(queryset=Alumno.objects.all(), label='Alumno')
32     curso = forms.ModelChoiceField(queryset=Curso.objects.all(), label='Curso')
33     nota = forms.DecimalField(max_digits=5, decimal_places=2, label='Nota')
34
35     class Meta:
36         model = Nota
37         fields = ['alumno', 'curso', 'nota']
38
39     def save(self, commit=True):
40         nota = super().save(commit=False)
41         if commit:
42             nota.save()
43         return nota
```

### Configuración de views.py

```
1 from django.shortcuts import render, redirect
2 from .forms import *
3 from .models import *
4
5 #CREAR ALUMNO
6 def crear_alumno(request):
7     if request.method == 'POST':
8         form = AlumnoForm(request.POST)
9         if form.is_valid():
10             form.save()
11             return redirect('lista_alumnos')
12     else:
13         form = AlumnoForm()
14         return render(request, 'registroCurso/crear_alumno.html', {'form': form})
15
16 def lista_alumnos(request):
17     alumnos = Alumno.objects.all()
18     return render(request, 'registroCurso/lista_alumnos.html', {'alumnos': alumnos})
19
20 #CREAR CURSO
21 def crear_curso(request):
22     if request.method == 'POST':
23         form = CursoForm(request.POST)
24         if form.is_valid():
25             form.save()
26             return redirect('lista_cursos')
27     else:
28         form = CursoForm()
29         return render(request, 'registroCurso/crear_curso.html', {'form': form})
30
31 def lista_cursos(request):
32     cursos = Curso.objects.all()
33     return render(request, 'registroCurso/lista_cursos.html', {'cursos': cursos})
34
```

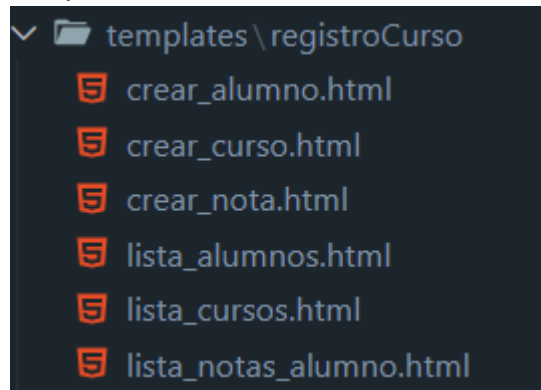
```
35 #CREAR NOTA
36 def crear_nota(request):
37     if request.method == 'POST':
38         form = NotaForm(request.POST)
39         if form.is_valid():
40             form.save()
41             return redirect('lista_notas_alumno')
42     else:
43         form = NotaForm()
44     return render(request, 'registroCurso/crear_nota.html', {'form': form})
45
46 def lista_notas_alumno(request):
47     notas = Nota.objects.select_related('alumno', 'curso').all()
48     return render(request, 'registroCurso/lista_notas_alumno.html', {'notas': notas})
```

Configuración de urls.py tanto dentro de la aplicación como del proyecto

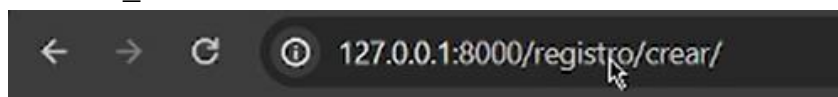
```
1 from django.urls import path
2 from . import views
3
4 urlpatterns = [
5     path('crear/', views.crear_alumno, name='crear_alumno'),
6     path('lista/', views.lista_alumnos, name='lista_alumnos'),
7     path('crearCur/', views.crear_curso, name='crear_curso'),
8     path('listaCur/', views.lista_cursos, name='lista_cursos'),
9     path('crearNota/', views.crear_nota, name='crear_nota'),
10    path('listaNota/', views.lista_notas_alumno, name='lista_notas_alumno'),
11 ]
```

```
1 from django.contrib import admin
2 from django.urls import path, include
3
4 urlpatterns = [
5     path('admin/', admin.site.urls),
6     path('registro/', include('registroCurso.urls')),
7 ]
8
```

Creación de los templates:



Plantilla crear\_alumno.html:



## Crear Alumno

Nombre del Alumno:

[Ver todas los alumnos](#)

Plantilla crear\_curso.html:



## Crear Curso

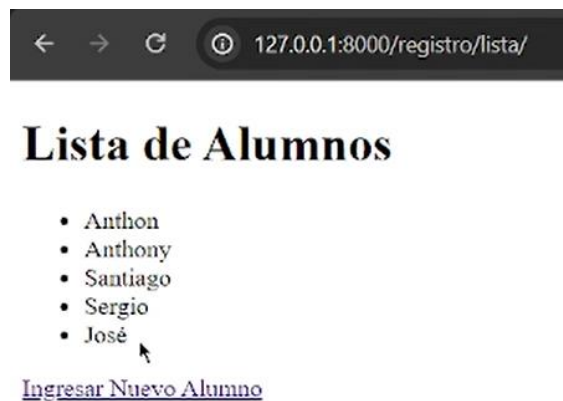
Nombre del Curso:

[Ver todas los cursos](#)

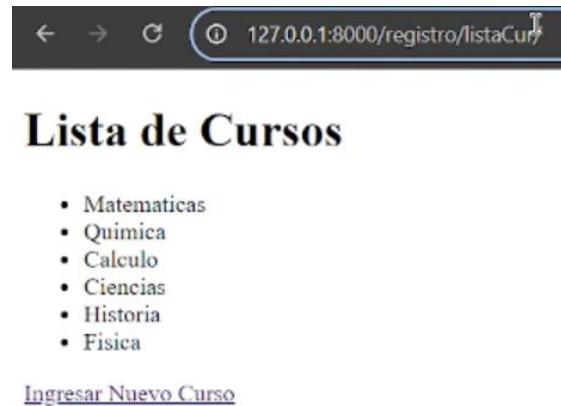
Plantilla crear\_notas.html:



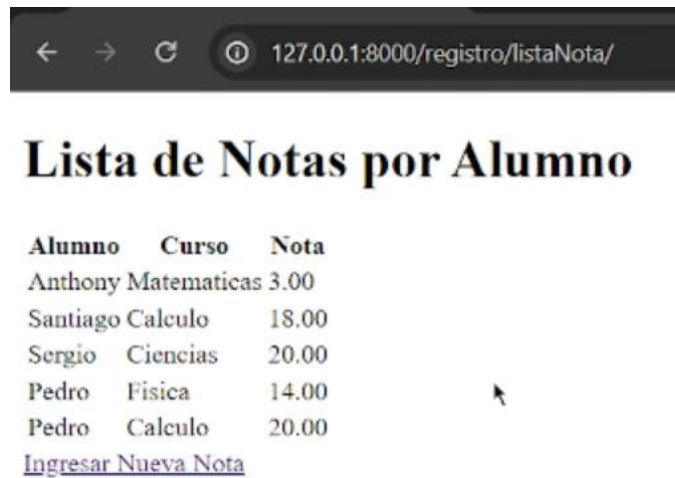
Plantilla lista\_alumnos.html:



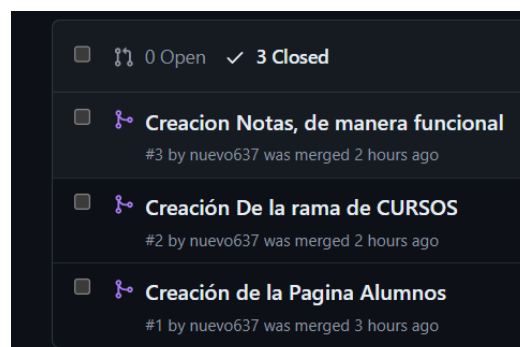
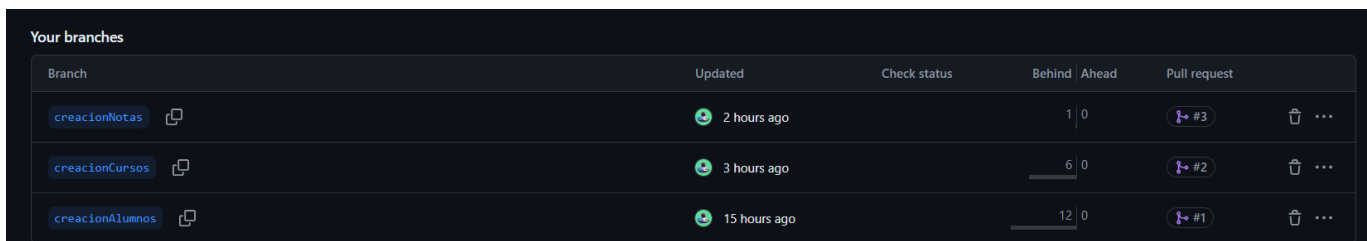
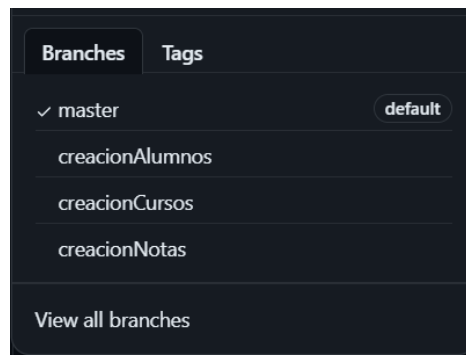
Plantilla lista\_cursos.html:



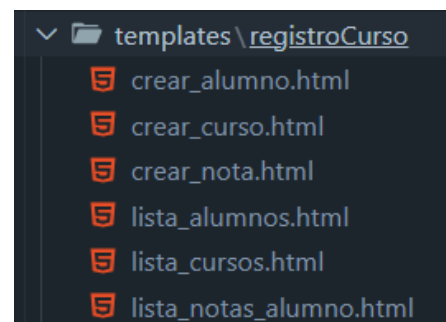
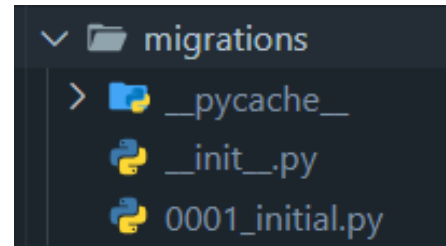
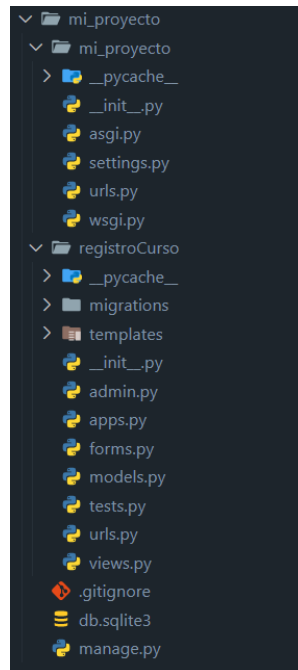
Plantilla lista\_notas\_alumno.html:



Se usaron ramas dentro de git, para implementar el proyecto, como se solicito dentro de la práctica del laboratorio.



## Estructura del Proyecto Realizado:



URL DE VIDEO DE EXPLICACIÓN: <https://flip.com/s/eiZ8NrLxvqgZ>

URL DE REPOSITORIO DE GITHUB: <https://github.com/nuevo637/Lab06-Django>

## 2. Referencias

- <https://www.w3schools.com/django/default.asp>