



**UTM**

M.G.T.I Maria Elena Benitez Ramirez



# **PROYECTO 2**

## **BIBLIOTECA VIRTUAL**

### **EQUIPO BIEN**

~Axl Baruck Galván López  
~Jazzani Ivett García Juárez  
~Nicolás Lemus Cisneros  
~Erick Francisco Cruz Robles



**9ºA**

## **Etapas 1:** Investigar y definir.

### **1.1 Planteamiento completo del problema.**

#### **Biblioteca Digital**

En la época digital, el acceso a la información y a los recursos educativos ha experimentado un avance significativo. Las bibliotecas digitales han sido creadas como una opción moderna, brindando acceso inmediato a una amplia gama de libros y recursos en línea. El propósito de este proyecto de biblioteca digital es brindar una plataforma en la que los usuarios puedan acceder, descargar y reseñar libros en formato PDF.

Los objetivos específicos de este proyecto son:

- **Proveer acceso a una colección de libros digitales:** Los usuarios deben poder buscar y acceder a libros en formato PDF.
- **Diferentes niveles de acceso y permisos:** Implementar roles de usuario con diferentes permisos y privilegios.
- **Funcionalidad de descarga de PDFs:** Permitir a los usuarios descargar los libros en formato PDF.
- **Sistema de reseñas:** Permitir a los usuarios escribir y leer reseñas de los libros.
- **Ranking de libros:** Mostrar un ranking de los libros más populares o mejor valorados.

### **1.2 Requisitos Funcionales**

- **Gestión de Información Bibliográfica:** La biblioteca digital debe permitir el almacenamiento de información bibliográfica de títulos, incluyendo:
  - Título
  - Autor
  - Edición
  - ISBN
  - Enlaces a las URLs alternativas para el acceso al PDF del título
- **Categorías de Títulos:**
  - Cada título debe ser catalogado en una o más categorías.
  - Las categorías deben ser gestionables por los administradores.
- **Gestión de Usuarios:**
  - Registro e Identificación:
  - Los usuarios deben registrarse y autenticarse en el sistema para acceder a la biblioteca.

- **Roles y Permisos:** Dos roles principales:
  - Administrador y Usuario.
  - Administrador: Puede gestionar títulos, usuarios, categorías, y realizar cualquier operación en el sistema.
  - Usuario: Puede buscar, consultar, evaluar y comentar sobre los títulos, y debe tener acceso basado en su categoría
- **Búsqueda y Recuperación de Información:** Los usuarios deben poder buscar títulos por:
  - Autor
  - Nombre del título
  - Categoría
  - La búsqueda debe retornar una lista de títulos coincidentes.
- **Evaluación y Comentarios:** Los usuarios pueden evaluar los títulos con una clasificación de 1 a 5 estrellas.
  - Los usuarios pueden escribir comentarios sobre los títulos.
  - La biblioteca debe calcular un promedio de calificaciones y utilizarlo para recomendar títulos basados en la popularidad y preferencia.

## Requisitos No Funcionales

**1. Interfaz de Usuario:** La interfaz debe ser intuitiva y fácil de usar tanto para administradores como para usuarios finales.

-Debe ser accesible desde diferentes dispositivos (ordenadores, tabletas, móviles).

**2. Desempeño:** La búsqueda y recuperación de información deben ser rápidas y eficientes, incluso con un gran volumen de datos.

**3. Escalabilidad:** El sistema debe ser capaz de manejar un creciente número de títulos y usuarios sin degradar el rendimiento.

**4. Disponibilidad y Fiabilidad:** El sistema debe estar disponible 24/7 con una alta fiabilidad y bajo riesgo de caída.

**5. Mantenibilidad:** El sistema debe ser fácil de mantener y actualizar, permitiendo la incorporación de nuevas funcionalidades sin afectar la operación actual.

### 3. Requisitos de Seguridad

**1. Autenticación y Autorización:** Implementar autenticación segura (por ejemplo, usando contraseñas cifradas).

- Los permisos de acceso deben ser adecuados a los roles y categorías de usuario.

**2. Protección de Datos:** Los datos personales y bibliográficos deben ser protegidos de acuerdo con las regulaciones de protección de datos (por ejemplo, GDPR).

- Implementar medidas de seguridad para proteger la integridad y confidencialidad de los documentos PDF y los comentarios de los usuarios.

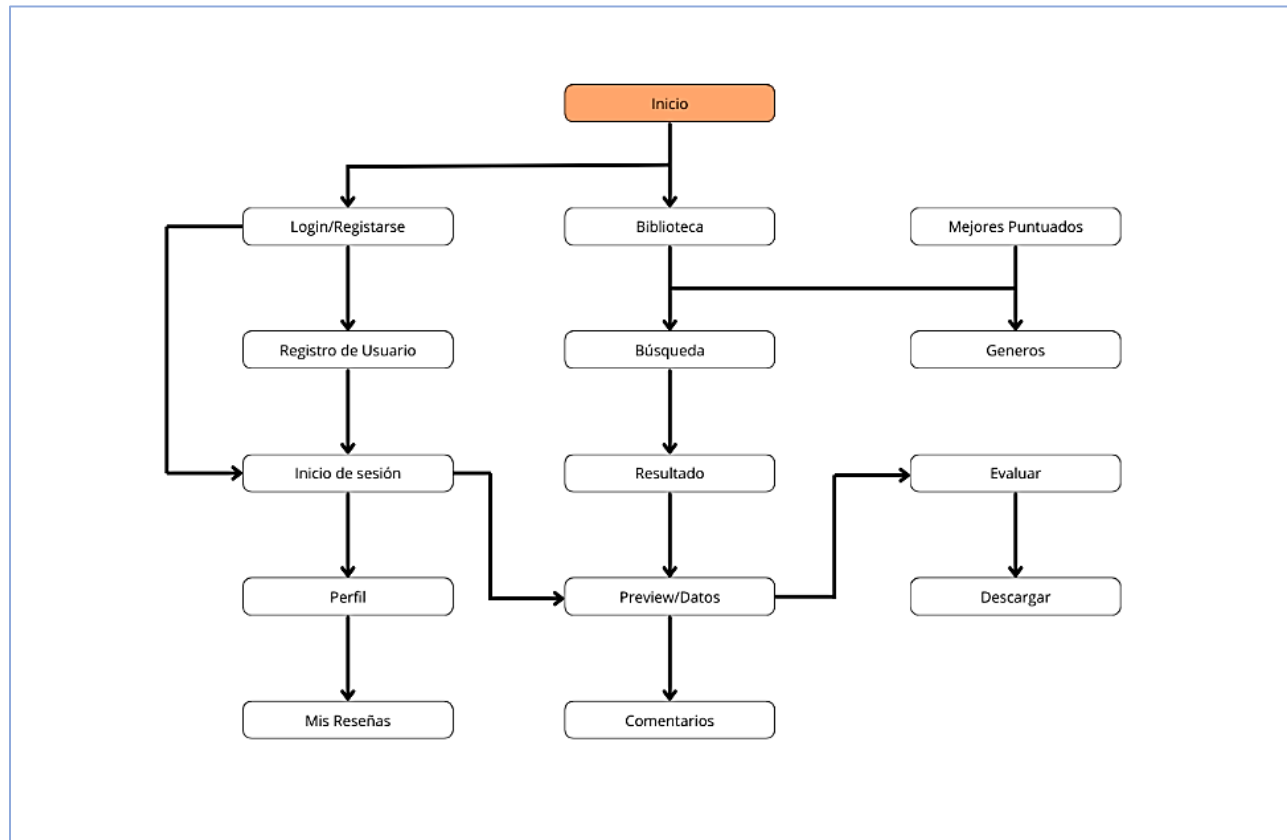
- Implementar cifrado de extremo a extremo para la transmisión de datos sensibles entre los clientes y el servidor. Con la herramienta del framework Django; Cookies seguras y HttpOnly.

**3. Auditoría y Registro:** Mantener registros de las actividades de los usuarios y administradores para auditoría y solución de problemas.

**4. Validación de entradas:** - Validación de entradas: Validar todas las entradas de usuario para prevenir inyecciones de código (inyección de SQL, XSS).

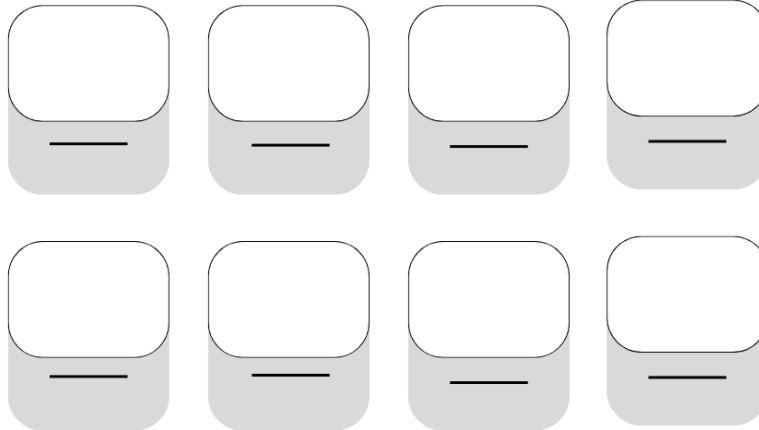
**Etapas 2:** Boceto - Arquitectura de la información. - FINAL

## 2.1 Mapa de navegación.



## 2.2 Wireframes.



[a](#)

## ¡Bienvenidos!

But I must explain to you how all this mistaken idea of denouncing pleasure and praising pain was born and I will give you a complete account of the system, and expound the actual teachings of the great explorer of the truth, the master-builder of human happiness. No one rejects, dislikes, or avoids pleasure itself, because it is pleasure, but because those who do not know how to pursue pleasure rationally encounter consequences that are extremely painful.

### TOP 3 MEJORES LIBROS DEL MOMENTO!




Libro 1





Libro 2




Libro 3




Search




## Biblioteca Digital




Libro x




Libro x




Libro x




Libro x




Libro x




Libro x





Libro x



Libro x




Search



## Editar perfil


NOMBRE DE USUARIO



CORREO



FOTO DE PERFIL



Guardar datos




Search

# Perfil

NOMBRE DE USUARIO

CORREO

FOTO DE PERFIL



Editar perfil

Search

# Crea una cuenta

Ya estas registrado? Inicio sesión

NOMBRE DE USUARIO

CORREO

CONTRASEÑA

VERIFICAR CONTRASEÑA

Registrarte



Search



# Inicio de sesión

No tienes cuenta? [Regístrate](#)

CORREO

CONTRASEÑA



Captcha

**Iniciar**

## 2.3 Mockups.





Search

### El libro de Bill



Autor: Alex Hirsch

El demonio que aterrizó Gravity Falls está de vuelta desde el más allá para contar por fin su versión de la historia en El libro de Bill, escrito nada menos que por el mismísimo Bill Clave. En estas páginas, Bill comparte sus extraños orígenes, su siniestro efecto en la historia de la humanidad, los secretos más vergonzosos de la familia Pines y la receta para conquistar el mundo (con una práctica guía paso a paso). Este caótico y bello tomo ilustrado contiene confusos acertijos, claves indecifrables, páginas perdidas del Diario 3, maneras de engañar la muerte, el significado de la vida y un capítulo completo sobre popotes bobos con formas divertidas. Pero lo más importante es que El libro de Bill está muy, muy maldito. Alex Hirsch, el autor bestseller, resucita a este infame villano e invita a los fans a ver el universo Gravity Falls desde la perspectiva de Bill. Muchos creen que este libro es demasiado peligroso para las manos humanas. Pero si no logras resistirte, debes saber esto: una vez que haces un trato con Bill, no es fácil retractarte...





Search

### El libro de Bill

Categorías: A, B, C.

Edición: 2024

Autor: Alex Hirsch

ISBN: 9788418271755

**CALIFICACIÓN: 4.7 estrellas**



Vista Previa  
Exclusiva para Usuarios

Iniciar Sesión



Descargar Solo Usuarios


### Reseñas




**Zellensxn\_2215** 5 estrellas  
Traducción espectacular, giros de trama y protagonistas de los godsss 10/10 papu.

Search


## GÉNEROS




Documental y Realidad




Ciencia Ficción y Fantasía




Deportivos




Romance




Misterio y Suspense



Manga



Infantiles y Juveniles




Desarrollo Personal

Search


## ¡Bienvenidos!

But I must explain to you how all this mistaken idea of denouncing pleasure and praising pain was born and I will give you a complete account of the system, and expound the actual teachings of the great explorer of the truth, the master-builder of human happiness. No one rejects, dislikes, or avoids pleasure itself, because it is pleasure, but because those who do not know how to pursue pleasure rationally encounter consequences that are extremely painful.

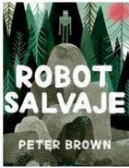
### ¡TOP 3 MEJORES LIBROS DEL MOMENTO!




El libro de Bill





Lugares Asombrosos 2




Robot Salvaje




Search



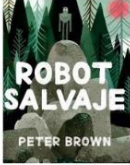
## Biblioteca Digital



El libro de Bill




Lugares  
Asombrosos 2




ROBOT  
SALVAJE  
PETER BROWN

Robot Salvaje



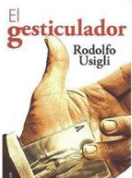
NIMONA

Robot Salvaje



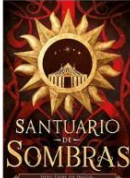
Wassily Kandinsky  
PUNTO Y LÍNEA  
SOBRE EL PLANO

Punto y Línea  
Sobre el Plano




El  
gesticulador  
Rodolfo Usigli

El gesticulador




SANTUARIO  
de  
SOMBRA


Santuario de  
Sombras




JOANA MARCÚS  
ETÉREO  
Santuario de la luz y la sombra

Etéreo




Search



## Editar Perfil

NOMBRE DE USUARIO



CORREO








FOTO DE PERFIL



Editar perfil



Search




# Perfil

NOMBRE DE USUARIO

CORREO

FOTO DE PERFIL



Editar perfil



Search



# Crea una cuenta

Ya estas registrado? [Inicio sesión](#)

NOMBRE DE USUARIO

CORREO

CONTRASEÑA



VERIFICAR CONTRASEÑA



Registrarte



Search



# Inicio de sesión

No tienes cuenta? [Regístrate](#)

CORREO

CONTRASEÑA

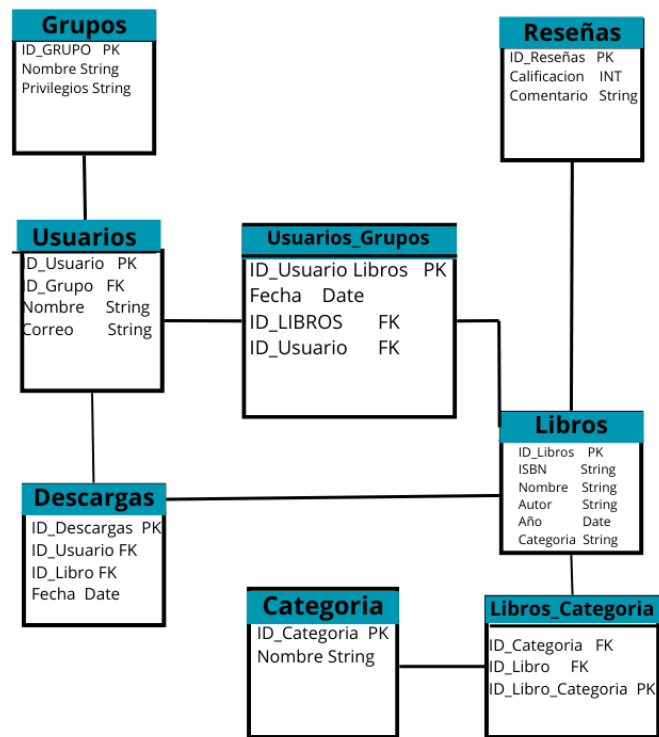


Captcha

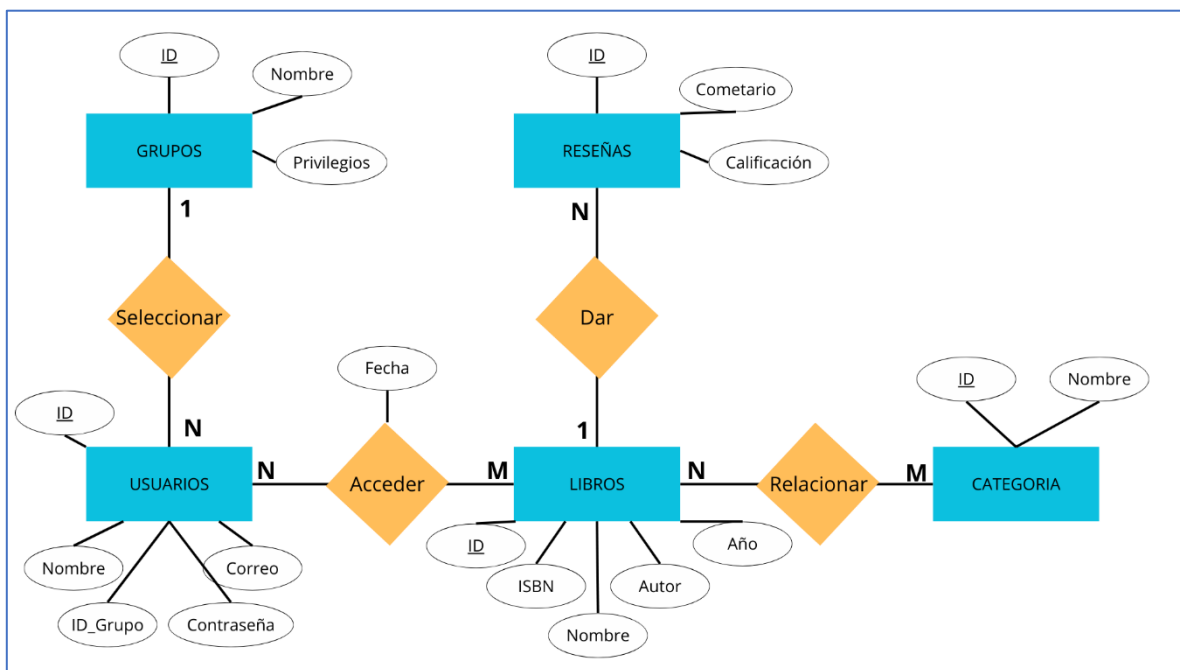
Iniciar

### Etapa 3. Diseño y Desarrollo

#### 3.1 Modelo E-R.

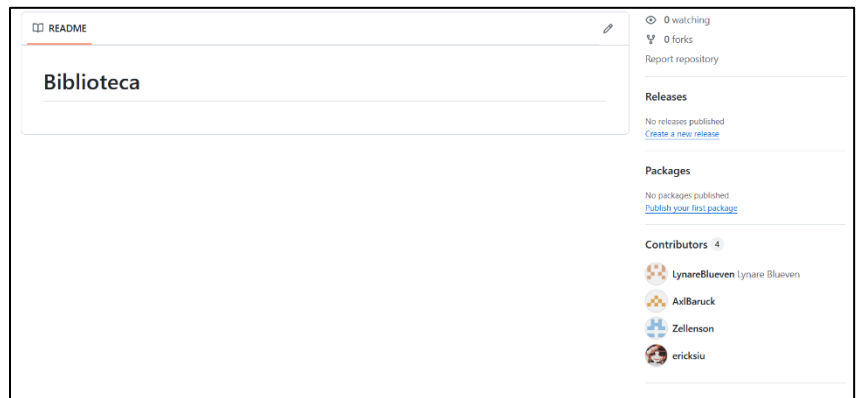
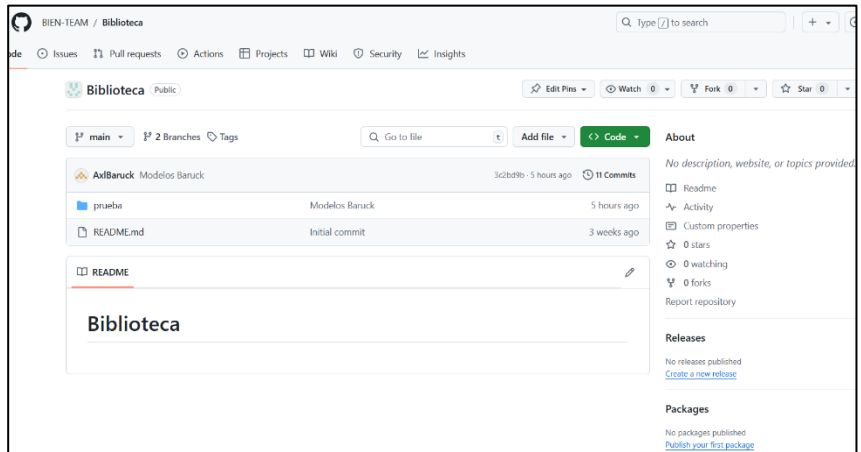
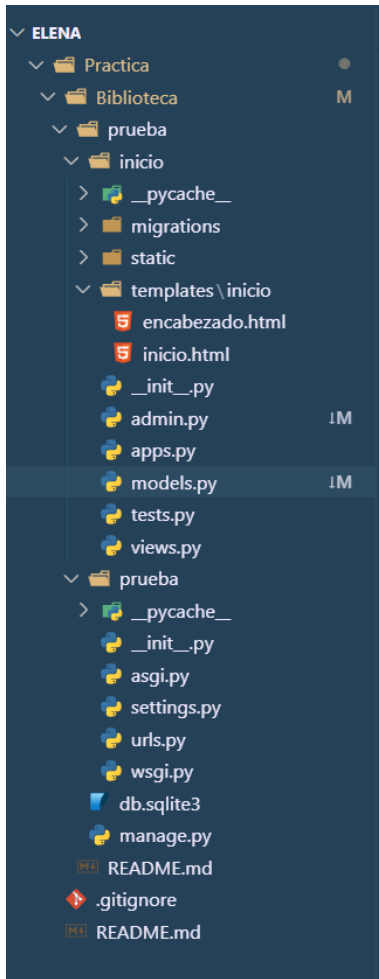


#### 3.2 Modelo Relacional.





### 3.3 Creación de proyecto. (Incluir una captura que muestre la estructura y contenido de su proyecto (carpetas), mostrar proyecto compartido en GitHub y Participantes)



### 3.4 Generación de modelos de datos “completos” (incluir códigos y capturas de envíos de actualizaciones por encargado de GitHub).

Modelos Jazzani (Libros y Grupos) :

```
class Libros(models.Model):
    id_libro = models.AutoField(verbose_name="ID: ", primary_key=True) # Usar AutoField en lugar de IntegerField
    ISBN = models.CharField(max_length=100, verbose_name="ISBN: ")
    nombre = models.CharField(max_length=100, verbose_name="Nombre: ")
    autor = models.CharField(max_length=100, verbose_name="Autor: ")
    año = models.DateField(verbose_name="Año: ") # Quitar max_length en DateField
    created = models.DateTimeField(auto_now_add=True) # Agregar campo created

ericksiu, 2 weeks ago | 1 author (ericksiu)
class Meta:
    verbose_name = "Libro"
    verbose_name_plural = "Libros"
    ordering = ["-created"]

def __str__(self):
    return self.nombre

You, 2 days ago | 2 authors (You and one other)
class Grupos (models.Model):
    id_grupo = models.IntegerField(max_length=10000, verbose_name="ID", primary_key=True)
    nombre = models.CharField(max_length=100, verbose_name="Nombre del grupo")
    privilegios = models.CharField(max_length=100, verbose_name="Privilegios")
    created = models.DateTimeField(auto_now_add=True) # Agregar campo created

You, 2 days ago | 1 author (You)
class Meta:
    verbose_name = "Grupo"
    verbose_name_plural = "Grupos"
    ordering = ["-created"]

def __str__(self):
    return self.nombre
```

Modelos Baruck (Descargas y Ver Libros):

```
models.py x admin.py
prueba > inicio > models.py > Ver_Libros > _str_
82 class Libros_Categorias(models.Model):
94     return str(self.id_categoria)
95
You, 5 minutes ago | 2 authors (You and one other)
96 class Descargas(models.Model):
97     id_descarga = models.AutoField(verbose_name="ID: ", primary_key=True) # Usar AutoField
98     id_libro = models.ForeignKey(Libros, on_delete=models.CASCADE, verbose_name="ID_Libro: ")
99     id_usuario = models.ForeignKey(Usuarios, on_delete=models.CASCADE, verbose_name="ID_Usuario")
100     fecha = models.DateField(auto_now_add=True, verbose_name="Fecha de descarga: ")
101     created = models.DateTimeField(auto_now_add=True) # Agregar campo created
102
You, 5 minutes ago | 1 author (You)
103 class Meta:
104     verbose_name = "Descarga"
105     verbose_name_plural = "Descargas"
106     ordering = ["-created"]
107
108 def __str__(self):
109     return str(self.id_descarga)
110
111
You, 5 minutes ago | 1 author (You)
112 class Ver_Libros(models.Model):
113     id_ver_libro = models.AutoField(verbose_name="ID: ", primary_key=True) # Usar AutoField
114     id_libro = models.ForeignKey(Libros, on_delete=models.CASCADE, verbose_name="ID_Libro: ")
115     id_usuario = models.ForeignKey(Usuarios, on_delete=models.CASCADE, verbose_name="ID_Usuario")
```

```

96 class Descargas(models.Model):
97
98     def _str_(self):
99         return str(self.id_descarga)
100
101
102 You, 5 minutes ago | 1 author (You)
103 class Ver_Libros(models.Model):
104     id_ver_libro = models.AutoField(verbose_name="ID: ", primary_key=True) # Usar AutoField
105     id_libro = models.ForeignKey(Libros, on_delete=models.CASCADE, verbose_name="ID_Libro: ")
106     id_usuario = models.ForeignKey(Usuarios, on_delete=models.CASCADE, verbose_name="ID_Usuario")
107     fecha = models.DateField(auto_now_add=True)
108     created = models.DateTimeField(auto_now_add=True) # Agregar campo created
109
110 You, 5 minutes ago | 1 author (You)
111 class Meta:
112     verbose_name = "Ver Libro"
113     verbose_name_plural = "Ver Libros"
114     ordering = ["-created"]
115
116     def _str_(self):
117         return str(self.id_ver_libro)
118
119 You, 5 minutes ago • Uncommitted changes

```

Modelos Nicolás (Categoría y Libros Categoría):

The screenshot shows a code editor with two Python files: `models.py` and `admin.py`. The `models.py` file contains the following code:

```

68 class Categorías(models.Model):
69     id_categoria = models.AutoField(verbose_name="ID: ", primary_key=True) # Usar AutoField
70     nombre = models.CharField(max_length=100, verbose_name="Nombre: ")
71     created = models.DateTimeField(auto_now_add=True) # Agregar campo created
72
73     class Meta:
74         verbose_name = "Categoría"
75         verbose_name_plural = "Categorías"
76         ordering = ["-created"]
77
78     def _str_(self):
79         return self.nombre
80
81
82 class Libros_Categorías(models.Model):
83     id_categoria = models.AutoField(verbose_name="ID: ", primary_key=True) # Usar AutoField
84     id_libro = models.ForeignKey(Libros, on_delete=models.CASCADE, verbose_name="Libros: ")
85     id_libro_categoria = models.ForeignKey(Categorías, on_delete=models.CASCADE, verbose_name="ID:")
86     created = models.DateTimeField(auto_now_add=True) # Agregar campo created
87
88     class Meta:
89         verbose_name = "Libros Categoría"
90         verbose_name_plural = "Libros Categorías"
91         ordering = ["-created"]
92
93     def _str_(self):
94         return str(self.id_categoria)
95
96
97
98
99

```

The `admin.py` file is also visible, showing the `admin.site.register` calls for `Categorías` and `Libros_Categorías`.

On the right side of the editor, there is a **SOURCE CONTROL** sidebar. It shows a commit history for the `Modelos-Categorías` repository. The commits are listed in reverse chronological order:

- Modelos-Categorías Z... (main)
- Agregando Modelos 2 LynareBlueven
- Merge branch 'main' of https://gi...
- Modelos Erick ericksiu
- Agregando Modelos Vol2 - Jazza...
- Merge pull request #2 from BIEN...
- Agregando Modelos ericksiu
- Merge pull request #1 from BIEN...
- Primer Actualización LynareBlueven
- Initial commit AxlBaruck

Below the commit history, there are tabs for **COMMITTS**, **BRANCHES**, **REMOTES**, **STASHES**, **TAGS**, **WORKTREES**, and **CONTRIBUTORS**.

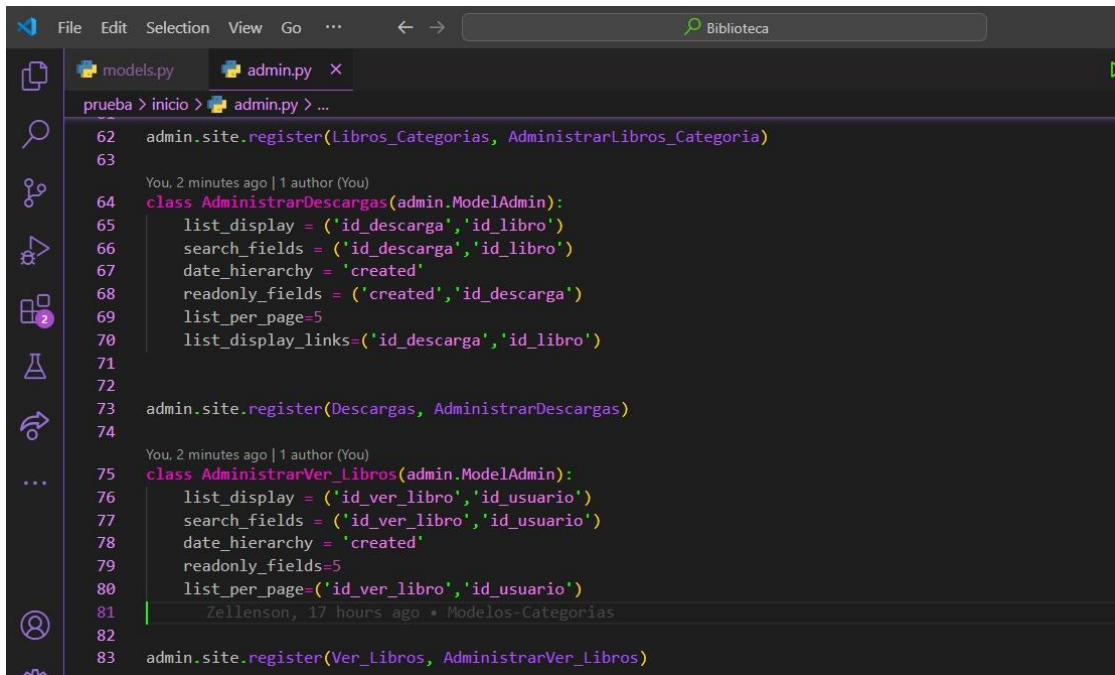
### Modelos Creados por Erick (Grupos y Reseñas):

```
23 class Grupos (models.Model):
24     id_grupo = models.IntegerField(max_length=10000, verbose_name="ID", primary_key=True)
25     nombre = models.CharField(max_length=100, verbose_name="Nombre del grupo")
26     privilegios = models.CharField(max_length=100, verbose_name="Privilegios")
27     created = models.DateTimeField(auto_now_add=True) # Agregar campo created
28
29     You, 2 weeks ago | 1 author (You)
30     class Meta:
31         verbose_name = "Grupo"
32         verbose_name_plural = "Grupos"
33         ordering = ["-created"]
34
35     def __str__(self):
36         return self.nombre
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56 class Reseñas(models.Model):
57     id_reseña = models.IntegerField(max_length=10000, verbose_name="ID: ", primary_key=True)
58     calificacion = models.IntegerField(max_length=1000, verbose_name="Calificación: ")
59     comentario = models.TextField(verbose_name="Comentario: ")
60
61     You, 2 weeks ago | 1 author (You)
62     class Meta:
63         verbose_name = "Reseña"
64         verbose_name_plural = "Reseñas"
65         ordering = ["-created"]
66
67     def __str__(self):
68         return self.comentario
```

3.5 Generación de accesos a modelos a través del admin, incluir mejora de visualización para el admin. Incluir capturas de envíos de actualizaciones por encargo de GitHub Admin Jazzani (Libros y Usuarios) :

```
admin.py X
Practica > Biblioteca > prueba > inicio > admin.py
1 from django.contrib import admin
2 from inicio.models import Libros, Usuarios
3
4 class AdministrarLibros(admin.ModelAdmin):
5     list_display = ('id_libro', 'nombre')
6     search_fields = ('id_libro', 'nombre')
7     date_hierarchy = 'created'
8     readonly_fields = ('created', 'id_libro')
9     list_per_page=5
10    list_display_links=('id_libro','nombre')
11
12
13    admin.site.register(Libros, AdministrarLibros)
14
15    class AdministrarUsuarios(admin.ModelAdmin):
16        list_display = ('id_usuario', 'nombre')
17        search_fields = ('id_usuario', 'nombre')
18        date_hierarchy = 'created'
19        readonly_fields = ('created', 'id_usuario')
20        list_display_links=('id_usuario','nombre')
21
22    admin.site.register(Usuarios, AdministrarUsuarios)
23
24
```

## Admin Axl (Descargas y Ver Libros) :



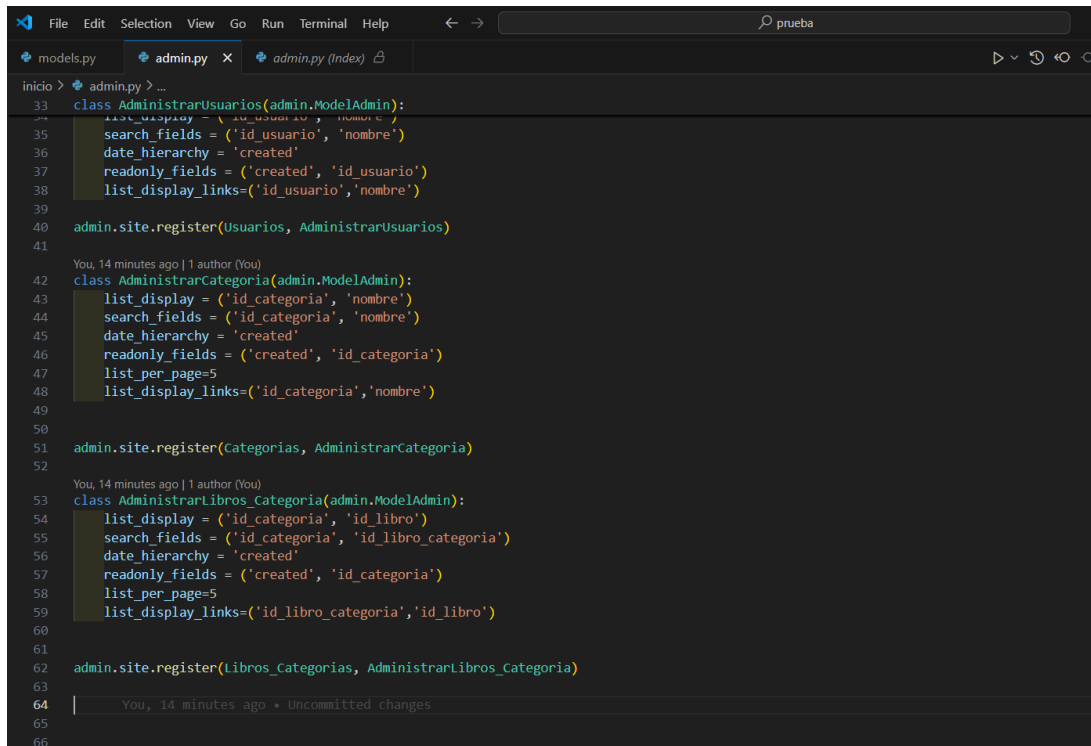
```
File Edit Selection View Go ... ← → Biblioteca

models.py admin.py x

prueba > inicio > admin.py > ...

62 admin.site.register(Libros_Categorias, AdministrarLibros_Categoria)
63
You, 2 minutes ago | 1 author (You)
64 class AdministrarDescargas(admin.ModelAdmin):
65     list_display = ('id_descarga', 'id_libro')
66     search_fields = ('id_descarga', 'id_libro')
67     date_hierarchy = 'created'
68     readonly_fields = ('created', 'id_descarga')
69     list_per_page=5
70     list_display_links=('id_descarga', 'id_libro')
71
72
73 admin.site.register(Descargas, AdministrarDescargas)
74
You, 2 minutes ago | 1 author (You)
75 class AdministrarVer_Libros(admin.ModelAdmin):
76     list_display = ('id_ver_libro', 'id_usuario')
77     search_fields = ('id_ver_libro', 'id_usuario')
78     date_hierarchy = 'created'
79     readonly_fields=5
80     list_per_page=('id_ver_libro', 'id_usuario')
81     Zellenson, 17 hours ago • Modelos-Categorias
82
83 admin.site.register(Ver_Libros, AdministrarVer_Libros)
```

## Admin Nicolás (Categorías y Libros Categorías):



```
File Edit Selection View Go Run Terminal Help ← → prueba

models.py admin.py x admin.py (Index)

inicio > admin.py > ...

33 class AdministrarUsuarios(admin.ModelAdmin):
34     list_display = ('id_usuario', 'nombre')
35     search_fields = ('id_usuario', 'nombre')
36     date_hierarchy = 'created'
37     readonly_fields = ('created', 'id_usuario')
38     list_display_links=('id_usuario', 'nombre')
39
40 admin.site.register(Usuarios, AdministrarUsuarios)
41
You, 14 minutes ago | 1 author (You)
42 class AdministrarCategoria(admin.ModelAdmin):
43     list_display = ('id_categoria', 'nombre')
44     search_fields = ('id_categoria', 'nombre')
45     date_hierarchy = 'created'
46     readonly_fields = ('created', 'id_categoria')
47     list_per_page=5
48     list_display_links=('id_categoria', 'nombre')
49
50
51 admin.site.register(Categorias, AdministrarCategoria)
52
You, 14 minutes ago | 1 author (You)
53 class AdministrarLibros_Categoria(admin.ModelAdmin):
54     list_display = ('id_categoria', 'id_libro')
55     search_fields = ('id_categoria', 'id_libro_categoria')
56     date_hierarchy = 'created'
57     readonly_fields = ('created', 'id_categoria')
58     list_per_page=5
59     list_display_links=('id_libro_categoria', 'id_libro')
60
61
62 admin.site.register(Libros_Categorias, AdministrarLibros_Categoria)
63
64 You, 14 minutes ago • Uncommitted changes
65
66
```