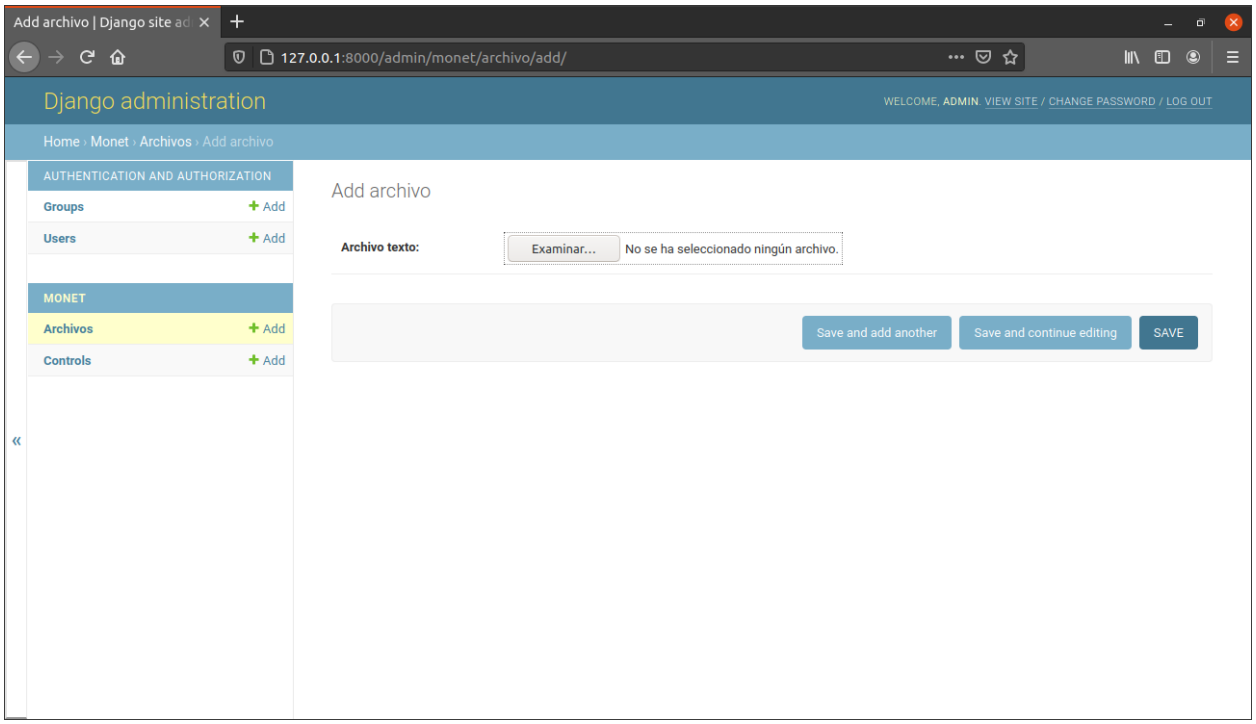
Respuesta Prueba Técnica Monet

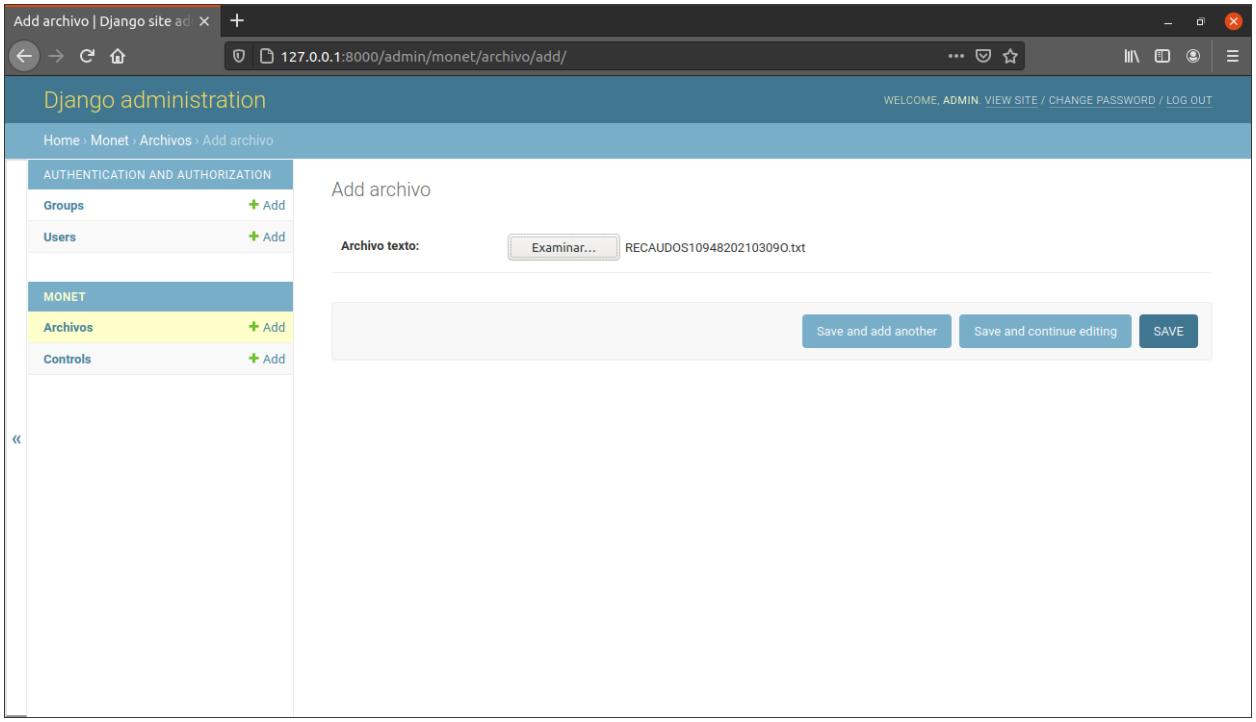
Se adjunta la descripción del proyecto Django realizado para la prueba técnica Monet.

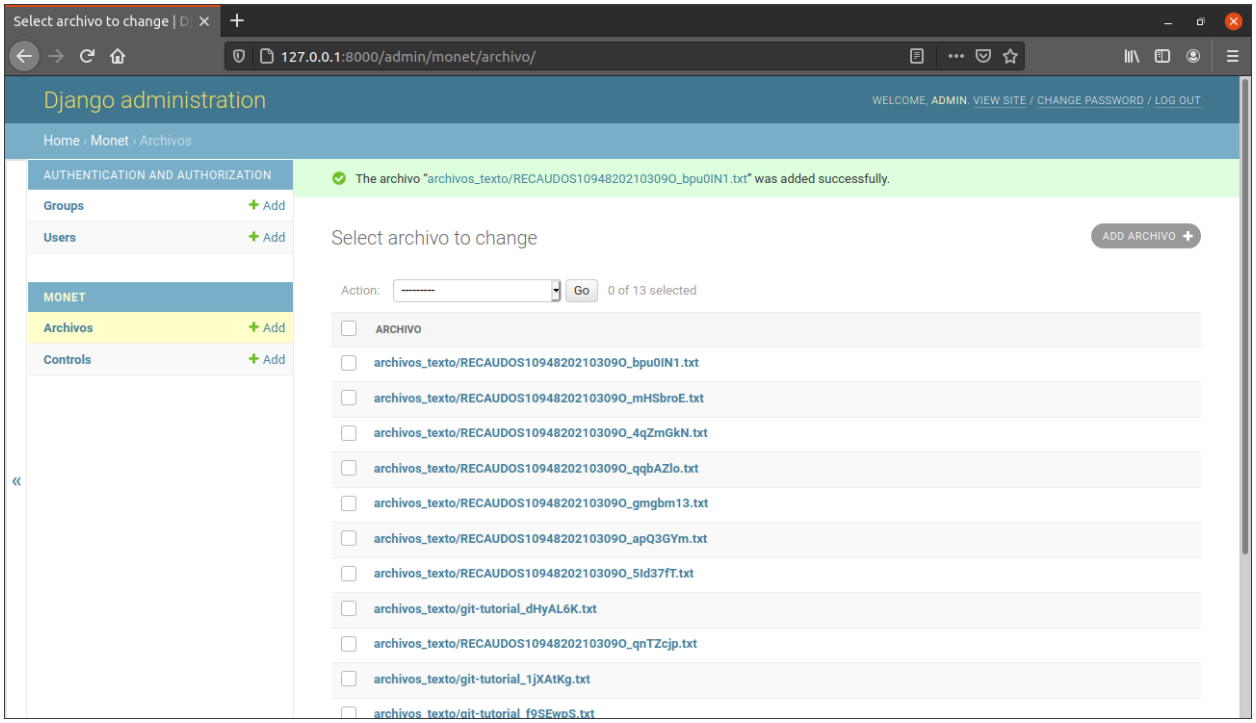
Requerimientos de Aplicación Django

1. Leer un archivo plano y extraer la información usando un Django Command.

Se creó dentro del Django Admin la funcionalidad para poder cargar los archivos.

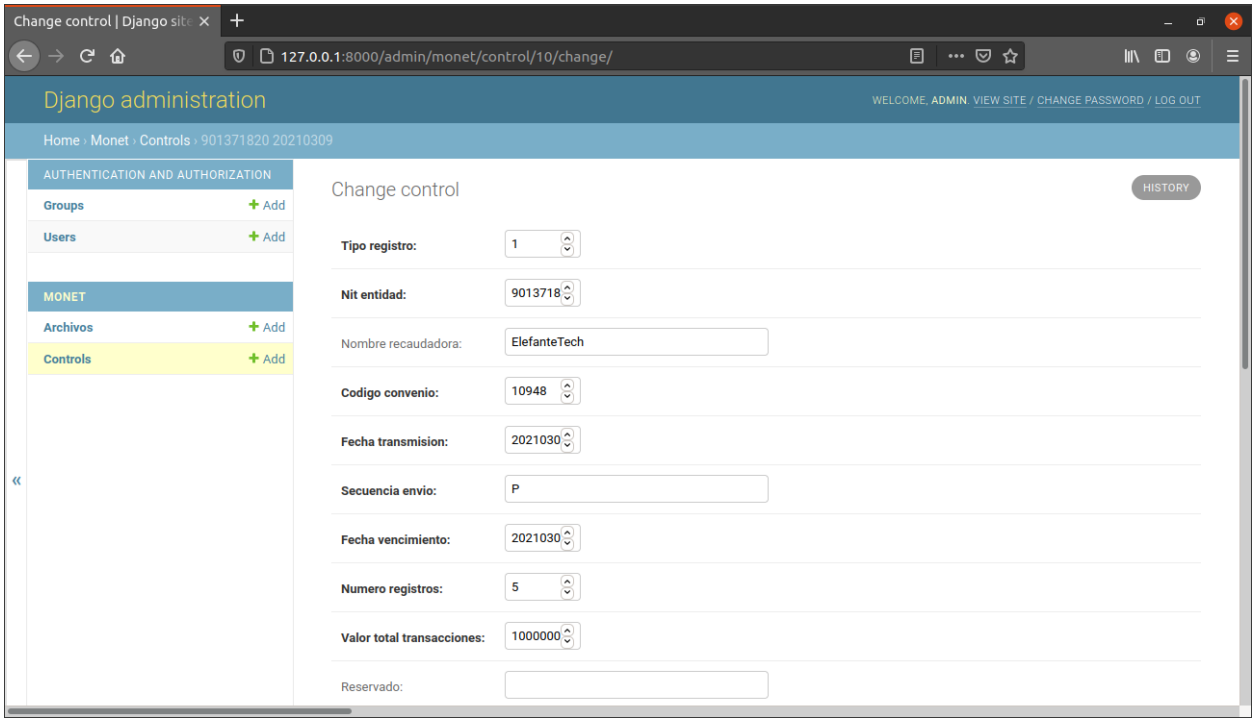


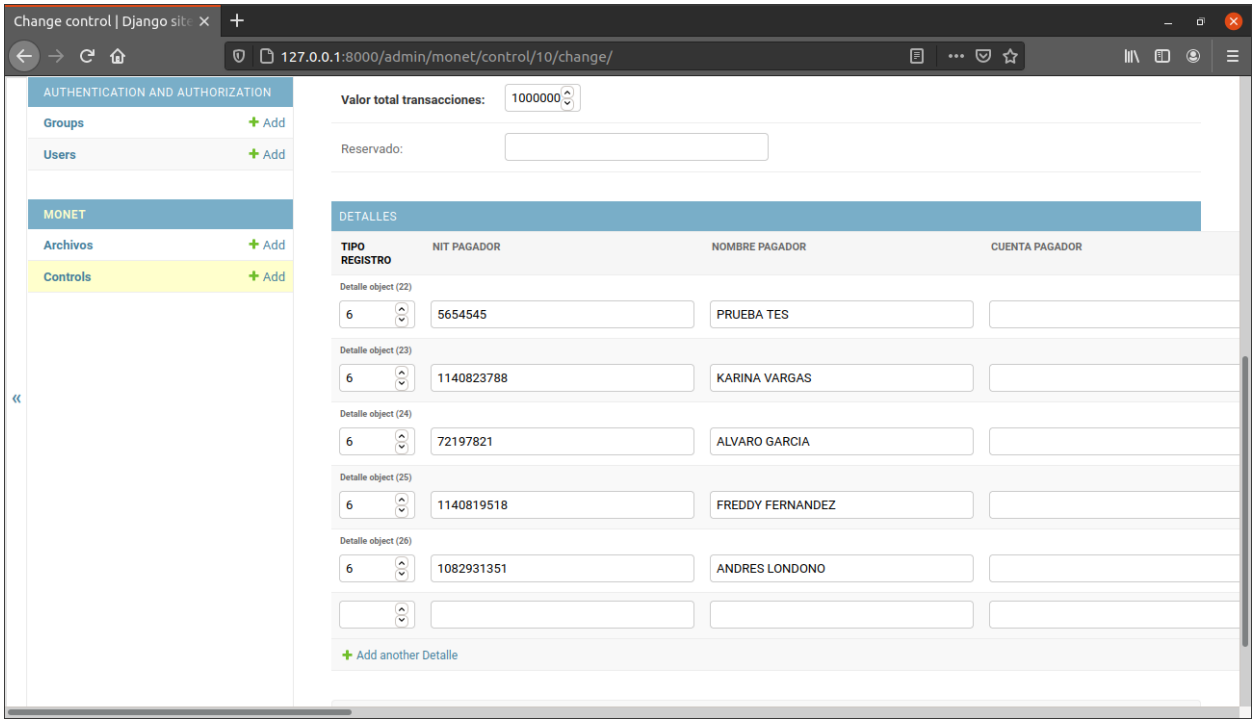




2. Guardar la información del archivo plano en la base de datos.

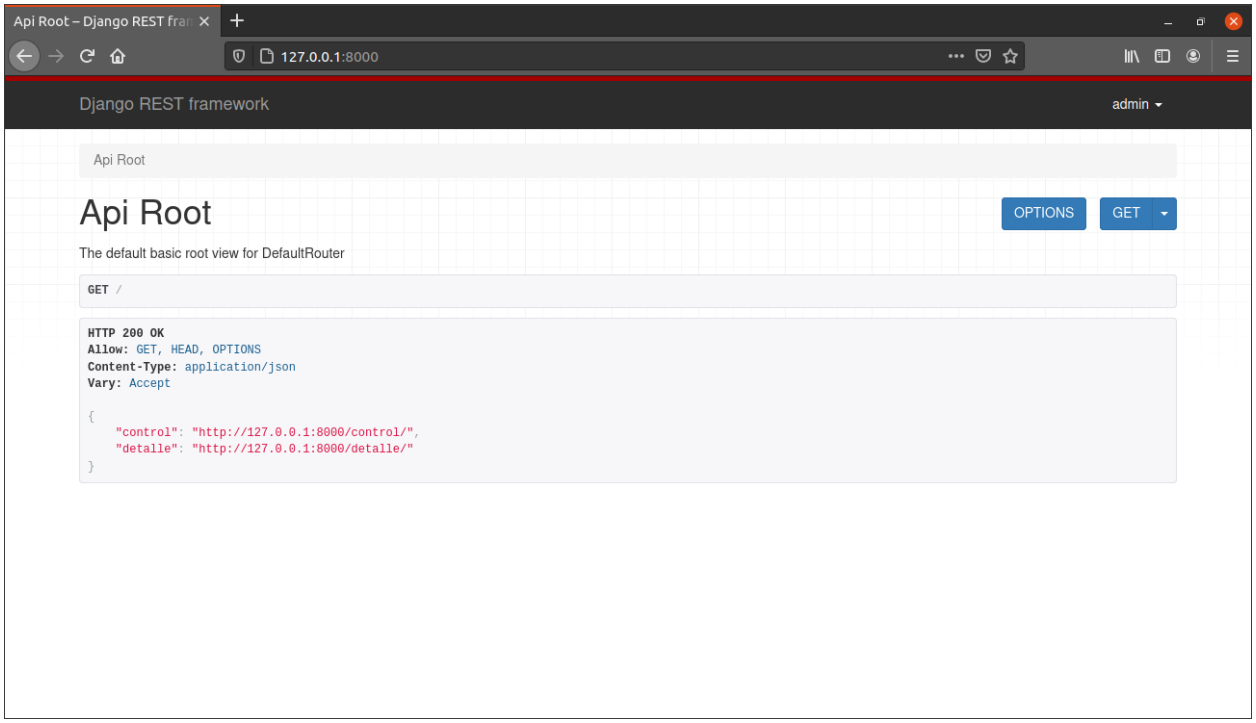
Luego de subir y guardar el archivo, los datos se extraen del archivo y quedan almacenados en la base de datos.

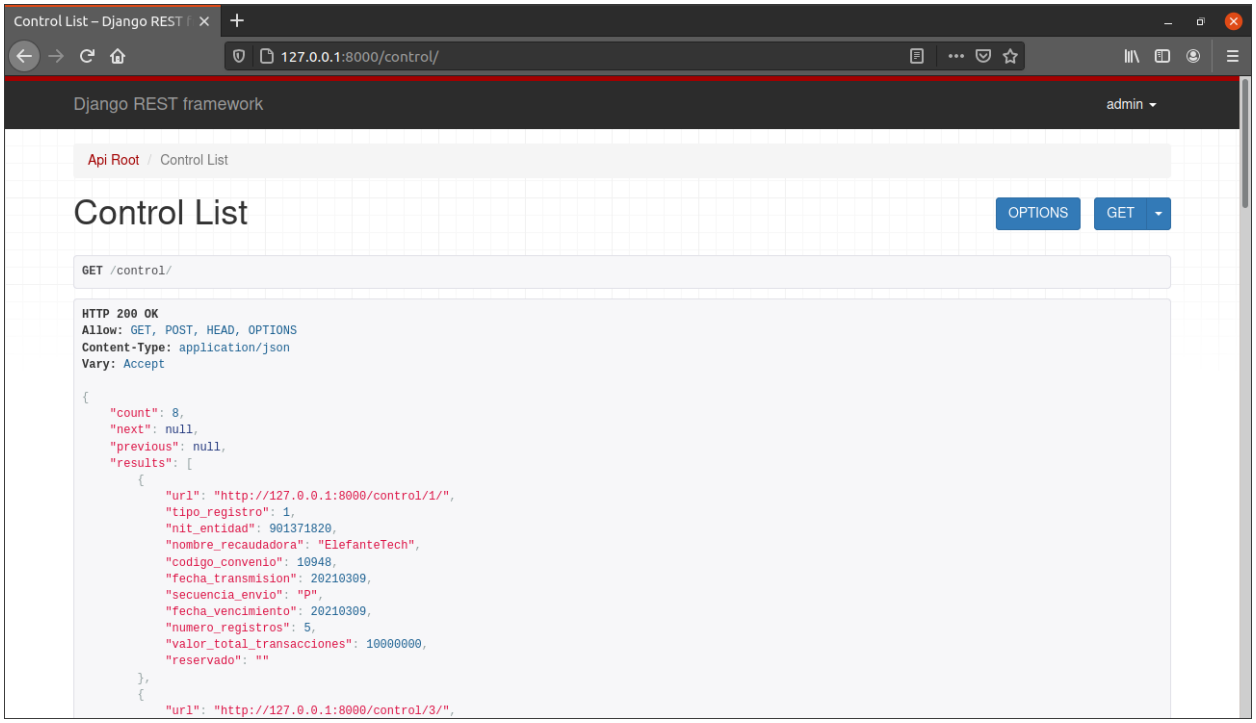


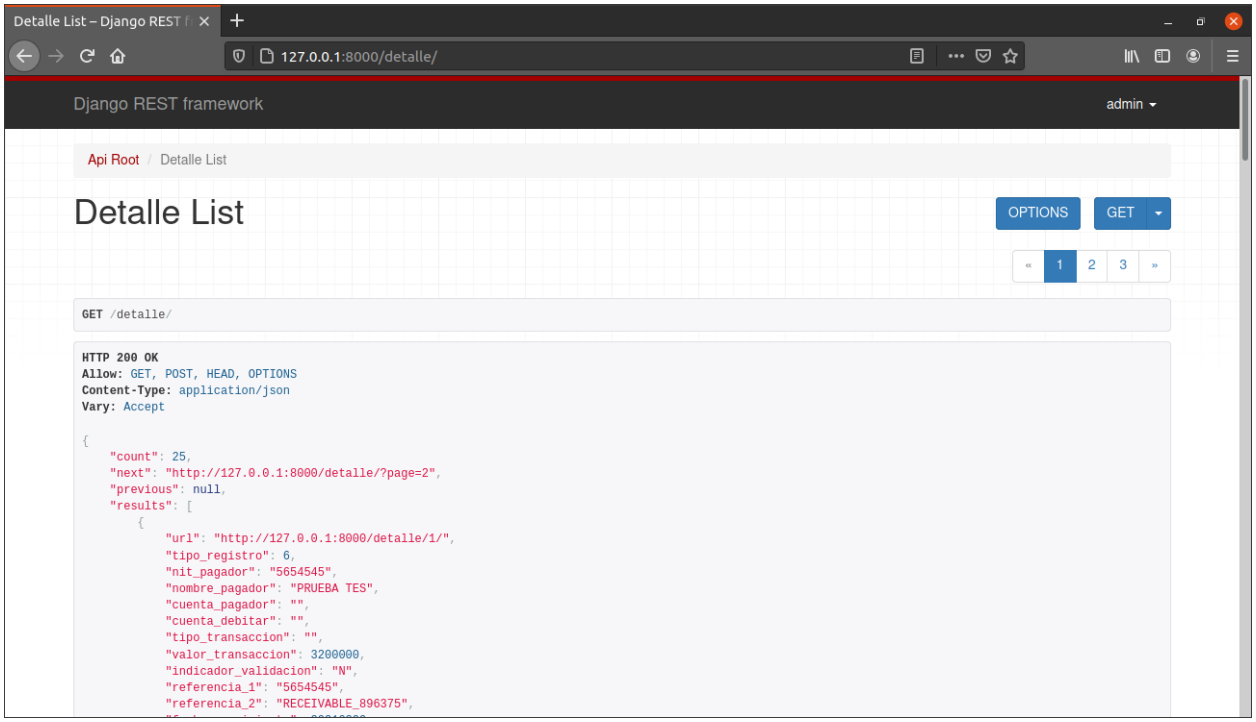


3. Crear un REST endpoint que devuelva la información del archivo guardada en la base de datos.

Se creó un REST endpoint utilizando Django REST framework para los modelos Control y Detalle los cuales contienen los datos extraídos del archivo de texto.

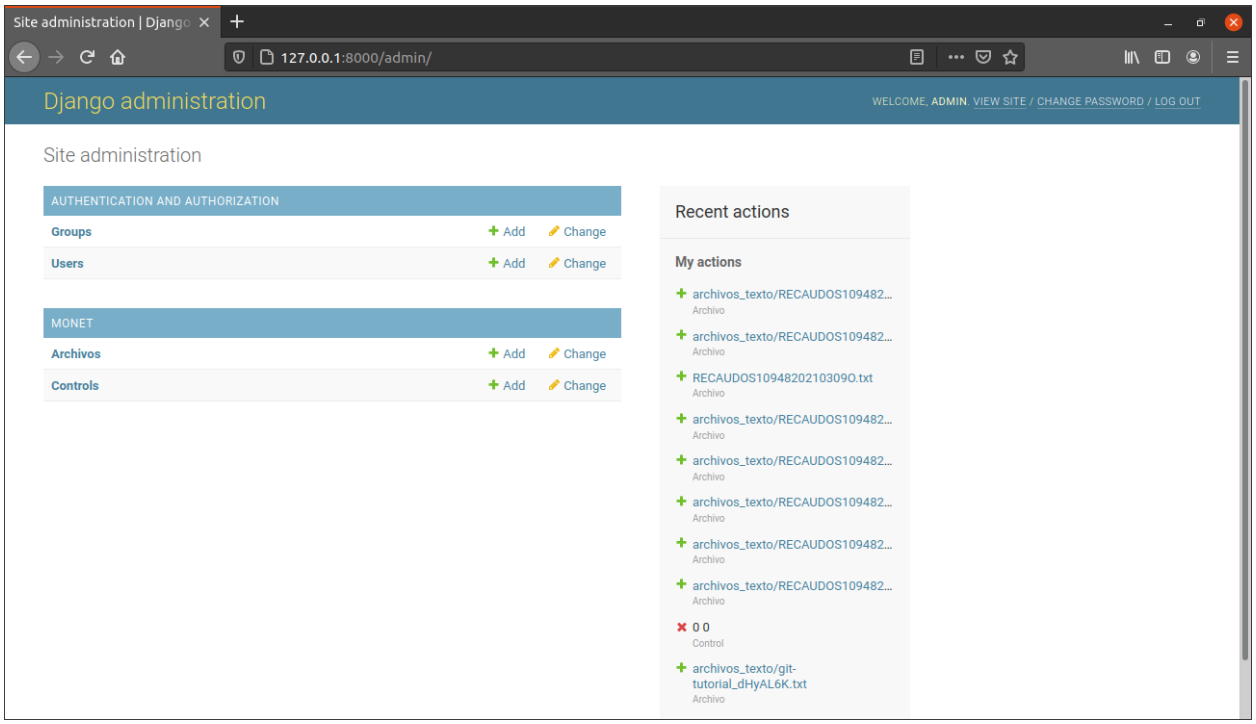

Api Root – Django REST fran
127.0.0.1:8000
Django REST framework
admin
Api Root
Api Root
OPTIONS
GET
The default basic root view for DefaultRouter
GET /
HTTP 200 OK
Allow: GET, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept
{
"control": "http://127.0.0.1:8000/control/",
"detalle": "http://127.0.0.1:8000/detalle/"
}

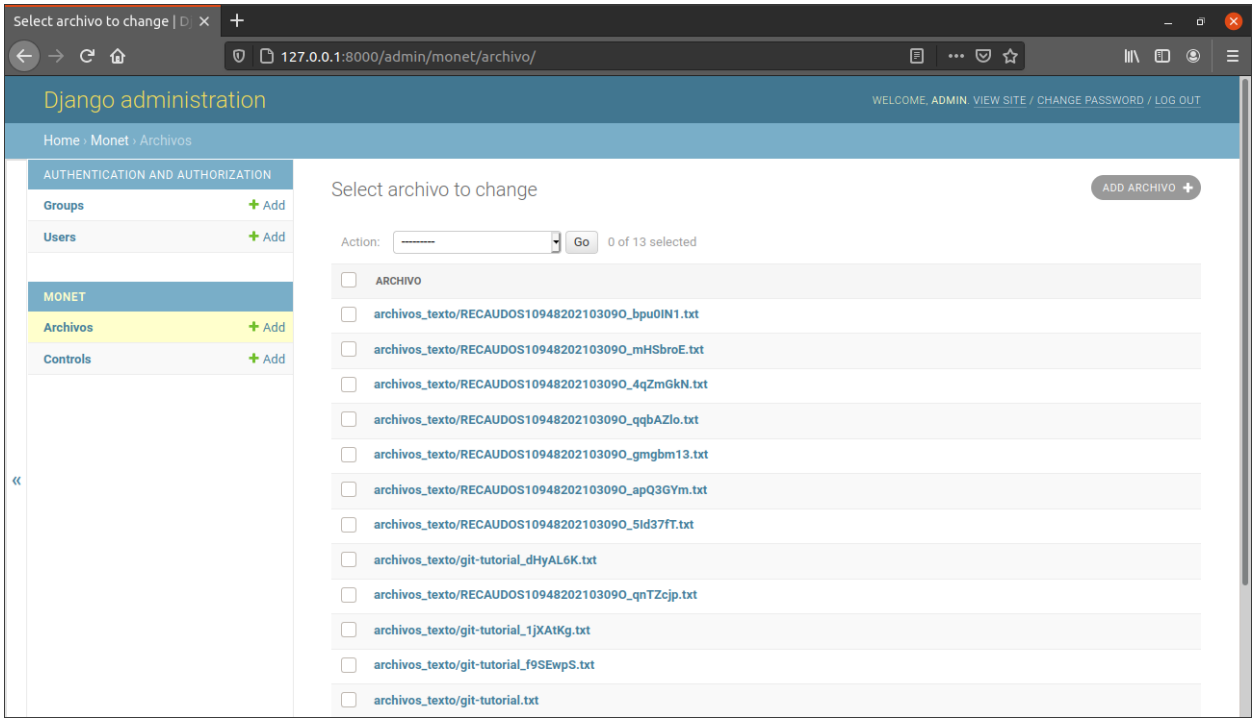

Control List – Django REST f
127.0.0.1:8000/control/
Django REST framework
admin
Api Root / Control List
Control List
OPTIONS
GET
GET /control/
HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept
{
"count": 8,
"next": null,
"previous": null,
"results": [
{
"url": "http://127.0.0.1:8000/control/1/",
"tipo_registro": 1,
"nit_entidad": 901371820,
"nombre_recaudadora": "ElefanteTech",
"codigo_convenio": 10948,
"fecha_transmision": 20210309,
"secuencia_envio": "P",
"fecha_vencimiento": 20210309,
"numero_registros": 5,
"valor_total_transacciones": 10000000,
"reservado": ""
},
{
"url": "http://127.0.0.1:8000/control/3/",

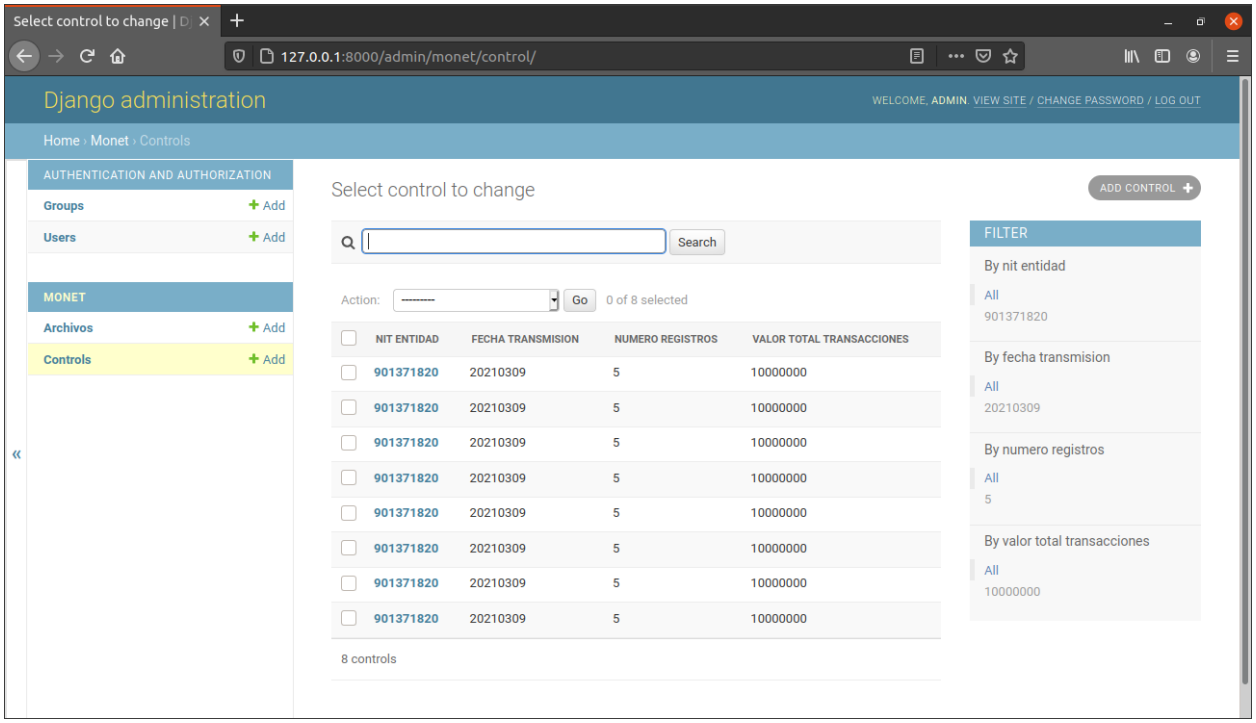


4. Poder ver y editar la información desde el Django Admin.

Se crearon todas las visualizaciones para los models Control, Detalle y Archivo desde el Django Admin, permitiendo editar los datos desde la interfaz.




Select archivo to change | D
127.0.0.1:8000/admin/monet/archivo/
Django administration
WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Monet › Archivos
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
MONET
Archivos + Add
Controls + Add
Select archivo to change
ADD ARCHIVO +
Action: -------- Go 0 of 13 selected
ARCHIVO
archivos_texto/RECAUDOS10948202103090_bpu0IN1.txt
archivos_texto/RECAUDOS10948202103090_mHSbroE.txt
archivos_texto/RECAUDOS10948202103090_4qZmGkN.txt
archivos_texto/RECAUDOS10948202103090_qqbAZlo.txt
archivos_texto/RECAUDOS10948202103090_gmgbm13.txt
archivos_texto/RECAUDOS10948202103090_apQ3GYm.txt
archivos_texto/RECAUDOS10948202103090_5Id37fT.txt
archivos_texto/git-tutorial_dHyAL6K.txt
archivos_texto/RECAUDOS10948202103090_qnTZcjp.txt
archivos_texto/git-tutorial_1jXAtKg.txt
archivos_texto/git-tutorial_f9SEwpS.txt
archivos_texto/git-tutorial.txt


Select control to change | D
127.0.0.1:8000/admin/monet/control/
Django administration
WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Monet › Controls
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
MONET
Archivos + Add
Controls + Add
Select control to change
ADD CONTROL +
Search
Action: -------- Go 0 of 8 selected
NIT ENTIDAD FECHA TRANSMISION NUMERO REGISTROS VALOR TOTAL TRANSACCIONES
901371820 20210309 5 10000000
8 controls
FILTER
By nit entidad
All
901371820
By fecha transmision
All
20210309
By numero registros
All
5
By valor total transacciones
All
10000000

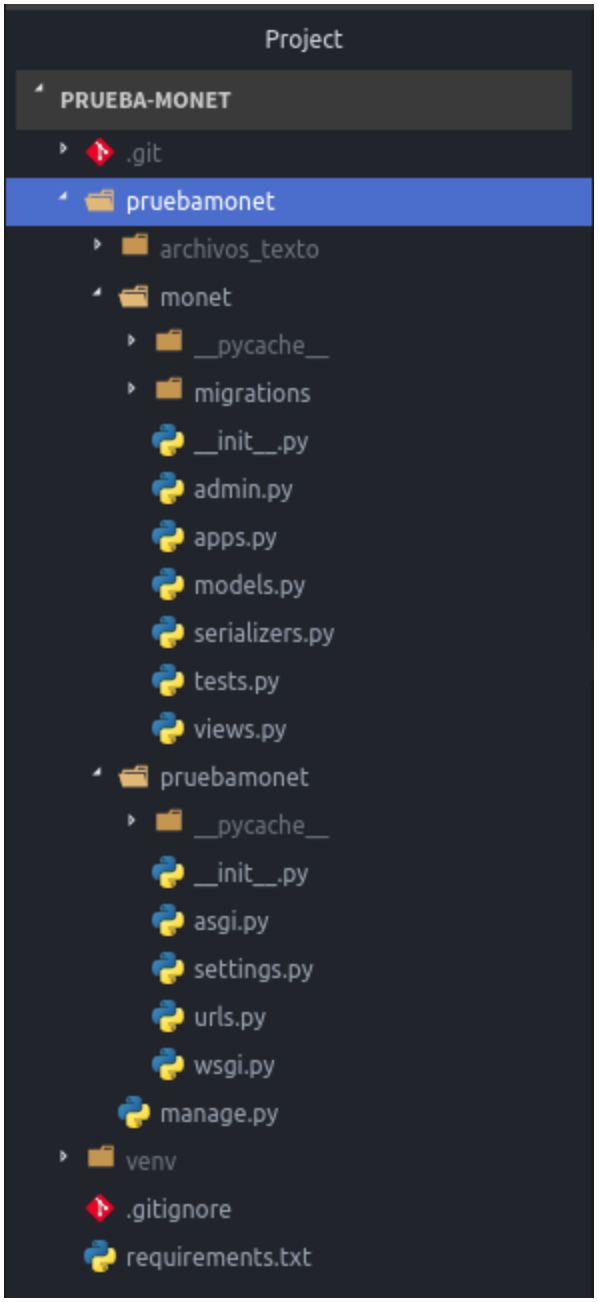
Recursos

- Archivo plano: https://gist.github.com/dco5/38fba2250170d18700f0e9524f1b38df#file-recaudos1094820210309o-txt
- Descripción de archivo plano: https://docs.google.com/spreadsheets/d/12QZtBMvKUyUxYc1Oj9w_bGtcJP7RcxJdnYqlx4-ljp0/edit#gid=0
- El valor de dinero en el archivo plano incluye los dos decimales, es decir $10,000 == 1000000

Puntos a Evaluar:

1. Estructura del proyecto.

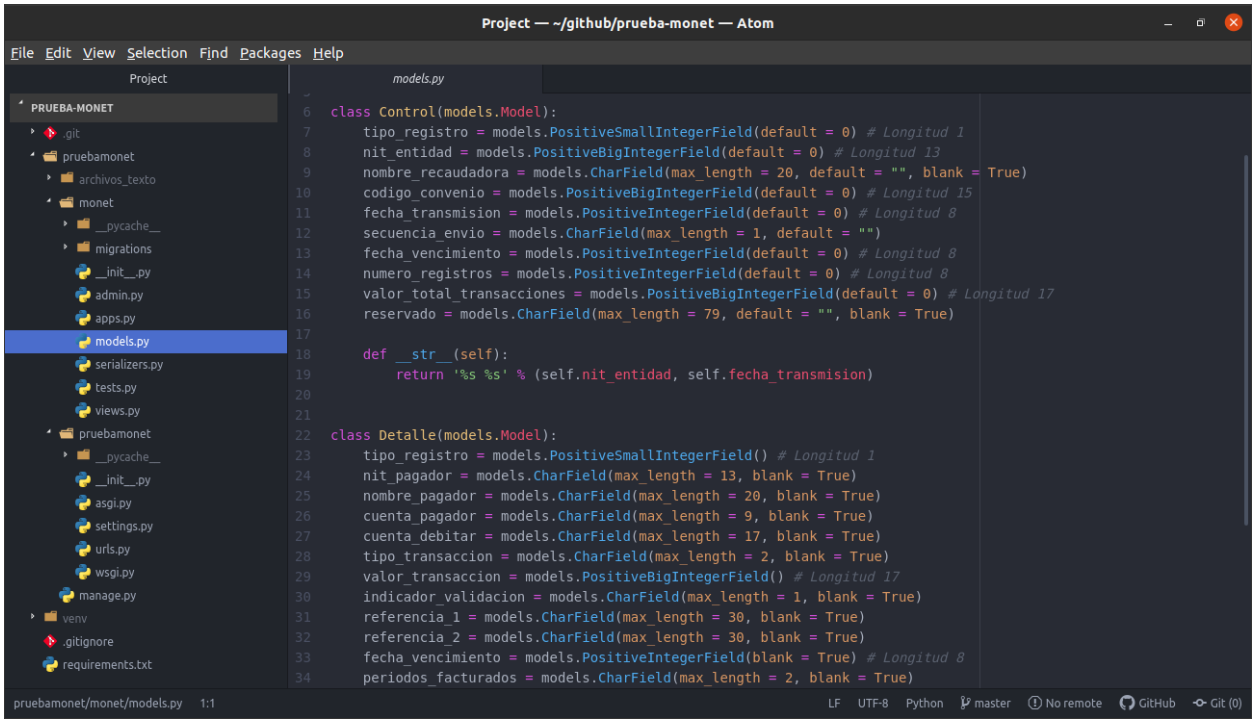
El proyecto se creó utilizando Django, se utilizó ambiente virtual e instalación de paquetes con requirements y pip, se creó una app llamada monet, se utilizó una base de datos Postgresql en lugar de SQLite3, se utilizó control de versiones.

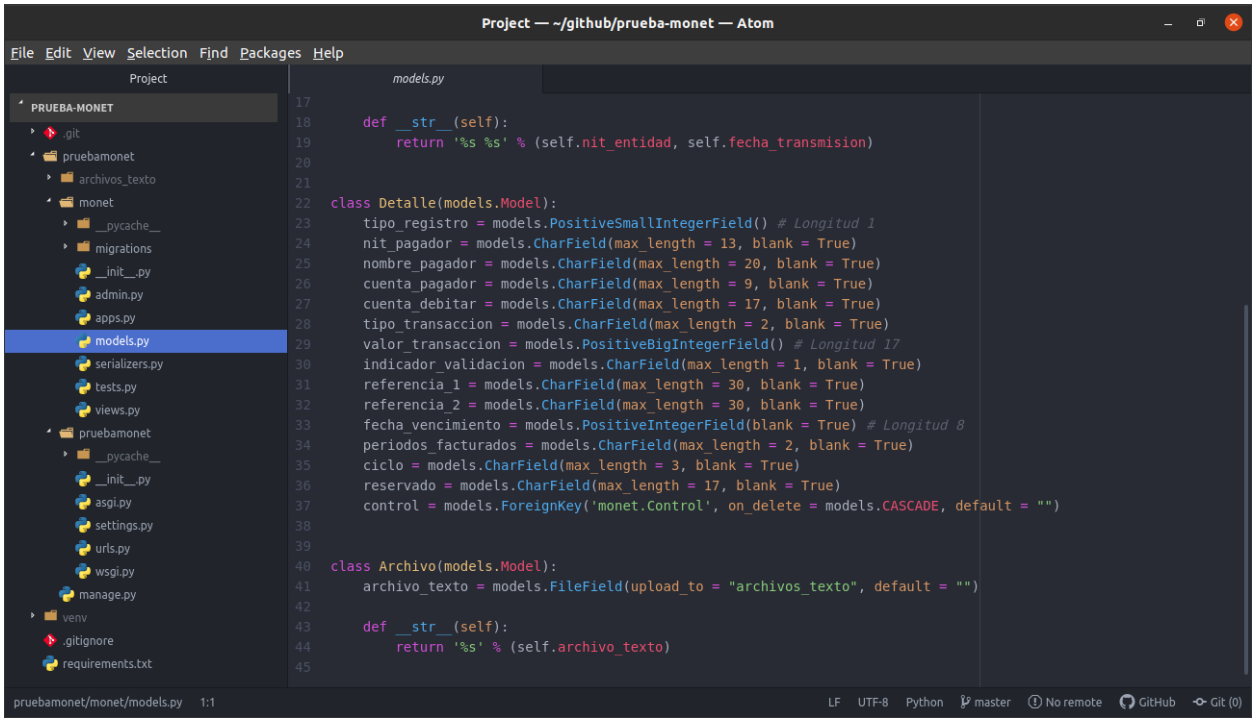

Project
PRUEBA-MONET
.git
pruebamonet
archivos_texto
monet
__pycache__
migrations
__init__.py
admin.py
apps.py
models.py
serializers.py
tests.py
views.py
pruebamonet
__pycache__
__init__.py
asgi.py
settings.py
urls.py
wsgi.py
manage.py
venv
.gitignore
requirements.txt

```
f46e484 (HEAD -> master) Leer archivo plano y popular base de datos
8b2cdeb Correccion en model Archivo
db200ee Creacion REST endpoint usando Django REST framework
fdb1869 Mejora Django admin
a8c3100 Actualizacion Control y Detalle
e34f11a Adicion modelo Archivo
8f01ce1 Adicion Django Admin incluyendo TabularInline
3b85382 Crear modelos Control y Detalle
775daa7 Creacion app monet
e50fb8e Cambio a Postgresql
2e3b7c2 Initial commit
```

2. Estructura de los modelos.

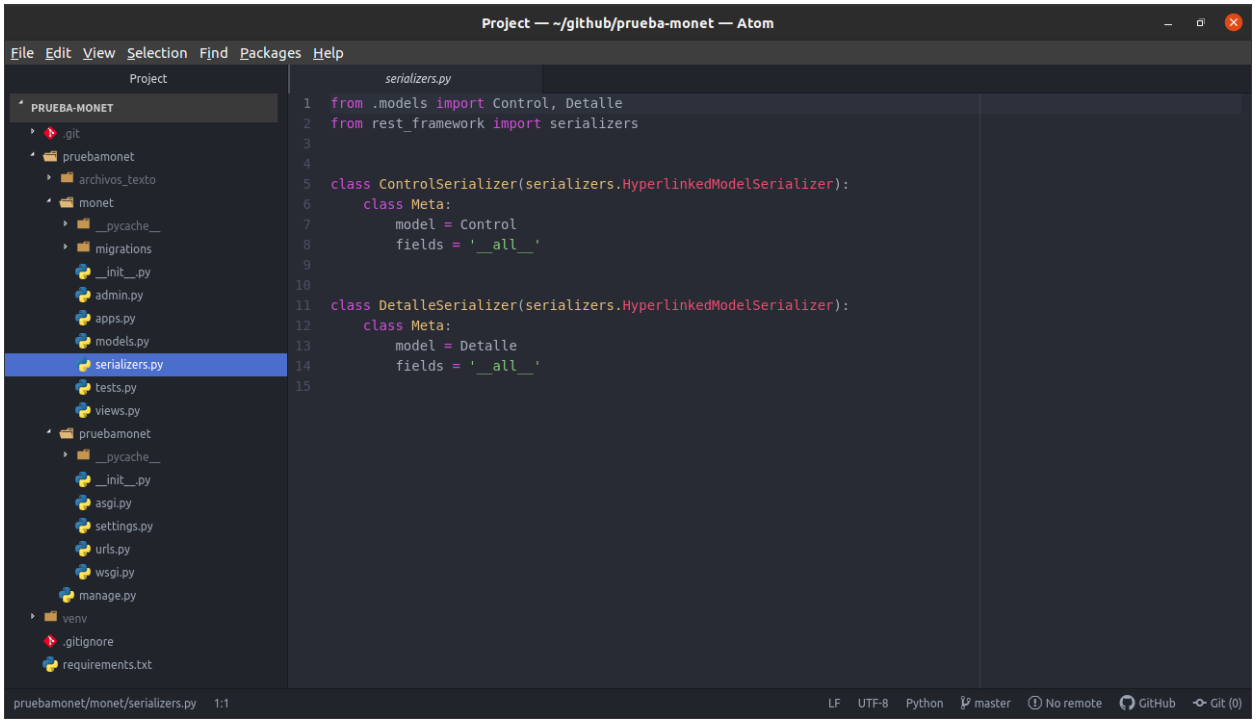
Los modelos se componen del modelo Control, el cual contiene la información de la primera línea del archivo, y el modelo Detalle, el cual contiene la información de las siguientes líneas. Pueden existir varias líneas del modelo Detalle, las cuales dependen de Control, por lo cual están conectadas por un ForeignKey. Hay un modelo Archivo que recibe y almacena los archivos txt subidos.

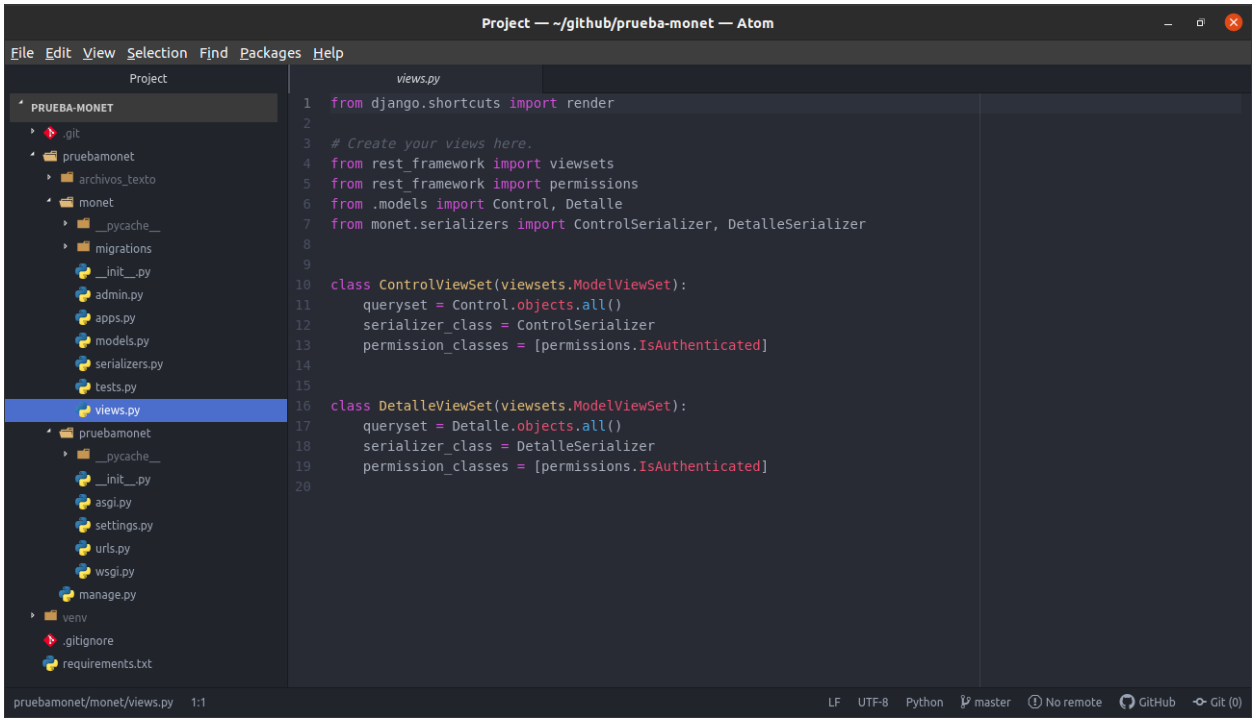


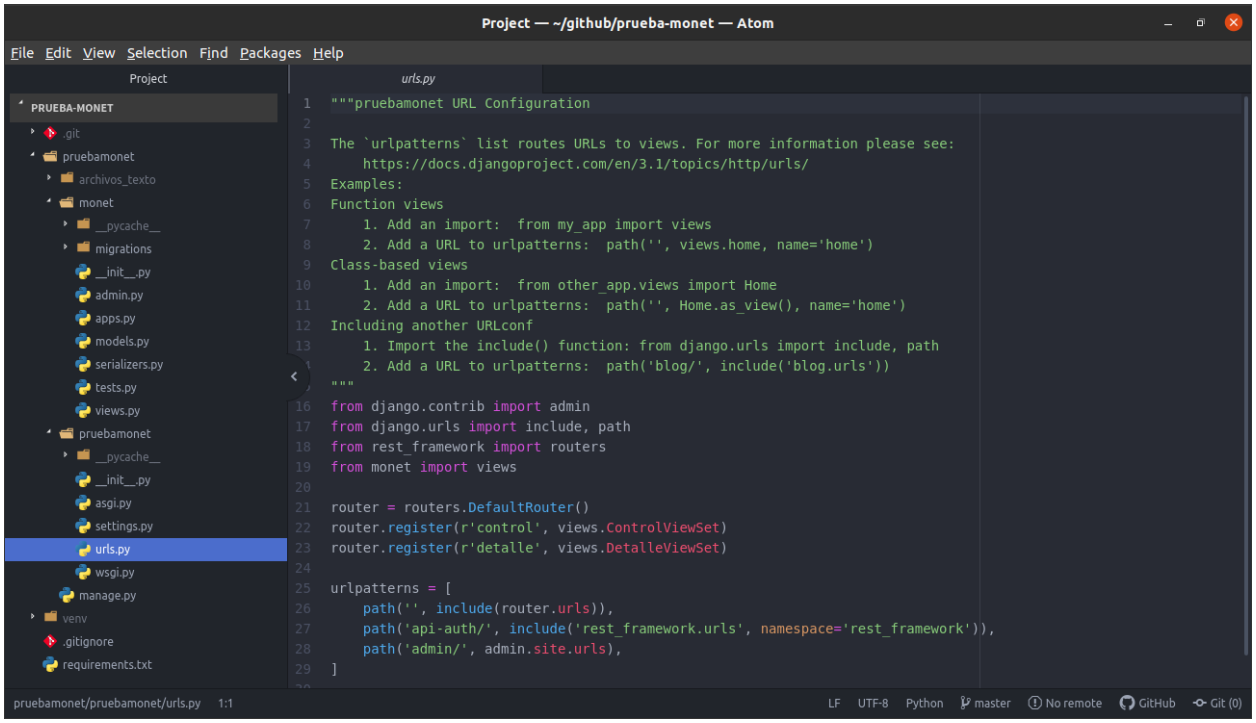


3. Implementación de REST endpoint.

Se implementó utilizando Django REST framework.







4. Manejo de excepciones.

No realizado.

5. Cobertura de tests.

No realizado.

6. Bonus: Setup de Docker compose para correr la aplicación.

No realizado.