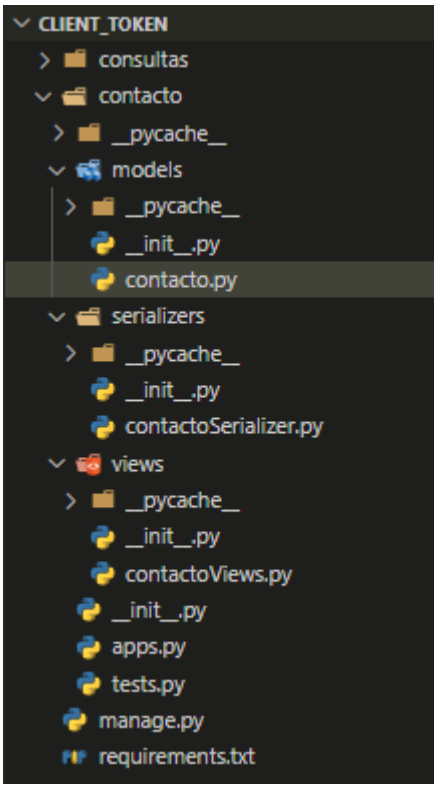
Segundo Microservicio:
Link del Repositorio: https://github.com/DavidCastro88/4a-ms2
Link de Despliegue: https://misiontic-g02.herokuapp.com
Framework: Django **Base de Datos:** MongoDB
Atención al Cliente: Por medio de un formulario donde se le pregunta al cliente la razón de su inquietud junto con sus datos personales.



Modelo:

```python
from djongo import models

class Contacto(models.Model):
    inquietudes = (
        (0, "Producto"),
        (1, "Servicio de entrega"),
        (2, "Otro"),
    )
    _id = models.ObjectIdField(primary_key= True)
    nombre = models.CharField(max_length=50, null=True)
    telefono = models.IntegerField(null=True)
    email = models.EmailField(max_length=50, null=True)
    inquietudes = models.CharField(max_length=2, choices=inquietudes, null=True)
    comentario = models.TextField(max_length=2000, null=True)
    sol_activa = models.BooleanField(default=True, null=True)
```

Serializer:

```python
from django.db.models import fields
from rest_framework import serializers
from contacto.models import Contacto

class contactoSerializer(serializers.ModelSerializer):
    class Meta:
        model = Contacto
        fields = '__all__'

    def to_representation(self, instance):
        return {
            "_id": str(instance._id),
            "nombre": instance.nombre,
            "email": instance.email,
            "telefono": instance.telefono,
            "inquietudes": instance.inquietudes,
            "comentario": instance.comentario,
            "sol_activa": instance.sol_activa,
        }
```

Views:

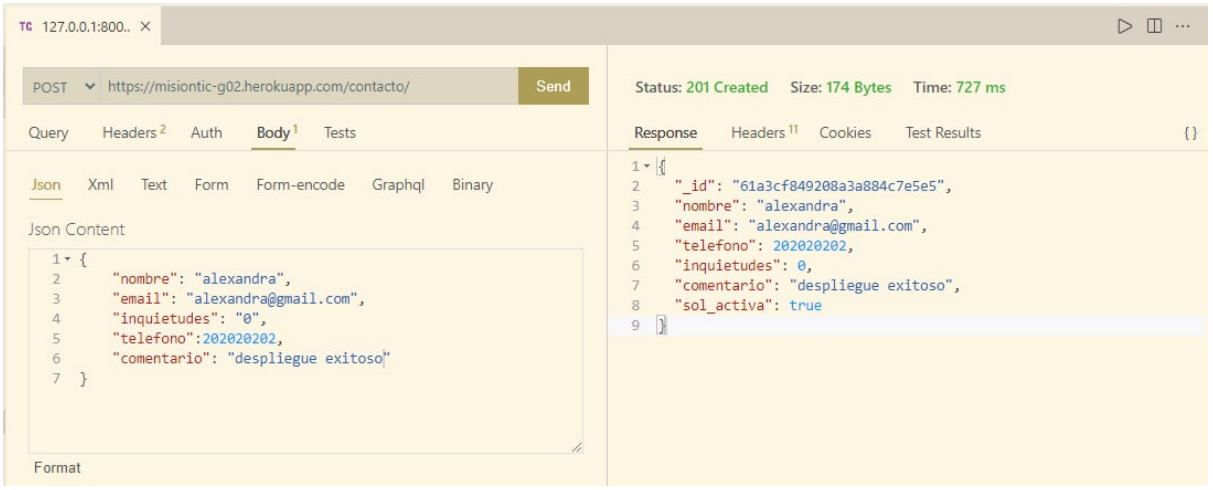
```python
from contacto.models import Contacto
from contacto.serializers import contactoSerializer
from rest_framework import generics, status
from django.shortcuts import get_object_or_404
from rest_framework.response import Response
from bson.objectid import ObjectId
from django.core import serializers
import json

class contactoCreate(generics.ListCreateAPIView):
    serializer_class = contactoSerializer
    queryset = Contacto.objects.all()

class contactoDetails(generics.RetrieveAPIView):
    serializer_class = contactoSerializer
    def get(self, request, *args, **kwargs):
        """
        se crea un objecto de tipo objectid para filtrar por la llave en mongo atlas,
        luego se trae el objeto y se serializa para enviar luego los campos necesarios
        """
        contacto = get_object_or_404(Contacto, _id=ObjectId(kwargs["_id"]))
        serialized_obj = serializers.serialize('json', (contacto,))
        struct = json.loads(serialized_obj)
        return Response(struct[0]["fields"], status=status.HTTP_200_OK)

class filterByEmail(generics.ListAPIView):
    serializer_class = contactoSerializer
    def get_queryset(self):
        queryset = Contacto.objects.filter(email=self.kwargs["email"])
        return queryset
```

URLS:

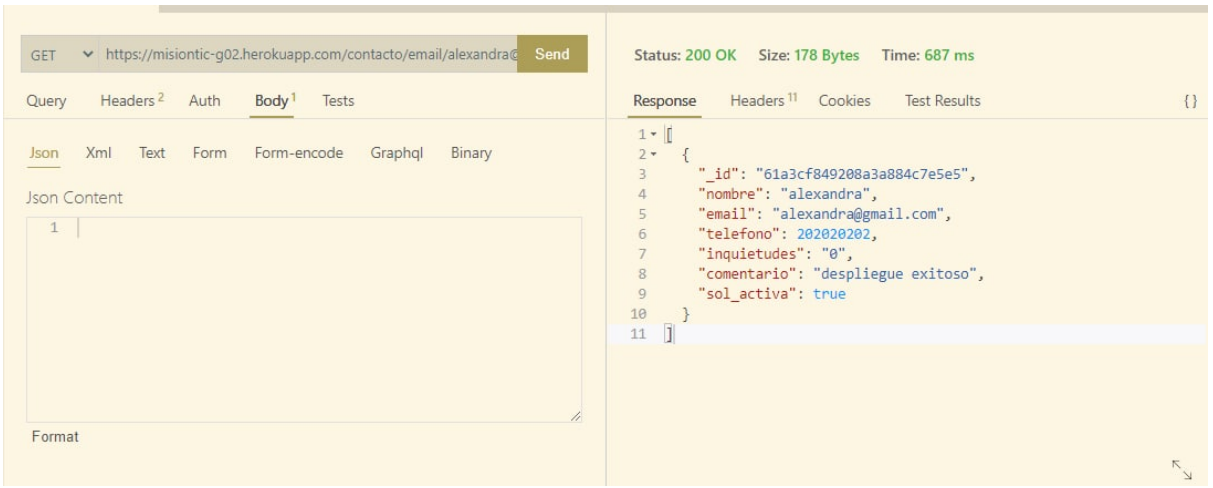
```python
from django.urls import path
from contacto import views

urlpatterns = [
    path('contacto/', views.contactoCreate.as_view()), # Crear una sol de contacto o listar todas
    path('contacto/<str:_id>', views.contactoDetails.as_view()), # Obtener una solicitud por id, modificar o borrar
    path('contacto/email/<str:email>', views.filterByEmail.as_view()), # Obtener solicitudes por email
]
```

Pruebas en los endpoints:

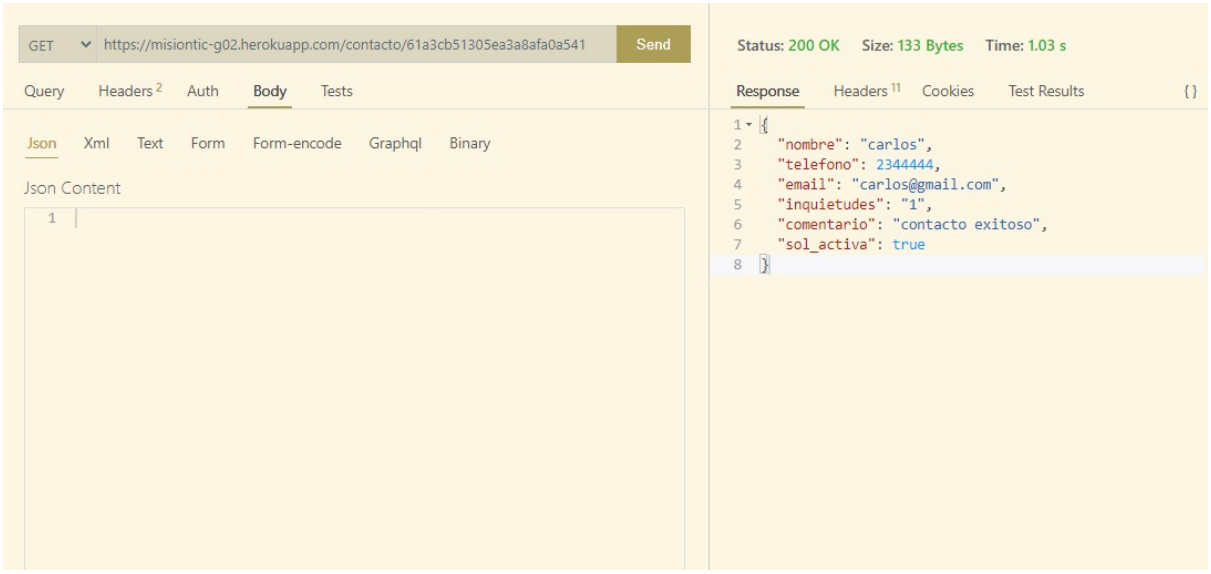
Post:



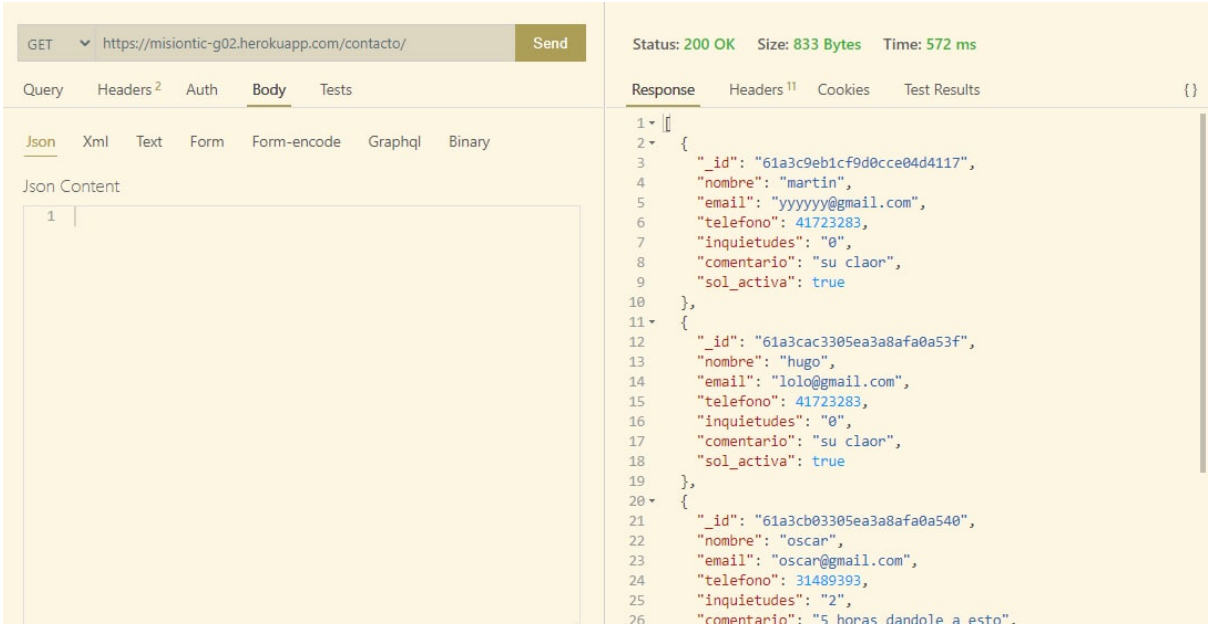
Filtrar por correo electrónico:



Filtrar por ID:



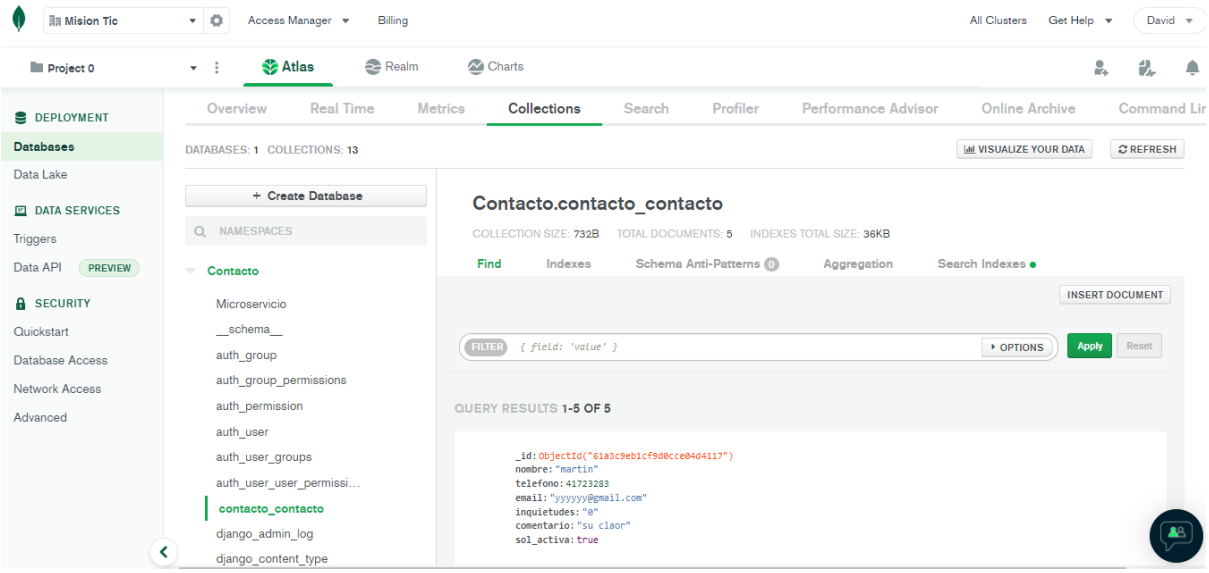
Get de todos los contactos:



Conexión con la base de datos de MONGO-Atlas:

```
85
86  DATABASES = {
87      # 'default': {
88      #     'ENGINE': 'django.db.backends.sqlite3',
89      #     'NAME': BASE_DIR / 'db.sqlite3',
90      # }
91      'default': {
92          'ENGINE': 'djongo',
93          'NAME': 'Contacto',
94          'ENFORCE_SCHEMA': False,
95          'CLIENT': {
96              'host': 'mongodb+srv://DavidC:123U5678@cluster0.ndlt2.mongodb.net/myFirstDatabase?retryWrites=true&w=majority',
97              "authMechanism": "SCRAM-SHA-1",
98          }
99      }
100 }
101
```

Atlas con las datos:




```
_id: ObjectId("61a3cb03305ea3a8afa0a540")
nombre: "oscar"
telefono: 31489393
email: "oscar@gmail.com"
inquietudes: "2"
comentario: "5 horas dandole a esto"
sol_activa: true
```

```
_id: ObjectId("61a3cb51305ea3a8afa0a541")
nombre: "carlos"
telefono: 2344444
email: "carlos@gmail.com"
inquietudes: "1"
comentario: "contacto exitoso"
sol_activa: true
```

```
_id: ObjectId("61a3cf849208a3a884c7e5e5")
nombre: "alexandra"
telefono: 202020202
email: "alexandra@gmail.com"
inquietudes: "0"
comentario: "despliegue exitoso"
sol_activa: true
```

EVIDENCIA DE TRABAJO EN EL JIRA: