Suspicious Activity Recognition

***Proyecto de Desarrollo***

El archivo que contiene la app se llama Simulador.py, en él está contenida la información del servidor que se levanta con flask-socketio en el puerto 5007.



Código Fuente:

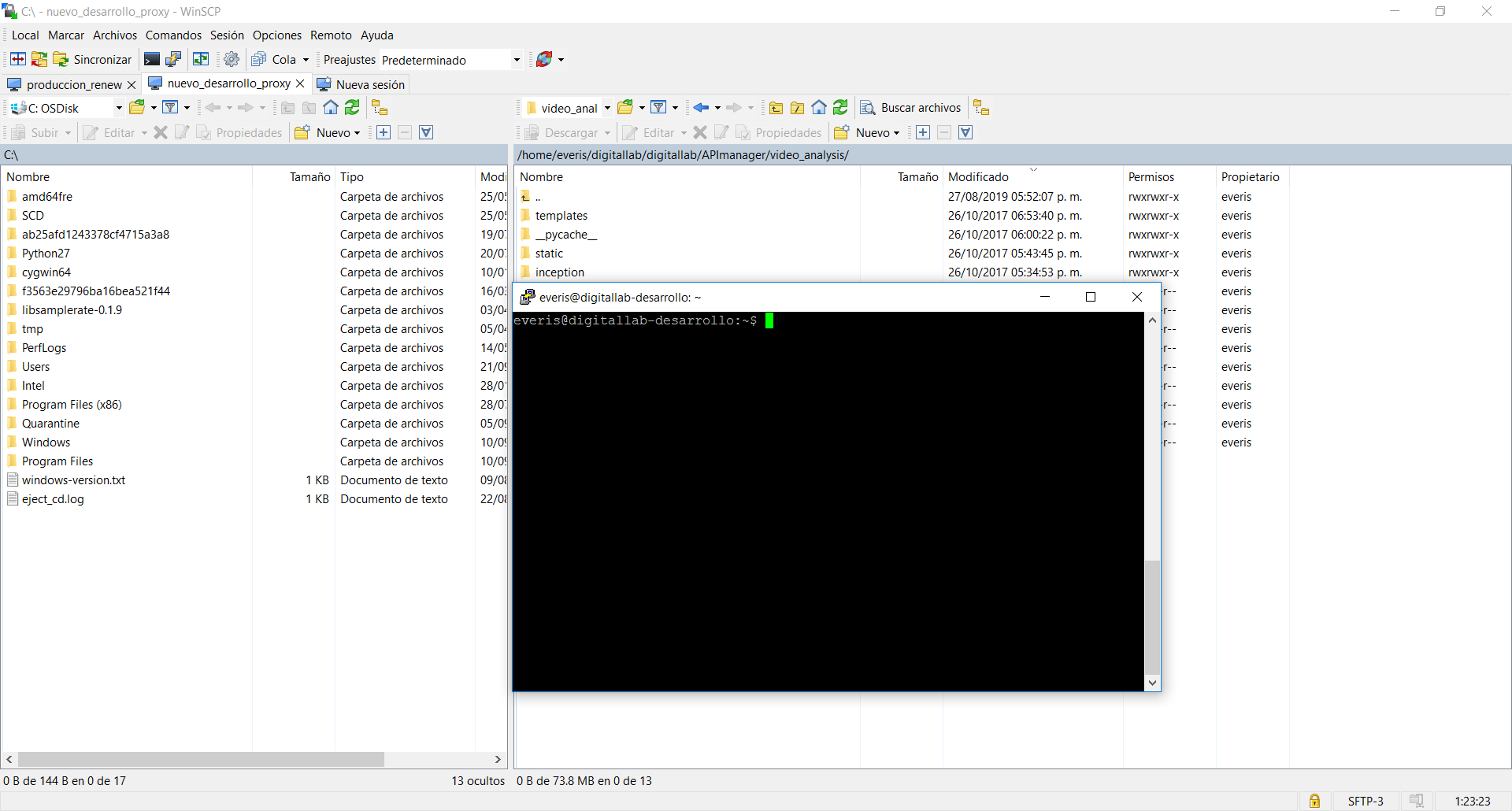
1. from flask import Flask, render\_template, jsonify, request, redirect, url\_for, send\_from\_directory
2. from flask\_socketio import SocketIO, send, emit
3. import time
4. import os
5. #################################################
6. import json
7. from time import time as t
8. import numpy as np
9. import math
11. app = Flask(\_\_name\_\_,static\_url\_path="/video\_index/static")
12. socketio = SocketIO(app)
14. @app.route('/video\_index/uploader', methods = ['GET', 'POST'])
15. def upload():
16. global filename
17. global datos
18. if request.method == 'GET':
19. filename = request.args.get('file')
20. if filename == 'CCTV.mp4':
21. datos = np.load('/home/everis/digitallab/digitallab/APImanager/simulador\_Video\_Analysis/cctv\_values.npy')
22. elif filename == 'Fast\_Food.mp4':
23. datos = np.load('/home/everis/digitallab/digitallab/APImanager/simulador\_Video\_Analysis/fast\_values.npy')
24. elif filename == 'Robo.mp4':
25. datos = np.load('/home/everis/digitallab/digitallab/APImanager/simulador\_Video\_Analysis/robo\_values.npy')
26. elif filename == 'Crimes.mp4':
27. datos = np.load('/home/everis/digitallab/digitallab/APImanager/simulador\_Video\_Analysis/crimes\_values.npy')
28. elif filename == 'Supermercado.mp4':
29. datos = np.load('/home/everis/digitallab/digitallab/APImanager/simulador\_Video\_Analysis/super\_values.npy')
30. elif filename == 'Asalto.mp4':
31. datos = np.load('/home/everis/digitallab/digitallab/APImanager/simulador\_Video\_Analysis/asalto\_values.npy')
32. print ("datos: ", datos.shape)
33. print ("nombre del video en get: ", filename)
34. return render\_template('./index.html',nameVideo=filename)
36. @socketio.on('my event')
37. def handle\_my\_custome\_event( json\_data ):
38. print('Received something: ' + str(json\_data))
39. n = json\_data.get('i')
40. s = json\_data.get('state')
41. print('state: ', s)
42. t\_state = math.floor(int(s))
43. print (" index ", t\_state)
44. clase = datos[t\_state][0]
45. accuracy = datos[t\_state][1]
46. time.sleep(0.4)
47. respond = {"results" : {"class\_name": clase, "score": accuracy, "limit": 1000, "status": 1} }
48. result = json.dumps(respond)
49. print ('Enviando.............')
50. socketio.emit('my response', result)

53. @app.route('/video\_index/')
54. def index():
55. return render\_template('./base.html', nameVideo=None)
57. if \_\_name\_\_ == '\_\_main\_\_':
58. socketio.run(app, debug = True, port=5007)

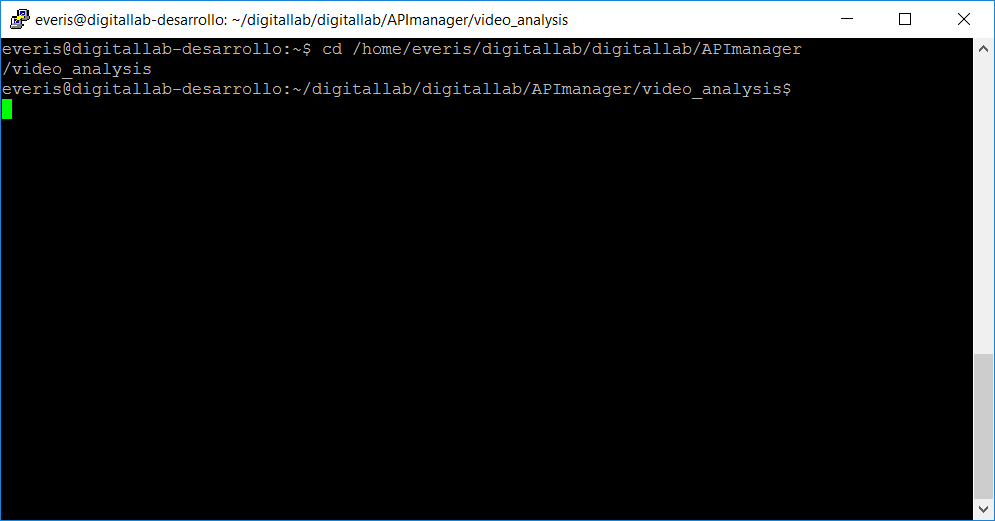


***Procedimiento de despliegue***

Abrimos sesión de putty

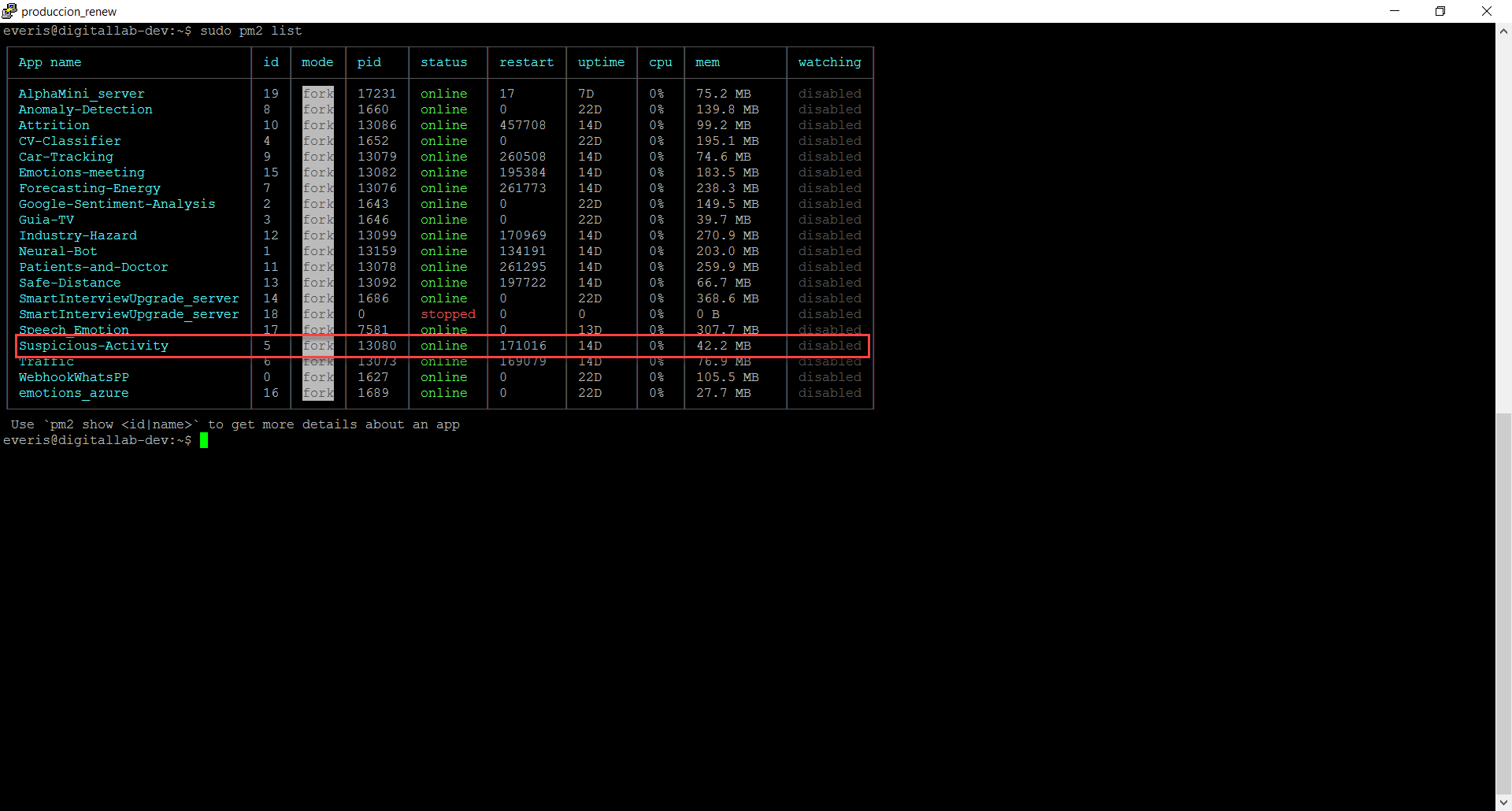


Nos ubicamos en la ruta */home/everis/digitallab/digitallab/APImanager/video\_analysis*



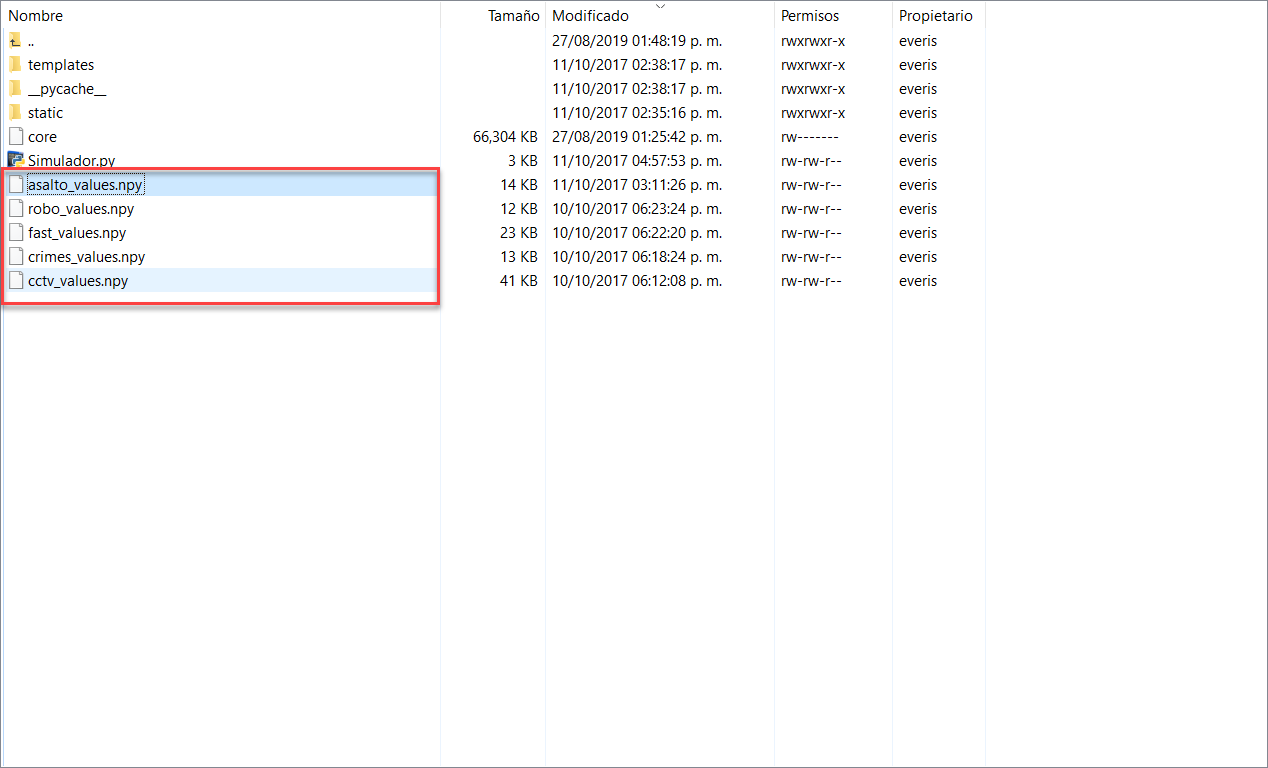
Ejecutamos el comando ***sudo pm2 start Simulador.py –interpreter python3 --name Suspicious-Activity***

Una vez hecho esto, veremos en consola la lista de los servicios levantados con pm2, ahí podremos ver el servicio que acabos de lavante y podremos ver su status actual. Debe aparecer como **online**



***Datos de entrada***

Los archivos de entrada que necesitamos para la aplicación deben encontrarse en la misma carpeta donde se encuentra Simulador.py. Se trata de unos archivos con terminación npy, los cuales contienes las respuestas de cada video analizado, que servirá para mostrar los resultados.



***Procedimiento de ejecución***

Se debe elegir una de las opciones disponibles



Una vez hecho esto, comenzaremos a ver los resultados desplegado correspondientes a la clasificación en cada segundo del video.

