



Universidad de Guadalajara

CUCEI – Centro Universitario de Ciencias Exactas e Ingenierías

Mtro. Michel Emanuel López Franco

Computación tolerante a fallas

Reporte 8



**Velasco Hernandez Victor
Manuel**

Código: 216598879

| D06 | 2022A |

| 28/Febrero/2022 |

Estatus





Universidad de Guadalajara
CUCEI
Computación tolerante a fallas

Victor Manuel Velasco Hernández
|Código: 216598879 | | 2022A |
|28/ Febrero /2022 | |D06|



Reporte 8- Estatus

Objetivo:

Realizar un programa que sea capaz de revisar el estatus de tu aplicación.

<https://python-para-impacientes.blogspot.com/2016/12/threading-programacion-con-hilos-i.html>

<https://tecnobillo.com/sections/python-en-windows/servicios-windows-python/servicios-windows-python.html>

Introducción:

El manejo de los estados de los procesos o servicios de un sistema de computo resultan ser fundamental al momento de querer manejar hilos, lo cuál hace posible que muchos procesos sean ejecutados en segundo plano de una manera correcta y con esto brindarles a los programadores excelentes herramientas para el estudio de estrategias para su implementación , lo cuál puede ayudar a mejorar la funcionalidad de los programas.

Desarrollo:

En esta actividad se va a crear un servicio que cierre un programa a partir de un script codificado en Python, para lo cuál se utilizara la herramienta de NSSM (Non-Sucking Service Manager) para poder ejecutar el script en segundo plano y poder manejar estados dentro del sistema operativo.

Para comenzar, se desarrolla el scrip, el cuál está diseñado para que reciba argumentos de programa NSSM, con el proposito de identificar el objetivo a cerrar, a continuación un fragmento del código, vale la pena aclarar que en el repositorio están los detalles:

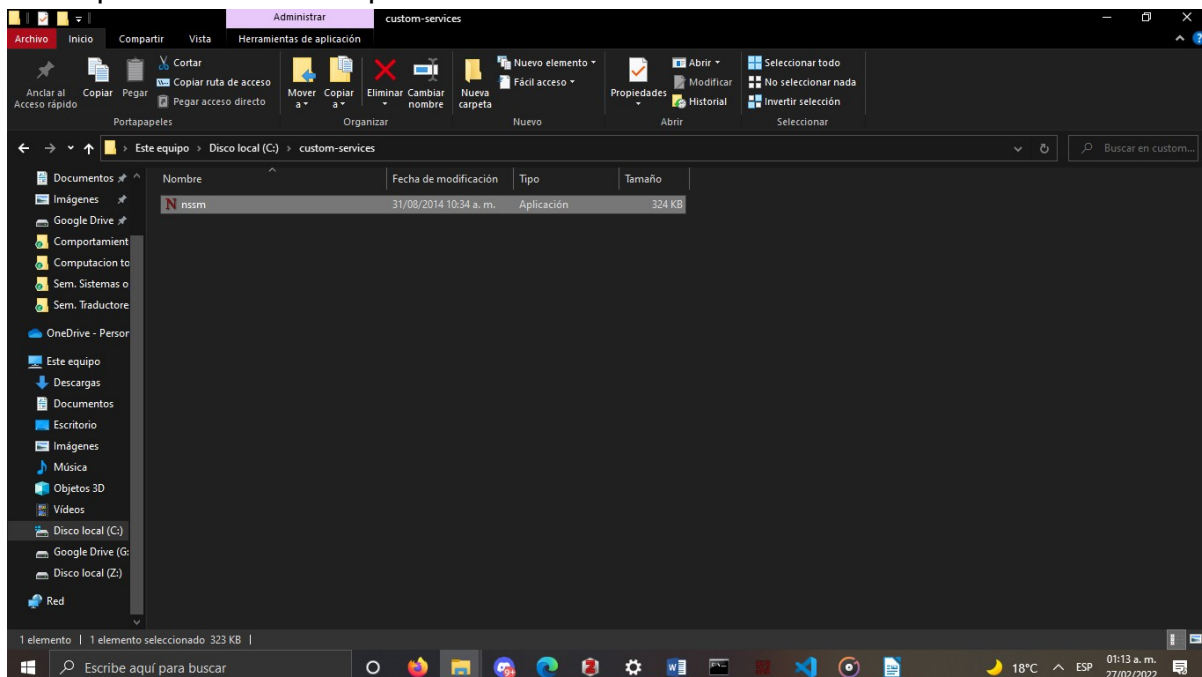


**Universidad de Guadalajara
CUCEI
Computación tolerante a fallas**

Victor Manuel Velasco Hernández
|Código: 216598879 | | 2022A |
|28/ Febrero /2022 | |D06|

```
File Edit Selection View Go Run Terminal Help script.py - ComputacionTolerante - Visual Studio Code
3-Estatus > script.py > ...
1 #Seminario de Solución de Problemas de Traductores de Lenguajes 2 [D02] |2022A|
2 #Victor Manuel Velasco Hernández
3
4 import sys
5 import psutil
6
7 def checar_argumento():#Revisa que se haya pasado algún argumento
8     if len(sys.argv) == 1:
9         print('Este programa no funciona sin argumentos')
10        sys.exit(0)
11
12 def fijar_objetivo():#Se pasa un argumento del servicio a cerrar
13
14     targets = sys.argv[1:]
15
16     i = 0
17     while i < len(targets):
18
19         if not targets[i].endswith('.exe'):
20             targets[i] = targets[i] + '.exe'
21
22         i += 1
23
24     return targets
25
26 def fijar_ataque(target):#En caso de que el proceso se haya encontrado se preparara para cerrarlo
```

Para poder ejecutar el NSSM, es recomendable poner el archivo .exe correspondiente en la carpeta “custom-services” en el disco C:

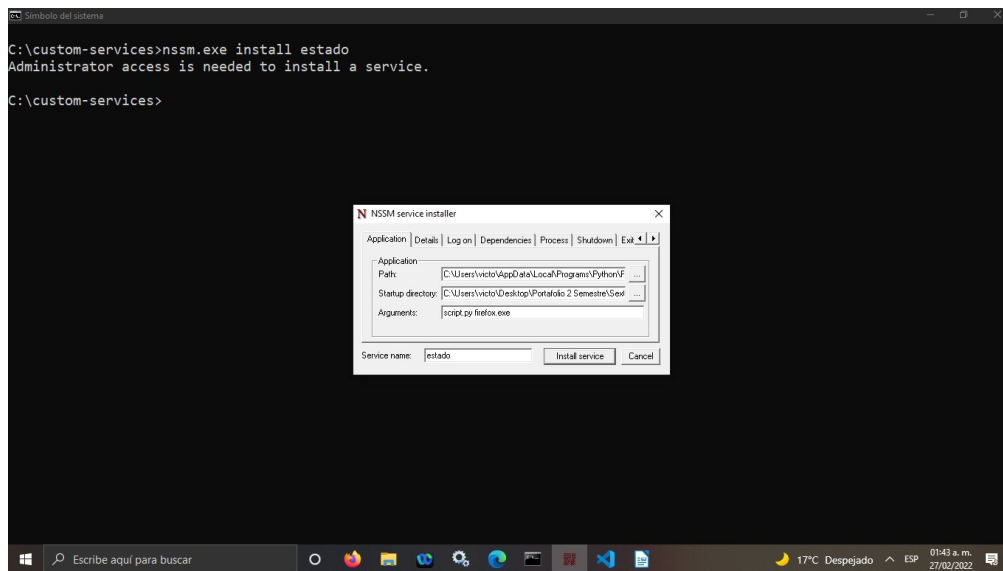




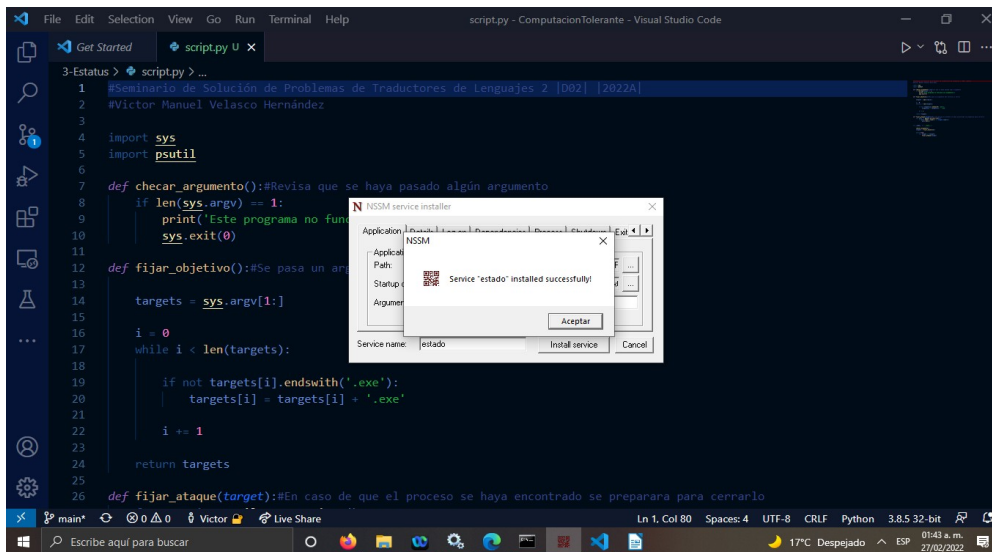
Universidad de Guadalajara
CUCEI
Computación tolerante a fallas

Victor Manuel Velasco Hernández
|Código: 216598879 | | 2022A |
|28/ Febrero /2022 | |D06|

Tras haber puesto el ejecutable en la carpeta, procedemos a ejecutarlo desde la consola de comando(no PowerShell), con la instrucción “nssm.exe install estado”, al introducir el comando se abrirá la siguiente ventana en la cuál en el apartado Path, pondremos el .exe correspondiente a Python y en Staartup directory pondremos la ruta del script a ejecutar, y después los argumentos correspondientes del proceso a cerrar.



Después procedemos a darle en “Install service” y se instalara exitosamente:





Universidad de Guadalajara
CUCEI
Computación tolerante a fallas

Victor Manuel Velasco Hernández
|Código: 216598879 | | 2022A |
|28/ Febrero /2022 | |D06|

Y por ultimo procedemos a iniciar el servicio con el comando “nssm.exe start estado”:

```
Administrador: Símbolo del sistema
C:\custom-services>nssm.exe start estado
estado: Unexpected status SERVICE_PAUSED in response to START control.
C:\custom-services>nssm.exe start estado
estado: START: Ya se está ejecutando una instancia de este servicio.
C:\custom-services>
```

Conclusiones:

El uso de procesos en segundo plano resulta ser muy útil para realizar procesos que queramos que se realicen inclusive cuando las aplicaciones se encuentren cerradas, además de que puede tener la utilidad de preservar información que se encontraba en memoria primaria, en caso de que se presenten fallas, lo cuál ayuda a hacer un sistema tolerante a fallos, por lo que resulta ser fundamental conocer la naturaleza de este tipo de funciones y estados para desarrollar cada vez, mejores funciones que garantizan el respaldo de la información, y la presencia de ejecución sin estar en el primer plano, lo cuál ayuda a tener una mejor comprensión de las eventualidades que se presentan en las computadoras, esto hace que podamos evitar posibles problemas en el futuro.