



# **UNIVERSIDAD TECNOLÓGICA DE PARRAL**

## **ACTIVIDAD PRACTICA 4**

**P R E S E N T A:**

**MARIAN AGUIRRE CARRILLO**

**CRISTO FER REYES MIJARES**

**MARIO DANIEL VILLADO CHAPARRO**

**GRUPO: IRIC 91V**

**DOCENTE:** Maria de Lourdes Garcia Carrasco

**MATERIA:** Automatización de infraestructura digital I

**FECHA:** 11/06/2025



# ÍNDICE

I	Sistema básico de monitoreo de red usando el ciclo de vida .....	2
1.1	Contexto .....	2
II	Paso a paso .....	2
2.1	Planeación y análisis de requerimientos .....	2
2.2	Diseño del sistema .....	3
2.3	Desarrollo o codificación .....	3
2.4	Pruebas .....	5
2.5	Implementación o Despliegue .....	5
2.6	Mantenimiento y actualización .....	6
	Ilustración 1 Diagrama de flujo .....	3
	Ilustración 2 Código del primer script .....	3
	Ilustración 3 Tarea en el programador de tareas de Windows .....	4
	Ilustración 4 Primeras dos líneas de la prueba manual .....	5
	Ilustración 5 Ultima línea de la ejecución por programa .....	5

# **I SISTEMA BÁSICO DE MONITOREO DE RED USANDO EL CICLO DE VIDA**

En esta actividad se desarrolló un software para automatizar el monitoreo de la conectividad de una red mediante un script en PowerShell.

## **1.1 Contexto**

Una empresa pequeña quiere asegurarse de que su conexión a internet esté activa todo el día. Para ello, desea un sistema que realice un ping automático cada 10 minutos a un servidor externo (por ejemplo, google.com) y registre los resultados para detectar posibles fallas.

# **II PASO A PASO**

## **2.1 Planeación y análisis de requerimientos**

El sistema hará un ping a la dirección 8.8.8.8 el cual se ejecutará de forma automática cada 10 minutos, esto a través de un script en PowerShell y el programador de tareas de Windows, los resultados se verán guardar en una carpeta de logs que muestre cuando las conexiones son exitosas y cuando no, esto junto a su fecha y hora de ejecución.

## 2.2 Diseño del sistema

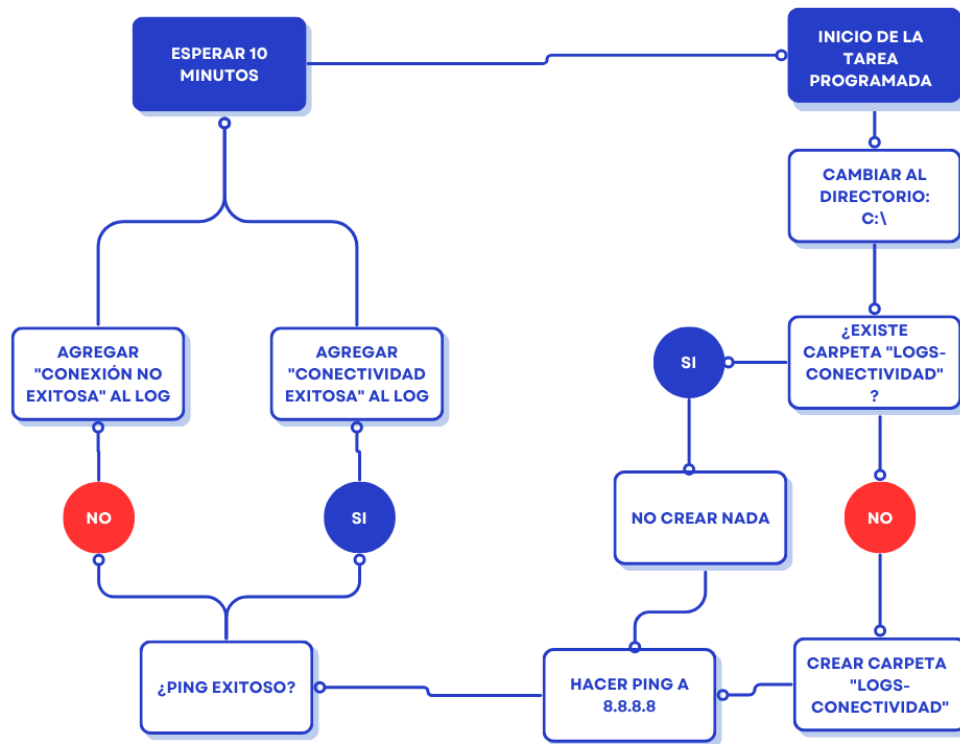


Ilustración 1 Diagrama de flujo

## 2.3 Desarrollo o codificación

Se crea el script en PowerShell que haga ping y registre el estado.

```
# Cambiar al directorio
Set-Location "C:\Scripts"

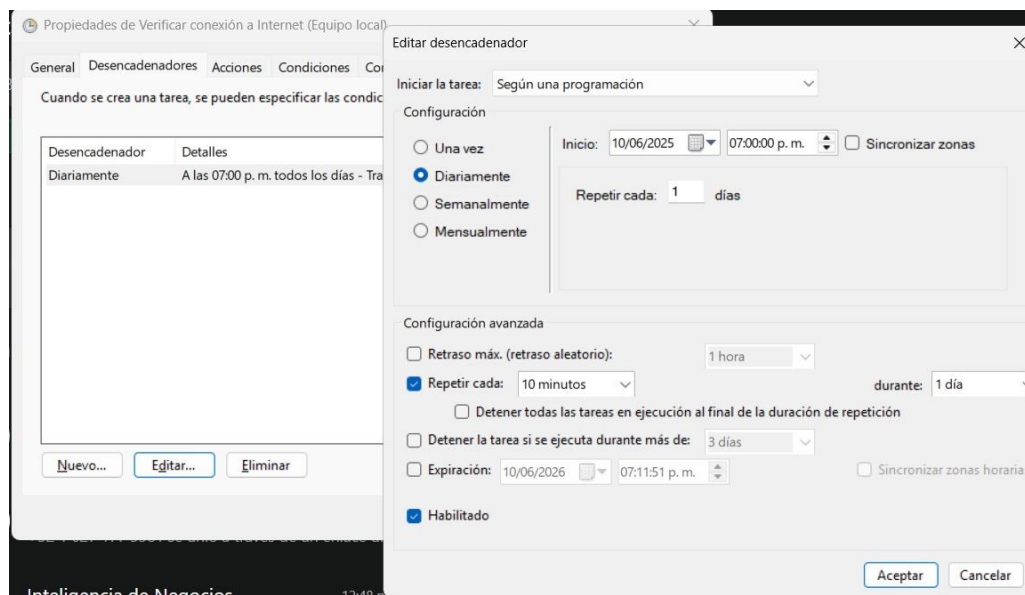
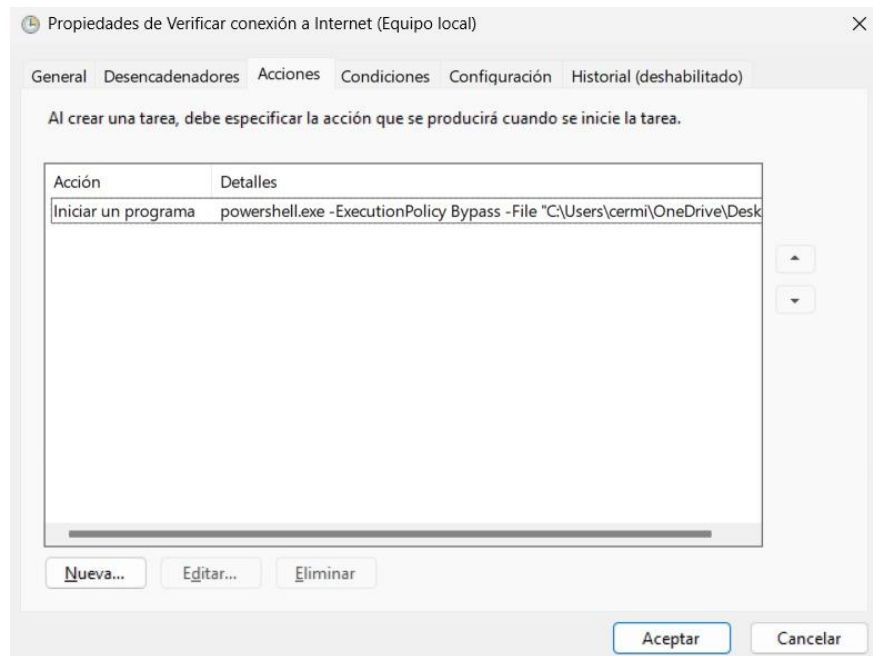
if (-not (Test-Path -Path ".\logs-conectividad" -PathType Container)) {
    New-Item -Path ".\logs-conectividad" -ItemType Directory
}

if (Test-Connection -ComputerName 8.8.8.8 -Count 1 -Quiet) {
    Add-Content -Path ".\logs-conectividad\logs.txt" -Value "Conectividad exitosa - $(Get-Date)"
} else {
    Add-Content -Path ".\logs-conectividad\logs.txt" -Value "Conexión no exitosa - $(Get-Date)"
}

Pause
```

Ilustración 2 Código del primer script

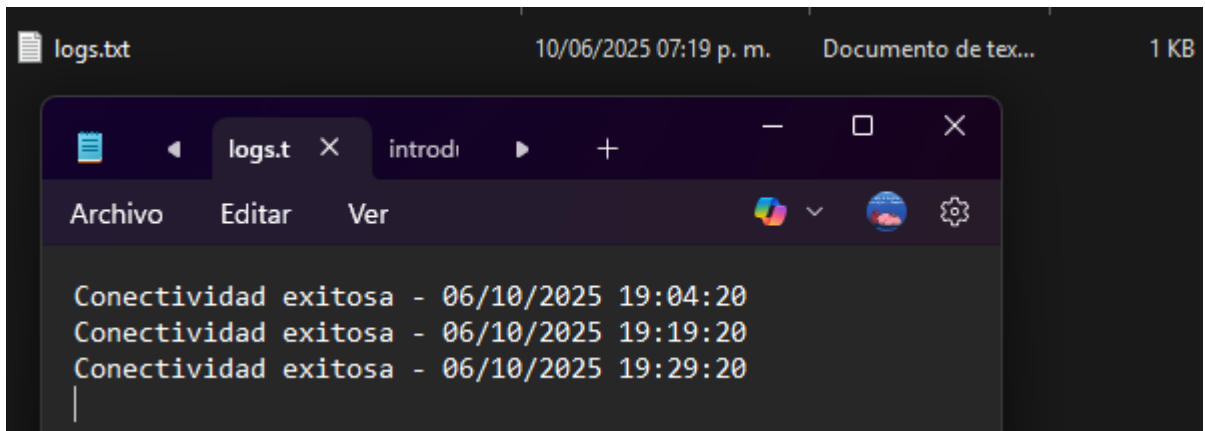
Mediante tareas programadas de Windows se genera la automatización de ejecución del script.



*Ilustración 3 Tarea en el programador de tareas de Windows*

## 2.4 Pruebas

Ejecución manual del script.

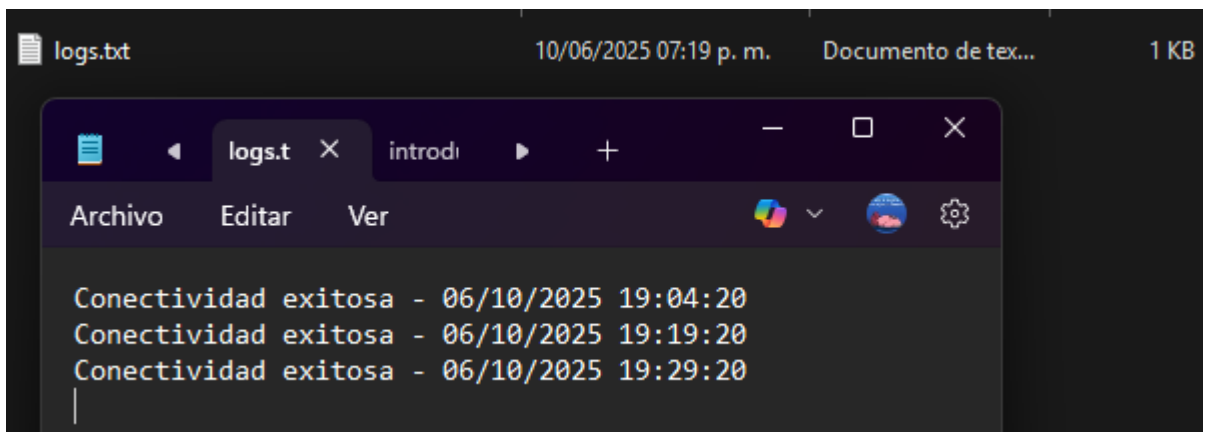


The screenshot shows a text editor window titled 'logs.txt' with a dark theme. The window has a menu bar with 'Archivo', 'Editar', and 'Ver'. Below the menu bar is a toolbar with icons for file operations. The main text area contains three lines of log output: 'Conectividad exitosa - 06/10/2025 19:04:20', 'Conectividad exitosa - 06/10/2025 19:19:20', and 'Conectividad exitosa - 06/10/2025 19:29:20'. The status bar at the bottom shows the date and time '10/06/2025 07:19 p. m.', the file name 'Documento de tex...', and the file size '1 KB'.

```
Conectividad exitosa - 06/10/2025 19:04:20
Conectividad exitosa - 06/10/2025 19:19:20
Conectividad exitosa - 06/10/2025 19:29:20
```

*Ilustración 4 Primeras dos líneas de la prueba manual*

Probar la tarea programada para confirmar que se ejecuta periódicamente.



This screenshot is identical to the one above, showing the same text editor window with the same three lines of log output: 'Conectividad exitosa - 06/10/2025 19:04:20', 'Conectividad exitosa - 06/10/2025 19:19:20', and 'Conectividad exitosa - 06/10/2025 19:29:20'.

```
Conectividad exitosa - 06/10/2025 19:04:20
Conectividad exitosa - 06/10/2025 19:19:20
Conectividad exitosa - 06/10/2025 19:29:20
```

*Ilustración 5 Ultima línea de la ejecución por programa*

## 2.5 Implementación o Despliegue

Ahora en la empresa nomas requieren descargar el script del GitHub donde este se encuentra, después colocar el script en alguna carpeta, acceder a las tareas programadas de Windows y agregar la nueva tarea que será el script, así mismo después de colocar la nueva tarea colocar que se ejecute cada 10 minutos.

(a la empresa le gusto, lo aprobó y ahora somos multimillonarios)

## **2.6 Mantenimiento y actualización**

Una de las mejores que tiene el script es que ahora el segundo script mejora la legibilidad, confiabilidad y experiencia de usuario, siendo más útil para monitorear la conectividad de forma automática y clara, mostrando de forma más visual la alerta de la falta de conexión de red.

Alguna otra mejora que se podría implementar sería la notificación con sonidos, agregar etiquetas como [INFO], [ERROR], [WARNING] para facilitar la lectura del log, crear un archivo de log por día para evitar que uno solo crezca demasiado, registrar cuanto tiempo estuvo sin conexión, hacer que los logs sean en CSV o JSON para analizar de mejor manera los datos entre otras cosas son las mejores que puede tener el script, en base al original y a la actualización.