

MediaSync-Daemon - Sistema de Reproducción Automática con VLC

DEPENDENCIA CRÍTICA: `pip install psutil`

Se ejecuta con `python MediaSync-Daemon.py`

Se ejecuta el registro de tarea automática con
`python create_task.py`

Descripción

MediaSync-Daemon es un sistema automatizado que gestiona la reproducción continua de videos mediante VLC, sincronizando contenido desde OneDrive y manteniendo la reproducción actualizada cuando se detectan cambios.

- Recomendación: Crear tarea en Task Manager para inicio ejecutar al inicio de sesión
- Opcional: Mantener un acceso directo en escritorio apuntando a `C:\Python312\python.exe C:\...\<dir>\MediaSync-Daemon\MediaSync-Daemon.py` para correr con simple doble clic dado el caso.

Requisitos del Sistema

- Python instalado en el sistema ej. `winget install --id Python.Python.3 --source winget`
- Windows 10 u 11 o superior
- OneDrive Configurado para mostrar directorio sharepoint en carpeta local del sistema
- Dependencia de librería psutil instalada mediante `pip install psutil` para manejar procesos del sistema.

- Acceso a powerShell de administrador solo para ejecutar el script que crea la tarea programada create_task.py o, en su defecto, crear la tarea manualmente.
- Acceso a lectura y escritura en carpeta %TEMP% de usuario.
- VLC instalado en directorio por defecto, o modificar ruta al ejecutable mediante el config.py
- Opcional: configurar preferencias para vlc: fullscreen, repeat, no titles, only one instance. Nota: se lanza vlc ya con parámetros óptimos y probados, pero configurarlo podría solucionar errores de haberlos.

Configuración de SharePoint y OneDrive

Hacer disponible un directorio de SharePoint en OneDrive

1. Abrir el navegador web y acceder al sitio de SharePoint que contiene la biblioteca de videos
2. Navegar hasta la biblioteca de documentos deseada
3. Hacer clic en "Sincronizar" en la barra superior
4. Se abrirá OneDrive automáticamente y comenzará la sincronización

Marcar como "Siempre disponible sin conexión"

1. Abrir OneDrive en el explorador de Windows
2. Localizar la carpeta sincronizada de SharePoint
3. Clic derecho en la carpeta
4. Seleccionar "Configurar siempre mantener en este dispositivo"
5. Esperar a que se complete la descarga inicial (círculo verde con palomita)

Nota: Es importante marcar como "Siempre disponible" el directorio entero para:

- Garantizar acceso inmediato a los videos
- Evitar problemas de streaming
- Permitir la detección confiable de cambios

Configuraciones cliente

- Recomendando configurar vlc para soportar una sola instancia
- Ajustar configuraciones en vlc para no mostrar ttítulosy transicionar suave entre videos
- Los videos se pueden cargar a un repo sharepoint, sin requisitos específicos más que ser archivos de video válidos. Nota: extensiones admitidas se configuran en config.py
- Se puede configurar la tarea programada manualmente, o mediante el script

```
$python create_task.py
```

Software

- Python 3.6 o superior
- librería python psutil 7.0.0 o superior
- VLC Media Player
- OneDrive configurado y sincronizado
- Windows OS (debido al uso de PowerShell para algunas funciones)

Permisos

- Acceso de escritura al directorio temporal del sistema
- Acceso de lectura al directorio de OneDrive
- Permisos de administrador para configurar tareas programadas mediante

```
create_task.py
```

Configuración

Estructura de Archivos

MediaSync-Daemon/

|

|— MediaSync-Daemon.py # Script principal PUNTO DE ENTRADA

|— config.py # Configuración centralizada

```
|— create_task.py      # Configurador de tarea programada
|— file_utils.py       # Utilidades de manejo de archivos
|— sync_utils.py       # Utilidades de sincronización OneDrive
|— vlc_utils.py        # Utilidades de control y validación de VLC
|— logging_utils.py    # Utilidades de logging con rotación
|— notes.txt           # Algoritmo fuente
|— diagrama-flujo.png  # Diagrama visual del flujo del sistema
|— stream_active.flag  # Flag de estado (creado durante ejecución)
```

Configuraciones Principales (config.py)

1. Configuración del directorio OneDrive/SharePoint de Videos fuente

```
VIDEO_CONFIG = {
    # Para carpeta personal de OneDrive:
    'VIDEO_DIR': r"C:\\Users\\<usuario>\\OneDrive\\...\\videos",
    # Para biblioteca de SharePoint:
    # 'VIDEO_DIR': r"C:\\Users\\<usuario>\\<Empresa>\\<Sitio> - <Biblioteca>\\videos",
    'VIDEO_EXTENSIONS': ['.mp4'],
    'MAX_FILE_SIZE_MB': 0 # 0 = sin límite
}
```

2. Configuración de VLC

```
VLC_CONFIG = {
    'VLC_EXE': r"C:\\Program Files\\VideoLAN\\VLC\\vlc.exe",
    'VLC_ARGS': [
        '--loop',          # Reproducir en bucle
        '--fullscreen',    # Pantalla completa
        '--no-video-title-show', # No mostrar título al inicio
        '--no-video-title',  # No mostrar título en ventana
        '--video-on-top'    # Mantener video siempre visible
    ],
```

```
'VLC_KILL_TIMEOUT': 3    # Timeout para cerrar VLC (segundos)
}
```

3. Configuración de Sincronización

```
SYNC_CONFIG = {
    'DOWNLOAD_FINISH_DELAY': 2,    # Espera post-estimulación OneDrive
    'REFRESH_CYCLE_DELAY': 10,     # Intervalo de ciclo principal
    'MAX_SYNC_RETRIES': 3,         # Reintentos de sincronización
    'MAX_CONSECUTIVE_ERRORS': 3,   # Límite de errores consecutivos
    'ERROR_RETRY_DELAY': 5,        # Espera entre reintentos
    'MIN_FILE_SIZE_FOR_TAIL_CHECK': 16384, # Tamaño mín. para verificación completa
    'FILE_CHECK_BLOCK_SIZE': 8192  # Tamaño de bloque para verificación
}
```

4. Configuración de Logging

```
LOG_CONFIG = {
    'LOG_PATH': os.path.join(VIDEO_CONFIG['VIDEO_DIR'], "stream_status.log"),
    'LOG_LEVEL': 'INFO',
    'MAX_LOG_SIZE_MB': 10,
    'LOG_BACKUP_COUNT': 3,
    'LOG_ENCODING': 'utf-8'
}
```

Flujo de Funcionamiento

1. Inicialización

1. Validación exhaustiva de configuración
2. Configuración del sistema de logging con rotación
3. Verificación de archivos de video disponibles

4. Verificación de archivos de video
5. Detención de instancias previas de VLC

2. Ciclo Principal de Monitoreo

1. Validación continua de contenido:

- Verificación de archivos válidos en directorio fuente
- Control de errores consecutivos con límite configurable
- Sistema de reintentos automáticos

2. Detección de cambios:

- Cálculo de hash del contenido del directorio
- Comparación con hash anterior almacenado
- Activación de actualización solo cuando hay cambios reales

3. Actualización de contenido (cuando se detectan cambios):

- Detención segura de VLC con verificación de cierre
- Limpieza de directorio temporal
- Copia de nuevos archivos al directorio temporal
- Regeneración de playlist con validación de archivos
- Actualización de hash de estado

4. Gestión de VLC:

- Verificación de FLAG de estado antes de iniciar
- Validación de playlist con contenido válido
- Inicio de VLC con validaciones múltiples:
 - Verificación de ejecución exitosa usando psutil
 - Confirmación de proceso activo
 - Creación de FLAG solo si todo es exitoso

5. Estimulación de OneDrive:

- "Toque" de archivos para forzar sincronización
- Verificación de integridad de archivos
- Comando PowerShell para estimular sincronización
- Reportes detallados de estado de sincronización

3. Características Avanzadas

Verificación de Integridad de Archivos

- Lectura de bloques iniciales y finales
- Detección de transferencias incompletas
- Verificación de acceso y permisos

Sistema de FLAG de Estado

- `stream_active.flag` : Indica VLC ejecutándose & reproduciendo contenido válido
- Creación solo después de validaciones exitosas
- Eliminación automática al detener VLC
- Detección de procesos VLC huérfanos

Gestión Robusta de Procesos VLC

- Detección de procesos usando psutil (instalar dependencia)
- Timeout configurable para cierre de procesos
- Limpieza automática de procesos fallidos
- Validación de ejecución antes de confirmar inicio

4. Manejo de Errores y Recuperación

- Sistema de reintentos configurables
- Logging detallado con niveles apropiados
- Límite de errores consecutivos
- Recuperación automática de estados inconsistentes

- Detección y limpieza de FLAG huérfano

Características de Seguridad y Robustez

- **Verificación de integridad de archivos** con lectura de bloques inicial/final
- **Validación exhaustiva de procesos VLC** usando psutil
- **Sistema de FLAG de estado** para prevenir ejecuciones múltiples
- **Manejo de errores con recuperación automática**
- **Sistema de logging con rotación automática**
- **Validación de configuración completa al inicio**
- **Detección y limpieza de procesos huérfanos**
- **Timeout configurable para operaciones críticas**

Configuración de Inicio Automático

Usando create_task.py

1. Ejecutar con permisos de administrador:

```
python create_task.py
```

1. Crea una tarea programada que:
 - Se ejecuta al inicio de sesión
 - Usa el intérprete de Python correcto
 - Mantiene los permisos necesarios

Monitoreo y Mantenimiento

Logs

- **Ubicación:** Mismo directorio que los videos OneDrive (Para monitorearlo remotamente desde el repo Sharepoint)
- **Formato:** `MediaSync.log`

- Rotación: Configurada para mantener histórico manejable

Archivo de Estado y Temporales

- `stream_active.flag` : Indica estado activo de VLC (creado solo después de validaciones)
- Archivos temporales en `%TEMP%` :
 - `daemon_temp_media/` : Copia de videos para reproducción
 - `playlistVLC.m3u` : Playlist generada automáticamente
 - `daemon_media.hash` : Hash de estado para detección de cambios

Estados del Sistema

- **FLAG existe + VLC activo:** Funcionamiento normal
- **FLAG existe + VLC inactivo:** Posible crash de VLC (se limpia automáticamente)
- **FLAG no existe + VLC activo:** Estado inconsistente (se corrige automáticamente)
- **FLAG no existe + VLC inactivo:** Estado de espera normal

Solución de Problemas

Problemas Comunes

1. VLC no inicia:

- Verificar ruta del ejecutable y actualizarla en `config.py`. Atención al formato de ruta para Windows. Recomendado, usar rutas absolutas.
- Restablecer preferencias de VLC, configuraciones exóticas podrían estar interfiriendo en el comportamiento deseado.
- Comprobar playlist en `%TEMP%\playlistVLC.m3u`
- Verificar permisos de ejecución, lectura y escritura para ficheros de VLC
- Verificar logs para detalles de validación

2. Sincronización OneDrive lenta:

- Verificar configuración de tiempos de espera en config.py
- Revisar estimulación en logs
- Verificar "Siempre disponible sin conexión"
- Comprobar velocidad de conexión

3. FLAG huérfano persistente:

- El sistema detecta y limpia automáticamente
- Verificar logs para procesos VLC huérfanos

4. Errores de integridad de archivos:

- Verificar sincronización completa de OneDrive
- Verificar almacenamiento de OneDrive
- Comprobar espacio disponible en disco
- Revisar permisos de directorio temporal

Limitaciones Conocidas

- Específico para Windows debido a uso de PowerShell
- Requiere OneDrive configurado
- Necesita VLC instalado en ruta predeterminada (o editar la ruta en [config.py](#))

Para más detalles técnicos, consultar los comentarios en el código fuente