Automatizar Sistema Legacy

Descargar Manual: **Controlar_Sistema_Legacy**

Descargar Manual y Base de Datos con ejemplo: **Legacy_Bot_Example**

Sistema Legacy

Los sistemas Legacy son aplicaciones obsoletas o desactualizadas que siguen dentro de una organización y que no pueden reemplazarse fácilmente, en este caso nos enfocaremos en aplicaciones de tipo Terminal.

Simularemos un sistema Legacy utilizando la cmd.

Estructura

Robot Principal:

- main_legacy
  - **Variables:**
    - **word:** Palabra a buscar en pantalla cmd.
    - **validate:** Variable que recibirá un True para indicar que la palabra asignada a ***{word}*** fue encontrada en la pantalla.
    - **path:** Ruta hacia la carpeta de usuario.

Robot Hijos:

- validateScreen_legacy
  - **Variables:**
    - text: Variable que guardará el texto copiado del cmd.
- screen2_legacy
  - **Variables:**
    - Ninguna

---

1. Iniciar Aplicación: Para abrir la aplicación lo podemos realizar mediante el comando **Ejecutar aplicación** o crear un **.bat** y llamarlo desde ese mismo comando.

   En este caso se trabajará con la cmd, por lo que la abriremos mediante el comando **Asignar Variable,** necesitamos abrir la cmd en la carpeta del usuario, por lo que realizaremos lo siguiente:

   - Obtendremos la ruta del usuario del PC controlado reemplazando los backslash (\) por slash (/) y lo guardamos en la variable ***{path}***:



   - Iniciamos la cmd en la ruta especificada en la variable ***{path}***:

2. Envío de Instrucción: Enviaremos las instrucciones mediante el comando "**Enviar Tecla**", en este caso necesitamos realizar el envío del texto "**dir**" y luego un **ENTER**.

3. Asignar palabra a buscar: Asignaremos la palabra que necesitamos buscar en la cmd a una variable, en este caso buscaremos el texto "Desktop" y lo guardaremos en ***{word}***.

> **Importante:** No es óptimo utilizar Virtualización en este tipo de Sistemas, puedes usar el envío de teclas y el portapapeles para buscar y/o esperar información en el terminal.

4. Ejecutar bot hijo: Ejecutamos el bot hijo encargado de buscar la palabra asignada a la variable ***{word}*** hasta que la encuentre (dentro de un While.)

- **validateScreen_legacy**
  - **Limpiar Variable: Limpiamos la variable *{validate}* del robot padre antes de iniciar el flujo, con el comando "Limpiar variable(s)" del módulo System++.**

  - **Enviar combinaciones: Enviaremos las combinaciones de tecla ctrl+a y ctrl+c mediante el comando "Enviar Combinaciones" del módulo "Teclado".**

> Si se está trabajando con Putty, no permitirá el envío de teclas ctrl+a o ctrl+c, pero se pueden realizar los siguientes pasos para poder copiar toda la información del terminal:

1.- Ir a Window -> Behaviour.
Luego marcar la casilla "***System Menu Appears on ALT-SPACE***" y "***Ensure Windows is always on top***".

2.- Ir a Session -> Escribir un nombre para guardar la sesión en "**Saved Sessions**"

> Cuando esté conectado a la terminal, puede realizar la siguiente combinación de tecla **ALT+SPACE** (en el comando **Enviar Tecla** del menú **Desktop** debes escribir **^{SPACE}**), esperar 1 segundo, y luego enviar la letra **O**, con eso se guardará en el portapapeles.

- **Guardar en Portapapeles: Lo que acabamos de copiar con el comando anterior (ctrl+a y ctrl+c) o ^{SPACE} y luego la letra O si es putty , lo asignaremos a una variable *{text}* mediante el comando "Obtener portapapeles" del módulo "Clipboard"**

- **Buscar hasta: Repetiremos las instrucciones anteriores (ctrl+a , ctrl+c y obtener portapapeles) hasta que la palabra asignada a *{word}* se encuentre en el texto obtenido en *{text}*.**
  Dejaremos la repetición dentro de un **While**, esto simula una espera, buscará hasta que la pantalla que queremos cargue y logre encontrar un texto que nos indique que estamos en la vista correcta, una vez la encuentre, saldrá del While y continuará el flujo.

  - Podemos también agregar un contador en el while para que después de X intentos se salga y no continúe en un ciclo infinito

- **Asignación de variables: Fuera del While limpiaremos la variable *{word}* del padre y luego asignaremos como valor un True a la variable *{validate}* del padre, para indicar que encontró la palabra deseada y nos encontramos en la pantalla correcta.**

5. Ejecutar bot hijo: Ejecutamos el bot hijo encargado de realizar las acciones en la pantalla correspondiente.

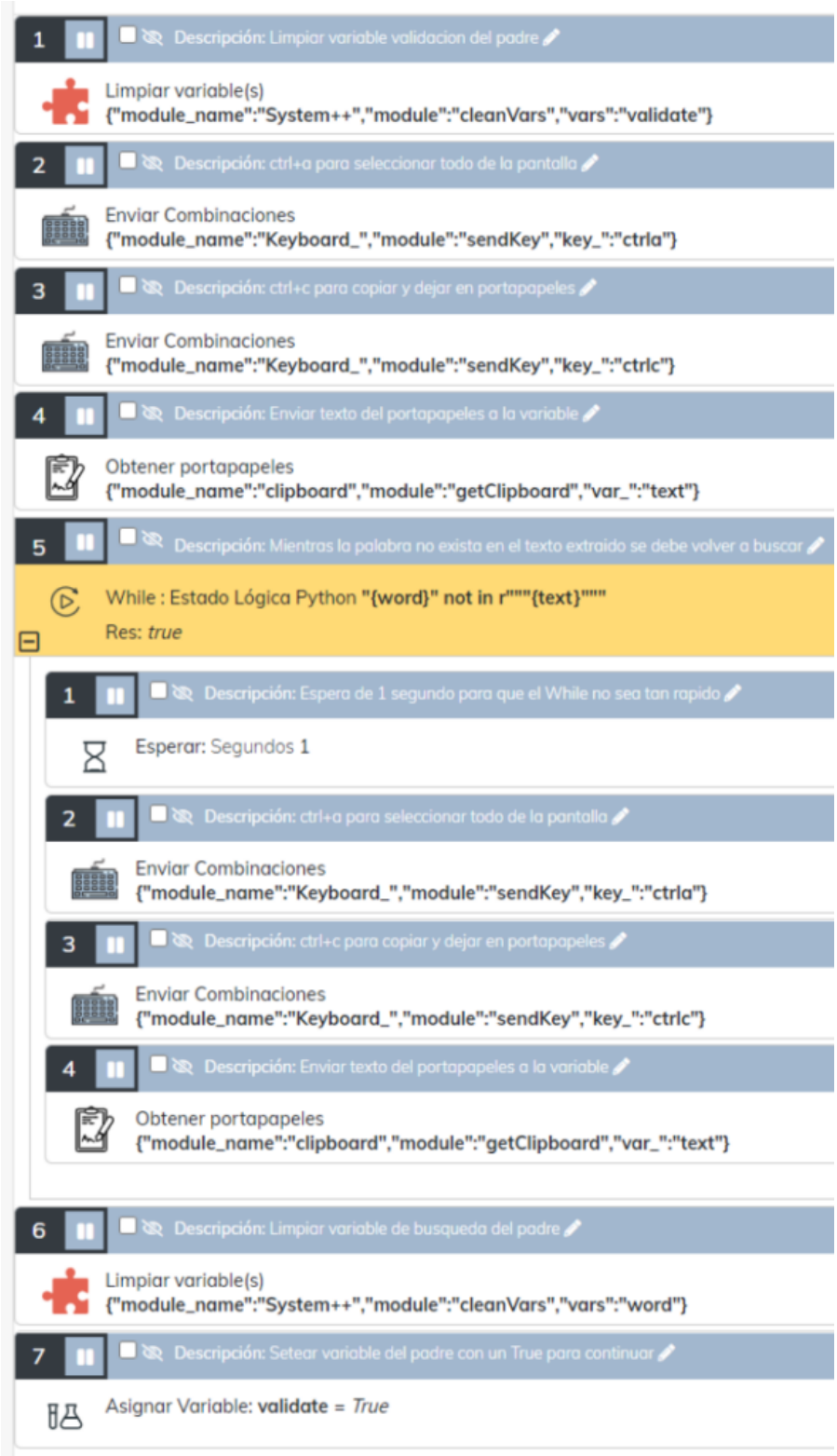
- **screen2_legacy**
  - **Validar ejecución anterior: Validaremos mediante un IF si la variable *{validate}* del robot padre es True, eso significa que sí nos encontramos en la pantalla correcta, por lo que ejecutaremos las instrucciones necesarias, en este caso "cd Desktop" y un ENTER para acceder a la carpeta.**

6. Cerrar todo: Al final de las instrucciones del robot principal, debemos cerrar la aplicación, en este caso la cmd, utilizaremos el comando "**Mata una Aplicación**" del módulo "**KillApp**"

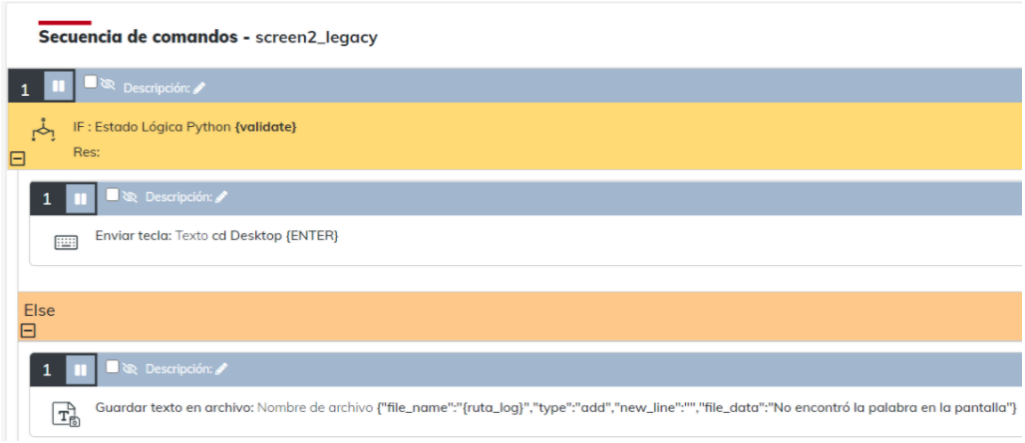
Secuencias:

main_legacy:

validateScreen_legacy:



screen2_legacy:


Secuencia de comandos - screen2_legacy
1
Descripción:
IF : Estado Lógica Python {validate}
Res:
1
Descripción:
Enviar tecla: Texto cd Desktop {ENTER}
Else
1
Descripción:
Guardar texto en archivo: Nombre de archivo {"file_name":"{ruta_log}","type":"add","new_line":"","file_data":"No encontró la palabra en la pantalla"}