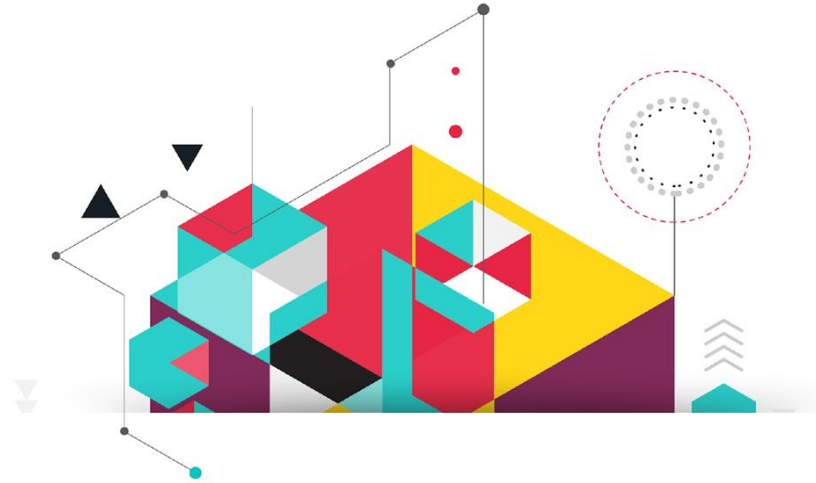


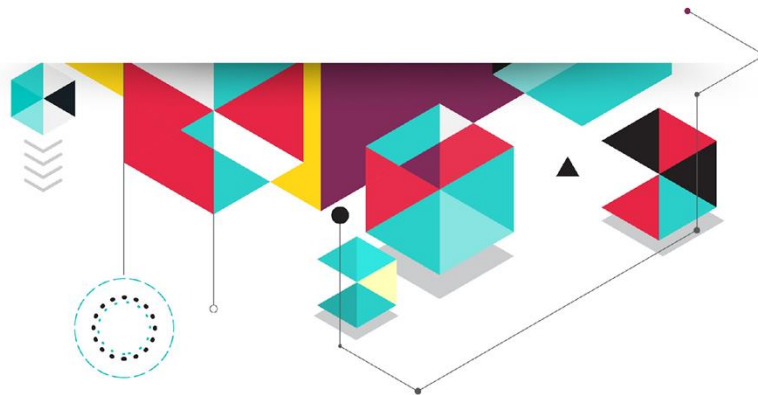


LECCIÓN 12: DEPURACIÓN Y MANEJO DE EXCEPCIONES, RESUMEN

Información general








En este módulo hemos profundizado en 2 de los aspectos más importantes del desarrollo de software en general y las herramientas que ofrece UiPath Studio al respecto. Hemos presentado la **pestaña Debug** y la **actividad Try Catch**, junto con otras técnicas muy útiles.






Enseñanzas clave

- La depuración se puede empezar pulsando el **botón Start Debug** en la **pestaña Execute**.
- Al ejecutar la depuración, se accionan 3 cosas:
 - Hay una marca amarilla en la acción que se está ejecutando y, si procede, una marca opcional roja en el elemento afectado por esta acción
 - El **panel Locals**, donde se puede comprobar el valor actual de todas las variables
 - Un registro detallado de todas las acciones que tienen lugar en el flujo de trabajo
- Si prefiere menos velocidad, utilice el **botón Slow Step** o pause completamente la ejecución utilizando el **botón Toggle Breakpoint** y continúe la ejecución paso a paso utilizando el **botón Step Over**.
- Al esperar que se cargue una aplicación, el tiempo de espera predeterminado de una actividad es 30 segundos, pero también puede utilizar actividades como **Element Exists**, **Find Element** o **Wait Element Vanish**, y sus equivalentes para imagen.

Enseñanzas clave

-  **Element Exists** no afecta al flujo de trabajo, solo devuelve un valor booleano, mientras las otras 2 actividades detienen la ejecución hasta que el elemento se localiza o desaparece.
-  La actividad **Try Catch** debe contener las acciones que pueden generar un error dentro del bloque **Try**, las acciones por realizar cuando se produzca un error dentro del bloque **Catch** y, opcionalmente, las acciones por ejecutar siempre detrás de los otros 2 bloques dentro del bloque **Finally**.
-  Es posible tener varios **Catches** para tipos distintos de excepciones.
-  Aunque se haya capturado una excepción, es posible que necesite garantizar que el flujo de trabajo se detenga realmente, en este caso puede utilizar la actividad **Rethrow**.
-  Es posible separar componentes individuales de la automatización en distintos archivos de flujo de trabajo y después hacer una llamada utilizando **Invoke Workflow**.

Procedimientos recomendados

-  Es muy importante utilizar nombres descriptivos en acciones y diagramas de flujo, es beneficioso a largo plazo.
-  Una buena estrategia, si desea evitar problemas generados por ventanas que puedan estar encima de la que desee utilizar, es mantenerse alejado del método de entrada predeterminado.
-  Si tiene problemas con los selectores, utilice las opciones **Indicate On Screen** y **Attach to live element** para “actualizarlos”.

Enlaces útiles



[Enabling Tracing \(Activación del seguimiento\)](#)
[How to use Try Catch Activity? \(Cómo se utiliza una actividad Try Catch\)](#)

