
Universidad
Tecmilenio®

Universidad Tecmilenio

Software Testing

Ejecución de pruebas con archivo de datos

Semana 10

Introducción

Te solicitan crear un API que permita mostrar al usuario, el clima y la hora de la ciudad donde se encuentra.

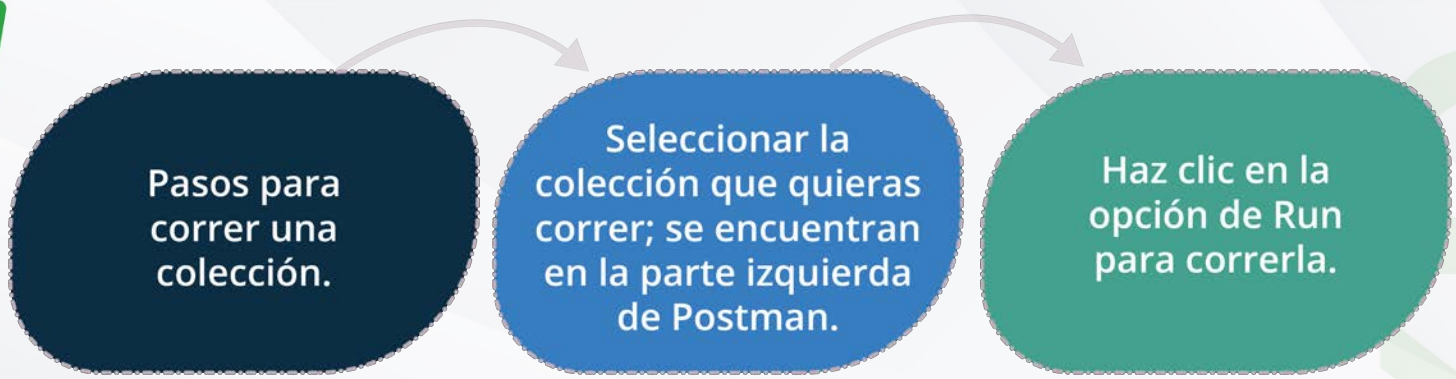
Dado este requerimiento, utilizarás Postman para simplificar el desarrollo de la API; decides crear una colección donde haces dos requerimientos; uno para establecer la conexión con la API de Weather Channel y el otro para establecer la conexión con la API que permita obtener la hora actual.

Al realizar esto, se creará un archivo JSON que contiene una variable que enlista todas las ciudades del mundo y que al correr la colección, los requerimientos realicen las conexiones pertinentes y regresen la información que se solicita.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Generar archivos CSV y JSON



Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Generar archivos CSV y JSON

Para utilizar un archivo en formato CSV debe de ser formateado; la primera fila debe tener el nombre de las variables a utilizar.

- Las siguientes filas serán utilizadas para los datos.
- Las terminaciones de las líneas deberán estar en formato UNIX.
- Todas las filas deben tener el mismo número de columnas, ya que, de no ser así, se generaría un error.

City	Soccer
Vancouver	38
San Francisc	67
Singapore	70
Yonkers	10
Austin	98
Los Angeles	42
San Diego	71
Makati	75
Seattle	63
San Jose	84
New York	58
Toronto	77
Quezon City	23
Chicago	62
Boston	11
Calgary	10
Washington	51
Houston	24
Montreal	18
Melbourne	62
Perth	68
Barcelona	56
Philadelphia	15
Brisbane	34
Dallas	34
Sydney	78
Jakarta	64
Milan	89
Madrid	5
Berlin	28
London	83
Mexico City	10
Bangkok	41
Warsaw	41
Paris	51

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Generar archivos CSV y JSON

En cuanto a los archivos JSON, se deben formatear, de tal forma que sean un arreglo que tenga pares de llave. Las llaves tendrán el nombre de la variable y dentro estarán los datos que serán usados en el requerimiento.

```
[
  {
    "City": "Vancouver",
    "Soccer": 38
  },
  {
    "City": "San Francisco",
    "Soccer": 67
  },
  {
    "City": "Singapore",
    "Soccer": 70
  },
  {
    "City": "Yonkers",
    "Soccer": 10
  },
  {
    "City": "Austin",
    "Soccer": 98
  },
  {
    "City": "Los Angeles",
    "Soccer": 42
  },
  {
    "City": "San Diego",
    "Soccer": 71
  },
  {
    "City": "Makati",
    "Soccer": 75
  },
  {
    "City": "Seattle",
    "Soccer": 63
  },
  {
    "City": "San Jose",
    "Soccer": 84
  },
  {
    "City": "New York",
    "Soccer": 58
  },
  {
    "City": "Toronto",
    "Soccer": 77
  },
```

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Explicación

Orquestar prueba con más de un request

Las pruebas te permitirán verificar que tu API esté funcionando bien, al igual que las integraciones con todos los servicios que se conecta tu API.

Dentro de Postman puedes hacer pruebas a un solo requerimiento o una colección completa.

1. Haz clic en colecciones y luego en +
2. Crea el primer requerimiento para tu colección.
3. Genera todos los requerimientos que necesites en la colección.
4. Crea el CSV o el JSON que contendrá las variables de datos que se enviarán en cada requerimiento.
5. Da clic en los tres puntos que se encuentran al lado del nombre de tu colección que quieres probar y haz clic en Correr.
6. Selecciona los requerimientos a probar, llena los datos solicitados y sube el archivo CSV o JSON.
7. Revisar los resultados obtenidos.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Actividad

¿Consideras que los requerimientos pueden ayudarte en tus desarrollos?

¿Crees que Postman es una herramienta útil para tus desarrollos?

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Cierre

Las colecciones en Postman te permitirán crear diferentes requerimientos de tal forma que puedas hacer pruebas al mismo tiempo en todos ellos. Dichas pruebas necesitarán la creación de un archivo JSON o CSV, en el cual establecerás las variables de datos que utilizarán los requerimientos para verificar que tu API funciona correctamente.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

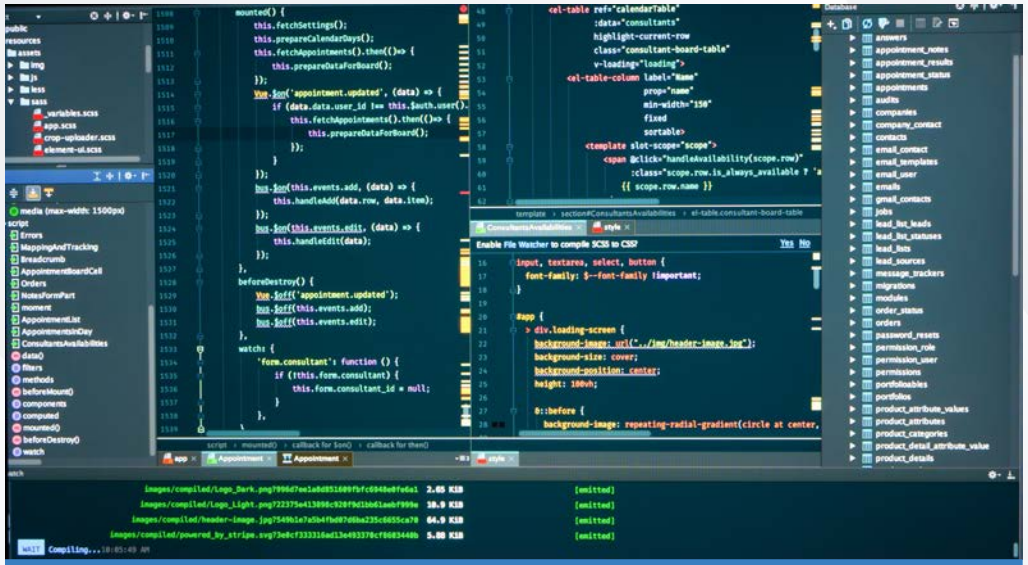
Software Testing

Ejecución de pruebas a través de la consola

Semana 10

Universidad Tecmilenio

Introducción



Has terminado el desarrollo de tu API, ahora hay que revisar que todo se encuentre funcionando correctamente, hay que exportar las colecciones para realizar las pruebas.

Realiza la instalación de Node.js y Newman, estos te permitirán hacer las pruebas desde la consola del servidor donde colocaste la API. Se van a colocar los archivos JSON de las colecciones y comenzarás a ejecutar verificando que las pruebas sean exitosas.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Explicación

Preparación de colección para ejecutar en consola

Primero deberás exportarla como un archivo; para realizar dicha exportación deberás seguir los siguientes pasos:

1. Haz clic derecho en la colección a exportar y selecciona dicha opción.
2. Selecciona el formato al que se exportará la colección.
3. Documentación de las API.
4. Selecciona la opción recomendada y haz clic en exportar; se descargará un archivo JSON en tu equipo.

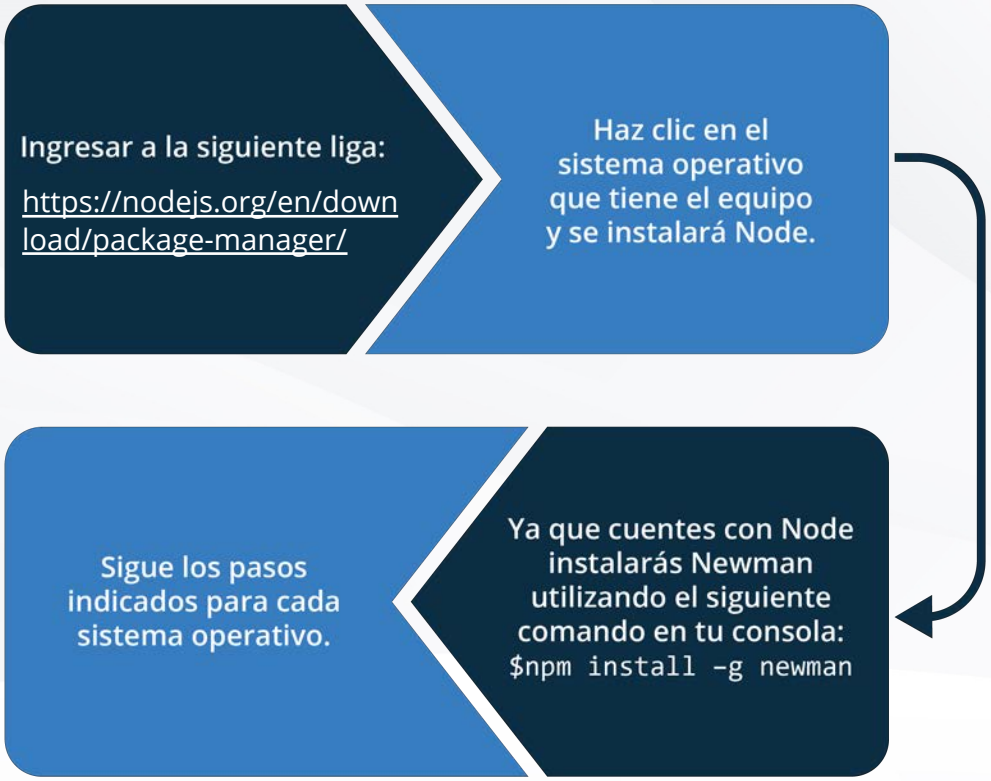
Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Explicación

Instalación, configuración y uso de Newman

Newman ha sido creado en Node.js. Antes de ejecutar Newman, debes asegurarte de tener Node.js instalado; para instalar Node debes seguir los siguientes pasos:



Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Explicación

Instalación, configuración y uso de Newman de una colección

Opción básica	Detalles
-h, --help	Para obtener información de uso.
-v, --version	Para obtener la versión de Newman.

Opción de requerimiento	Detalles
--delay-request [number]	Definir un retraso entre requerimientos.
--timeout [number]	Definir el tiempo que se esperará para que se complete la ejecución de la colección.
--timeout-request [number]	Definir el tiempo muerto para un requerimiento.
--timeout-script [number]	Definir el tiempo que se esperará para que se complete la ejecución de los scripts.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Explicación

Instalación, configuración y uso de Newman de una colección

Opción miscelánea	Detalles
--bail	En caso de que falle la prueba, se detiene Newman.
--silent	Apagar la terminal de salida.
--color off	Apagar la salida de colores (auto/encendido/apagado).
--disable-unicode	Forzar la opción de deshabilitar unicode.
-k, --insecure	Apagar los SSL en modo “estricto”.
-x, --suppress-exit-code	Continuar con la prueba, aunque falle, regresar el código =0.
--ignore-redirects	Apagar las respuestas automáticas 3XX.
--verbose	Mostrar información detallada cuando se ejecute la colección o cuando se envíe un requerimiento.
--cookie-jar [path]	Especificar la ruta para una cookie en JSON.
--export-cookie-jar [path]	Especificar la ruta para guardar el resultado del archivo de “cookies” antes de que se complete la ejecución.
--global-var "[global-variable-name]=[global-variable-value]"	Especificar las variables globales.
--env-var "[environment-variable-name]=[environment-variable-value]"	Permite colocar las variables de ambiente en un formato de nombre=valor.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Instalación, configuración y uso de Newman de una colección

Explicación

Opción de configuración	Detalles
--folder [folderName]	Para especificar la carpeta desde la cual se ejecutará la colección.
-e, --environment [file|URL]	Especificar el ambiente que se encuentra en un archivo JSON.
-d, --iteration-data [file]	Especificar qué archivo de datos usar (JSON o CSV).
-g, --globals [file]	Especificar un archivo global que se encuentra en un archivo JSON.
-n, --iteration-count [number] --working-dir [path]	Definir el número de iteraciones para ejecutar la colección.
--no-insecure-file-read	Ayuda a prevenir la lectura de archivos que se encuentren fuera del directorio definido.
--export-environment [path]	La ruta en la cual Newman pondrá el resultado de las variables de ambiente antes de que se termine la ejecución.
--export-globals [path]	La ruta en la cual Newman pondrá el resultado de las variables globales antes de que se termine la ejecución.
--export-collection [path]	La ruta en la cual Newman pondrá el resultado de la colección antes de que se termine la ejecución.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Actividad

¿En qué tipo de proyectos puedes utilizar Postman?

¿En qué proyecto piensas que no se puede utilizar Postman?

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Cierre

Las colecciones en Postman permiten crear diferentes requerimientos, de manera que puedas hacer pruebas al mismo tiempo en todos ellos, dichas pruebas pueden ser realizadas directamente en Postman o desde la consola de comandos, para esto deberás instalar Newman en el equipo. Las pruebas permitirán verificar con rapidez que todo funciona correctamente.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

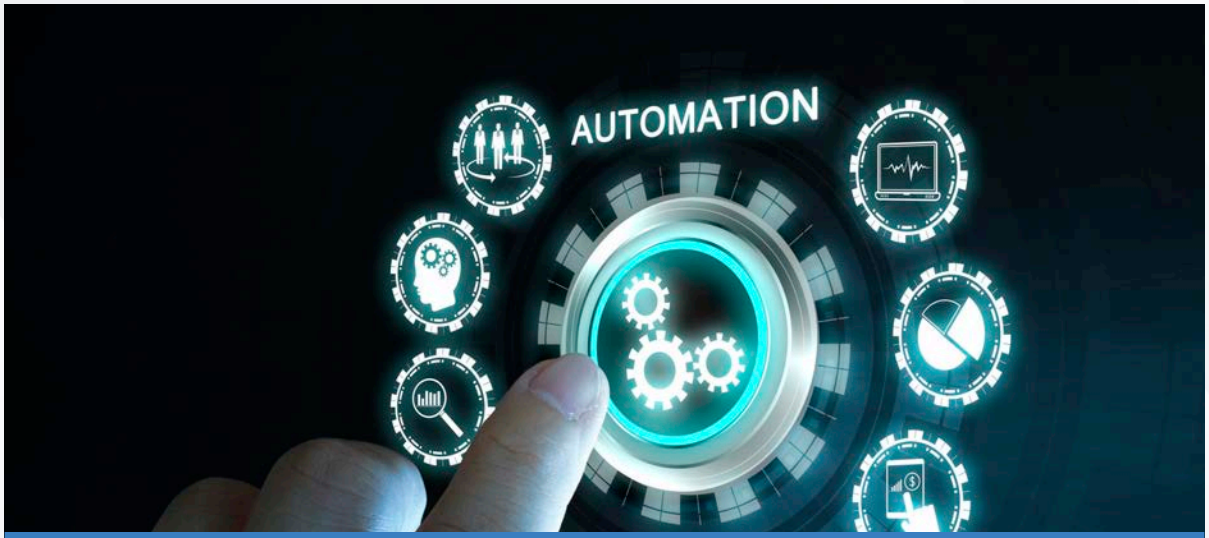
Universidad Tecmilenio

Software Testing

Introducción a la automatización de pruebas

Semana 10

Introducción



En los últimos años la evolución y el crecimiento de las ciencias de la computación, las tecnologías de información y telecomunicaciones, concretamente, el desarrollo de software, han provocado que la innovación tecnológica sea constante y muy rápida.

Lo que ha llevado a la necesidad de realizar pruebas para verificar el uso correcto de cada elemento en el desarrollo de software. A su vez con el paso del tiempo se han desarrollado métodos para automatizar estas pruebas, con el fin de aumentar la eficacia y la velocidad en el despliegue de proyectos de desarrollo funcionales.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Explicación

Conceptos básicos de la automatización

La automatización de las pruebas de software permitirá lograr una mayor cobertura y mejorará los tiempos de respuesta.

Las pruebas automatizadas permiten ejecutar instancias de pruebas que se repiten y aprovisionan recursos.

Identifican la correcta ejecución del código fuente y el aseguramiento del funcionamiento idóneo del proceso de negocio del software.

Los beneficios de la automatización son: reducen el tiempo de la ejecución de pruebas, minimizan los errores, ejecutan pruebas puntales en múltiples sistemas y plataformas.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Factores que aseguran la calidad de las pruebas

- **Correcto.** La ejecución de la prueba debe ser viable.
- **Exacto.** Debe tener un objetivo específico y alcanzable.
- **Económico.** Contar con los pasos necesarios para cumplir objetivo.
- **Confiable y repetible.** Se debe considerar como una iteración para siempre llegar al mismo resultado esperado.
- **Rastreable.** Se documenta los requisitos del caso de uso de la prueba.
- **Medible.** Permite hacer un seguimiento en cualquier punto y determinar el grado de cumplimiento hacia el objetivo.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Tipo de prueba	Descripción	Objetivo
Unitarias	Pruebas enfatizadas en encontrar errores en la lógica de programación.	Validar que cada unidad de software funcione de la manera adecuada.
Integración	Agrupa los diferentes módulos y son integrados de manera lógica como un grupo.	Encontrar errores en las interfaces, entre los módulos.
Smoke	Se encargan de asegurar que en una compilación del software, sea estable o no.	Confirmar la estabilidad de cada compilación.
Funcional	Validan la funcionalidad del software contra la funcionalidad de los requerimientos y/o especificaciones.	Probar cada función de la aplicación del software con la entrada correcta de datos.
Keyboard driven	Conjunto de acciones determinadas, con la finalidad de probar la aplicación de software en contextos específicos.	Detectar errores al realizar un conjunto de acciones específicas.
Regresión	Busca que los cambios realizados en el código o programa, no tengan un impacto negativo en la funcionalidad.	Asegurar la funcionalidad con recientes cambios.
Data driven	Análisis de la consistencia de los datos almacenados dentro de las bases de datos.	Validar la integridad de los datos al ser almacenados en las bases de datos.
Blackbox	Realiza las entradas de datos (inputs) y asegurar que las salidas (outputs) sean las esperadas.	Certificar que el sistema de software obtenga las salidas de datos esperadas al alimentar el sistema con las entradas de datos.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Explicación

Estrategias de automatización

Es el proceso de actividades que detalla el cómo llevar a cabo las pruebas.

El proceso de generación de la estrategia se podrá definir en los siguientes pasos:

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Actividad

¿Consideras que las pruebas de automatización mejoran la efectividad, la cobertura, y la velocidad de ejecución?

¿Consideras que las estrategias de automatización son necesarias?

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Cierre

La automatización de pruebas de software es una técnica que permite utilizar una herramienta de pruebas automatizada para ejecutar un caso de prueba, es la mejor manera de incrementar la efectividad, la cobertura, y la velocidad de ejecución. Para asegurar el éxito de la automatización es importante establecer una planeación mediante la creación de una estrategia de implementación de prueba. Al finalizar la automatización, la fase de mantenimiento permitirá actualizar las nuevas funcionalidades añadidas o las correcciones generadas para asegurar que el software opere correctamente.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Software Testing

Herramientas para automatización de pruebas

Semana 10

Introducción

Las herramientas de automatización de pruebas de software permiten a los desarrolladores aumentar la eficiencia y reducir los costos. Con la automatización de pruebas; un conjunto de estas se escribe una vez y luego se ejecuta automáticamente cada vez que la base de código cambia.

Para que las herramientas de automatización de pruebas sean eficaces deben configurarse adecuadamente para probar características específicas y verificar que el nuevo código funciona correctamente.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Herramientas de grabación vs frameworks de automatización

Los frameworks, permiten realizar un caso de prueba donde los ingenieros a pueden ejecutar pruebas del tipo “end-to-end”.

Las herramientas de automatización (low-code), se basan en una combinación entre el uso de una interfaz de usuario y las creaciones de scripts.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Herramientas de automatización de pruebas

Característica	Basada en código	Codeless
Conocimiento en programación.	Conocimiento en la lógica computacional y, en algunos casos, en lenguajes de programación específicos.	No es necesario contar con alguna experiencia previa, ya que hay frameworks que permiten trabajar con cero conocimientos en ciencias computacionales.
Personas involucradas.	Los probadores y desarrolladores son capaces de crear casos de automatización de pruebas por la necesidad del conocimiento en scripts.	Cualquier persona es capaz de utilizar este tipo de herramientas.
Complejidad.	Requiere el prerrequisito de conocer la estructura y uso del código de programación del framework a implementar.	Una complejidad menor basada en el uso de herramientas intuitivas.
Mantenimiento.	Debido a que las tareas de la preparación, codificación y puesta a marcha, un cambio en el desarrollo, puede llegar a ocasionar el mantenimiento del caso de prueba se extienda.	El mantenimiento puede ser más cómodo y rápido de implementar.
Seguridad.	Es más segura, ya que para ser usada en la mayoría de las veces, la prueba se hace de manera local.	Aun cuando se implementa de manera robusta, el hecho de que este tipo de herramientas se haga en la nube genera vulnerabilidades que pueden ser explotadas.
Tiempo de ejecución.	Requiere más tiempo en la planeación, generación del código aprovisionamiento de la infraestructura	Al no requerir infraestructura adicional y uso de un lenguaje de programación, es mucho más rápido en la construcción.
Flexibilidad.	Permite una completa personalización con base en el manejo de código fuente.	No hay punto de comparación, ya que están limitadas las configuraciones y opciones que ofrece la interfaz del usuario.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Tipos de herramientas de automatización

Automatización lineal o secuencia de comandos lineales:	• Se basa en la creación de una secuencia lineal de instrucciones.
Arquitectura basada en una librería de pruebas:	• Son almacenadas en una especie de módulos, que contienen conjuntos de scripts con las funcionalidades específicas de las tareas a realizar.
Basada en los datos:	• Son almacenadas en una especie de módulos, que contienen conjuntos de scripts con las funcionalidades específicas de las tareas a realizar.
Basado en palabras claves (basado en tablas):	• Requiere datos que se encuentran estructurados en una tabla y palabras clave, independientes a la herramienta de autorización.
Herramienta híbrida:	• Es una combinación de uno o varios frameworks de automatización aprovechando los puntos fuertes y tratando de mitigar aquellos puntos débiles.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Actividad

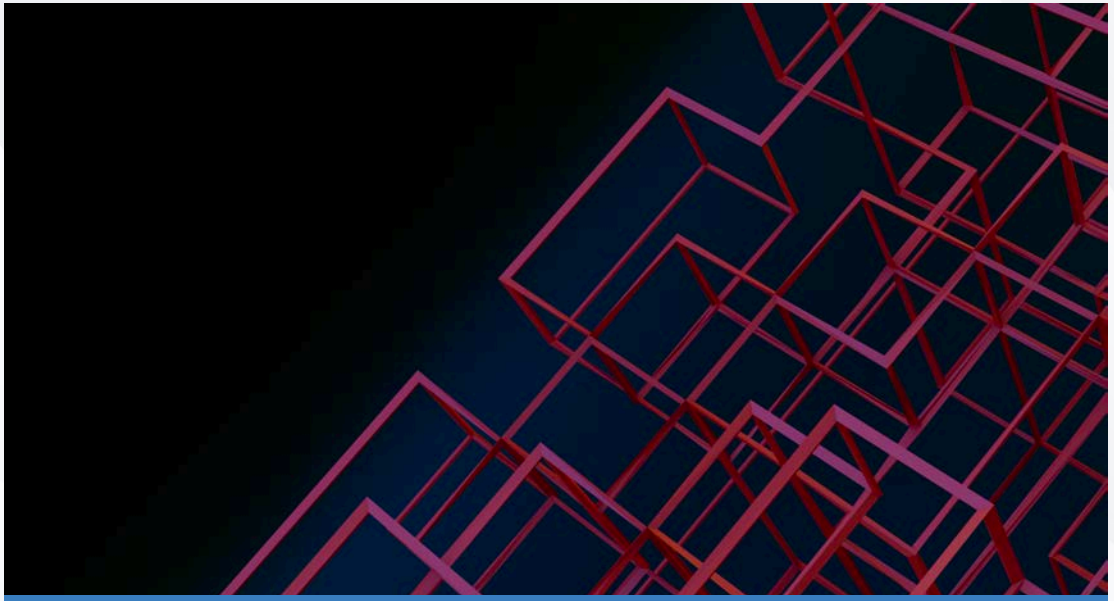
Considera la siguiente cualidad: la flexibilidad capaz de soportar una amplia gama de aplicaciones y lenguajes, ¿es necesaria para las herramientas de automatización?

¿Crees que deberían de existir más herramientas de automatización?

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Cierre



Es recomendable trabajar con un framework híbrido para las pruebas automatizadas, para lograrlo; es necesario encontrar una herramienta que pueda adaptarse fácilmente a tus procesos. Una herramienta de pruebas automatizadas adecuada debe ser flexible y capaz de soportar una amplia gama de aplicaciones y lenguajes.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .