

TEST AUTOMATION ENGINEER – FORMACIÓN INTEGRAL

{desafío}
latam_

CURSO:

TEST AUTOMATION
ENGINEER – FORMACIÓN
INTEGRAL

- Módulo 1: FUNDAMENTOS DEL TESTING DE SOFTWARE
- Módulo 2: CONTROL DE VERSIONES Y ENTORNOS DE DESARROLLO
- Módulo 3: FUNDAMENTOS DE PROGRAMACIÓN APLICADOS AL TESTING (JAVASCRIPT)
- Módulo 4: CARACTERISTICAS DE CYPRESS – AUTOMATIZACIÓN WEB I
- Módulo 5: CARACTERISTICAS DE CYPRESS – AUTOMATIZACIÓN WEB II
- Módulo 6: PRUEBAS DE APIS CON POSTMAN Y SUPERTEST

CURSO:

TEST AUTOMATION

ENGINEER – FORMACIÓN

INTEGRAL

- Módulo 7: HERRAMIENTAS DE PLAYWRIGHT Y PRUEBAS CON MÚLTIPLES NAVEGADORES
- Módulo 8: DESARROLLO GUIADO EN EL COMPORTAMIENTO (BDD) CON CUCUMBER.JS
- Módulo 9: HERRAMIENTAS DE AUTOMATIZACIÓN MÓVIL CON APPIUM
- Módulo 10: HERRAMIENTAS DE INTEGRACIÓN DE PRUEBAS EN CI/CD
- Módulo 11: HERRAMIENTAS DE DOCKER, ENTORNOS VIRTUALIZADOS Y PRUEBAS EN LA NUBE
- HERRAMIENTAS DE AUTOMATIZACIÓN DE UN FLUJO COMPLETO WEB + API + CI/CD



Te encuentras aquí

Módulo 8: Herramientas de automatización móvil con appium.



OBJETIVO ESPECÍFICO DEL MÓDULO

- UTILIZAR HERRAMIENTAS DE AUTOMATIZACIÓN MÓVIL CON APPIUM, DE ACUERDO A LAS APLICACIONES WEB, MÓVILES Y APIS.



¿Qué particularidades cree que presenta el testing en aplicaciones móviles en comparación con las aplicaciones web, y por qué podría ser clave simular gestos e interacciones en estos entornos?



TESTING EN APLICACIONES MÓVILES

- Las aplicaciones móviles presentan desafíos únicos: variedad de dispositivos, gestos táctiles, rendimiento, etc.
- **Android:** Sistema operativo más utilizado globalmente; se prueba mediante emuladores o dispositivos reales.
- **React Native:** Framework híbrido basado en JavaScript; genera apps tanto para Android como iOS desde un solo código base.



- Las pruebas móviles deben validar funcionalidad, UI, compatibilidad y usabilidad.
- Automatizar pruebas móviles mejora la cobertura y reduce errores en producción, especialmente en flujos críticos como login, pagos o navegación.

INSTALACIÓN Y CONFIGURACIÓN DE APPIUM

- Appium es una herramienta open-source que permite automatizar pruebas en apps móviles nativas, híbridas y web.
- **Requisitos previos:**
- Node.js, JDK, Android Studio, emulador configurado
- **Instalación de Appium:**

```
npm install -g appium
```



- **Configuración de entorno:**
- Configurar variables ANDROID_HOME
- **Iniciar Appium server y emulador:**

```
appium  
emulator -avd NombreDeTuEmulador
```

- Appium utiliza el protocolo WebDriver y puede integrarse con múltiples lenguajes y frameworks (JavaScript, Python, Java).

SIMULACIÓN DE GESTOS MÓVILES

- Appium permite simular interacciones típicas del usuario:
- **Tap:** driver.touchAction({ action: 'tap', x: 100, y: 200 })
- **Swipe:** arrastrar un elemento o desplazar la pantalla
- Scroll hasta ver un elemento visible
- **Teclado virtual:** ingreso de texto, enter, borrar
- **Rotación de dispositivo:** cambio horizontal ↔ vertical

{desafío}
latam_



- **Se pueden automatizar flujos reales como:**
 - Desbloqueo de pantalla
 - Validación de formularios
 - Gestos de navegación
-
- Simular correctamente la experiencia del usuario permite validar apps de forma realista, especialmente en test de regresión.



VALIDACIÓN DE UI Y COMPONENTES NATIVOS

- Se inspeccionan y validan componentes por su accesibilidad (accessibilityLabel, resource-id, etc.)
- **Validaciones comunes:**
- element.isDisplayed(), element.isEnabled(), element.getText()



- **Automatización típica:**
- Asegurarse que el botón de login exista
- Verificar que un mensaje de error se muestre si los datos son incorrectos
- Confirmar que una lista cargue correctamente elementos dinámicos
- En React Native, los componentes deben ser diseñados con identificadores accesibles para ser detectados por Appium o Detox.



DETOX: TESTING ESPECÍFICO PARA REACT NATIVE

- Detox es una herramienta desarrollada por Wix para pruebas E2E en apps React Native.
- **Ventajas:**
- Integra directamente con el ciclo de vida de React Native.
- Permite validaciones rápidas sin depender de WebDriver.
- Corre pruebas en dispositivos reales o simuladores.



- **Instalación básica:**

```
npm install --save-dev detox  
detox init -r jest
```

- **Ejemplo:**

```
await expect(element(by.id('loginButton'))).toBeVisible();  
await element(by.id('username')).typeText('cata');
```

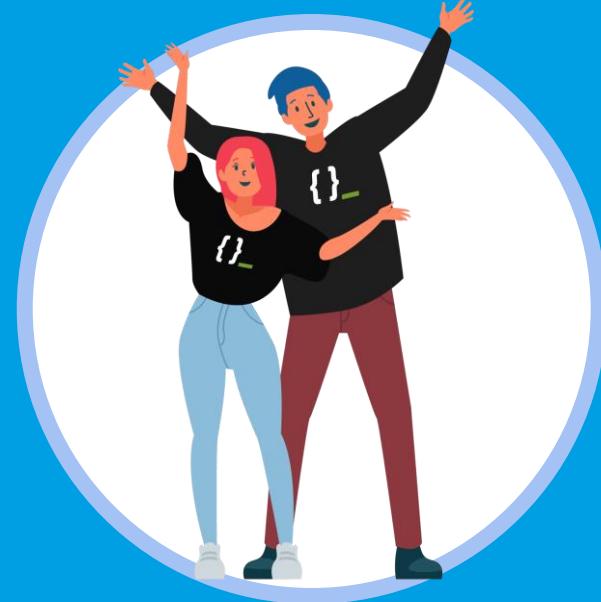
- Detox es una excelente opción cuando el proyecto está 100% en React Native, ya que proporciona velocidad, precisión y fácil integración.





**No olvide desarrollar los ejercicios que
contiene el Módulo...**

¿Cómo impactan herramientas como Appium y Detox en la capacidad de validar aplicaciones móviles de forma robusta y automatizada, y qué factores consideraría para elegir entre una u otra?



**Éxito en la evaluación parcial y
en la Prueba Final...**

{desafío}
latam_

*Academia de
talentos digitales*

