

Abode QA

A Hub For Testing Minds...

[HOME](#)[ABOUT US](#)[SELENIUM-WEBDRIVER](#)[JAVA TUTORIAL](#)[MANUAL TESTING](#)[KATALON](#)[DAILY BYTES!!](#)[CONTACT US](#)

INTRODUCTION TO CI/CD (CONTINUOUS INTEGRATION & CONTINUOUS DELIVERY)

[Katalon Studio Team](#) | [Leave a comment](#)

Machine Learning, Data Science and Deep Learning with Python Course for \$10.99



DevOps has brought with it the rise of CI/CD. In the present context, CI/CD practices are the most widely accepted choice to shorten software development and delivery cycle time. In this article, we will provide you with more insights into this notion, their benefits and how these practices are performed.

What is Continuous Integration?

Continuous Integration is a development practice that helps ensure that software components work together. Integration should be completed frequently; if possible, on an hourly or daily basis.

En una práctica de CI, los desarrolladores construyen, ejecutan y prueban el código en sus propias estaciones de trabajo antes de ingresar código en el repositorio de control de versiones. Después de realizar los cambios en el repositorio, se pone en marcha una cadena de eventos. Un primer paso típico en esta cadena es construir la última versión del código fuente. Si la construcción es exitosa, se ejecutan pruebas unitarias. Si la prueba de la unidad es exitosa, la compilación se implementa en entornos de prueba donde se realizan las pruebas del sistema (generalmente utilizando pruebas automatizadas). Se notifica al equipo sobre el estado de este proceso y se entrega un informe para proporcionar detalles, como el número de compilación, los defectos y el número de pruebas.

¿Qué es la entrega continua?

La entrega continua (CD) es la capacidad de implementar su código integrado en producción sin la necesidad de intervención humana. CD recoge donde termina la integración continua. Si bien CI es el proceso para compilar y probar automáticamente, el CD implementa todos los cambios de código en una compilación en el entorno de prueba o de ensayo. El CD permite lanzar compilaciones al entorno de producción cuando sea necesario. Al permitir que el equipo se despliegue a voluntad, el CD reduce efectivamente el tiempo de comercialización.

Antes de implementar el software en producción, el proceso del CD incluye realizar pruebas automatizadas del sistema, pruebas unitarias (incluidas las pruebas de API y pruebas de carga) y pruebas de integración. Los pasos de CI a CD generalmente se completan automáticamente, incluidas las pruebas automatizadas a nivel de unidad, integración y sistema. Como las pruebas pueden fallar en cualquier nivel y

LAS CATEGORÍAS

- Artículos
- Byte diario
- Fitnesse
- Tutorial de GIT
- Tutorial de Java
- jQuery
- Tutorial Mac
- Pruebas de software
 - Pruebas de automatización
 - Blazemeter
 - Calabaza
 - Pepino
 - Katalon
 - Selenio
 - Selenium Grid Tutorial
 - Selenium WebDriver Tutorial
 - Prueba manual
 - Insecto
 - Pruebas en la nube
 - Pruebas funcionales
 - Pruebas de instalación
 - Pruebas de regresión y re-pruebas
 - Pruebas de seguridad
 - Caso de prueba
 - Escenario de prueba y casos de prueba
 - Examen
- Publicación patrocinada



ETIQUETAS

La prueba de aceptación ágil Scrum interfaz de programación de aplicaciones assertEquals () **las**

entorno, CI / CD debe incluir un canal de comentarios para informar rápidamente los fallos a los desarrolladores.

Más información: [¿Qué es CI / CD? Integración continua y entrega continua](#)

Canalización de flujo de trabajo CI / CD

Una tubería de CI típicamente involucra las siguientes tareas:

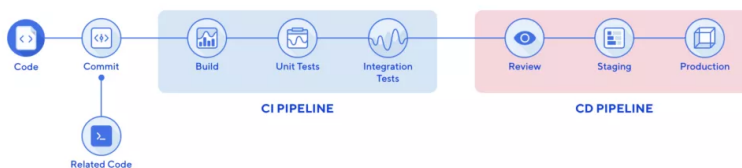
- Detectar cambios en el repositorio de código fuente (aparecen nuevas confirmaciones)
- Análisis de calidad del código fuente
- Construir
- Ejecutar todas las pruebas unitarias.
- Ejecutar todas las pruebas de integración.
- Generar artefactos desplegables
- Estado del informe

Si uno de los pasos anteriores falla:



Aprende SOAPUI paso a paso

- La integración puede detenerse o continuar dependiendo de la gravedad del defecto y la configuración
- Los resultados son notificados al equipo por correo electrónico o sistema de chat.
- El equipo arregla los defectos y los vuelve a cometer.
- Las tareas se realizan de nuevo.



Un canal de CI / CD es una ruta para entregar una unidad de cambio que comienza desde el desarrollo hasta la entrega, generalmente consta de las siguientes fases principales:

Fase 1: Cometer

Cuando los desarrolladores completan un cambio, envían el cambio al repositorio.

Fase 2: Construir

El código fuente del repositorio está integrado en una compilación.

Fase 3: automatizar pruebas

Las pruebas automatizadas se ejecutan contra la construcción. La automatización de pruebas es un elemento esencial de cualquier canalización de CI / CD.

Fase 4: Implementar

pruebas de automatización

de pruebas Herramientas de Automatización de errores Errores ChromeDriver base de datos de pruebas de Eclipse función Fitnesses Pruebas funcionales getText () Google Chrome Pruebas de Instalación de Java JavaScript JDBC jQuery

Katalon grabadora Katalon

Estudio Características Katalon

Estudio estudio Katalon de código abierto herramienta de pruebas de carga Pruebas manuales Programación

Control de calidad Pruebas de regresión Requisito

Selenium Selenium

IDE Selenium RC Selenium

Webdriver Configuración del

nodo en la red de selenium

Software Testing

String en Java Test Case

testing TestNg Test Plan

Prueba de la unidad Aplicación web

Webdriver

PRINCIPALES PUBLICACIONES Y PÁGINAS

- Xpath en Selenium :: Contiene () y comienza con la función () en Xpath
- Programas java comúnmente solicitados en Selenium Entrevista
- Cómo usar assertEquals () en WebDriver usando driver.getText ()
- Encontrar elementos secundarios en WebDriver usando findElements
- assertTrue (mensaje, condición) en Selenium WebDriver y su implementación
- Cargue y descargue el archivo desde el servidor FTP utilizando Java FTP Client
- Revisiones, tutorial e inspección en pruebas de software.
- Maximizar la ventana usando Selenium WebDriver y usando el kit de herramientas Abstract Window
- Desafíos para ejecutar los scripts de Selenium WebDriver en el navegador IE
- Selenium WebDriver Quiz

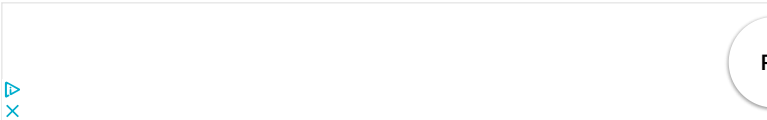
La versión construida se entrega a producción.

Importancia de la automatización de pruebas para CI / CD

La automatización de pruebas es un componente crucial de cualquier canal de CI / CD. Los beneficios de aplicar CI / CD no pueden realizarse si faltan pruebas automatizadas y un bajo nivel de cobertura de prueba. Los equipos deben realizar pruebas automatizadas en todos los niveles, incluidas las pruebas de unidad, integración y sistema.

Conclusión

La adopción de CI / CD está en aumento, este crecimiento posiblemente indicó la forma en que los desarrolladores modernos enfocan la construcción de grandes productos. CI / CD es el núcleo de DevOps, y tanto CI como CD constituyen dos de los procesos más importantes en cualquier canal de DevOps. Así que ahora es un buen momento para que comiences a aprender CI / CD si no lo has hecho, será un conjunto de habilidades valiosas con una gran demanda de futuro.



Artículos Pruebas de automatización Pruebas de software

«Las 3 últimas investigaciones de Gartner sobre la industria de pruebas de software

DEJA UNA RESPUESTA

Enter your comment here...

Este sitio utiliza Akismet para reducir el spam. [Aprenda cómo se procesan los datos de sus comentarios](#) .