



UNIVERSIDAD PRIVADA DE TACNA  
INGENIERIA DE SISTEMAS

TITULO:

**”Herramientas de gestión de pruebas”**

**CURSO:**

CALIDAD Y PRUEBAS DE SOFTWARE

**DOCENTE:**

Ing. Patrick Cuadros Quiroga

Integrantes:

Maldonado Cancapi, Carlos Alejandro	(2018000660)
Villanueva Yucra, Josue Joel	(2018000722)
Contreras Murguia, Jose Manuel	(2016056346)
Rojas Bedregal, Brian Erik	(2018060904)
Mamani Laura, Juan Carlos	(2017059565)

Tacna - Perú  
2021

## Resumen

Con el auge de las aplicaciones web y basadas en la nube, numerosos programas Han surgido herramientas que nos permiten gestionar diversas tareas. En el área de ingeniería de software y, en particular, pruebas de software (pruebas de software), existen nuevas herramientas para registrar información y presentar informes de estado en el diferentes fases del ciclo de vida, según el desarrollo del software metodologías utilizadas. También contamos con nuevas herramientas para automatizar Pruebas.

**Palabras Clave:** Fases del ciclo de vida, pruebas de software

---

## Abstract

With the rise of web and cloud-based applications, numerous software tools have emerged that allow us to manage various tasks. In the area of software engineering and in particular software testing (Software Testing), there are new tools to record information and present status reports in the different phases of the life cycle, according to the software development methodologies used. We also have new tools to automate Tests.

**Keywords:** class structure, software design pattern.

# 1 Introducción

Con el auge de las aplicaciones web y basadas en la nube, numerosos programas Han surgido herramientas que nos permiten gestionar diversas tareas. En el área de ingeniería de software y, en particular, pruebas de software (pruebas de software), existen nuevas herramientas para registrar información y presentar informes de estado en el diferentes fases del ciclo de vida, según el desarrollo del software metodologías utilizadas. También contamos con nuevas herramientas para automatizar Pruebas.

# 2 Desarrollo

## 2.1 ¿QUÉ SON LAS HERRAMIENTAS DE GESTIÓN DE PRUEBAS?

Es la herramienta que proporciona soporte a la gestión de pruebas y control de parte del proceso de pruebas. A menudo tiene varias capacidades, tales como gestionar los productos de soporte de pruebas, planificación de pruebas, registro de resultados, seguimiento del proceso, gestión de incidencias y generación de informes de las pruebas.

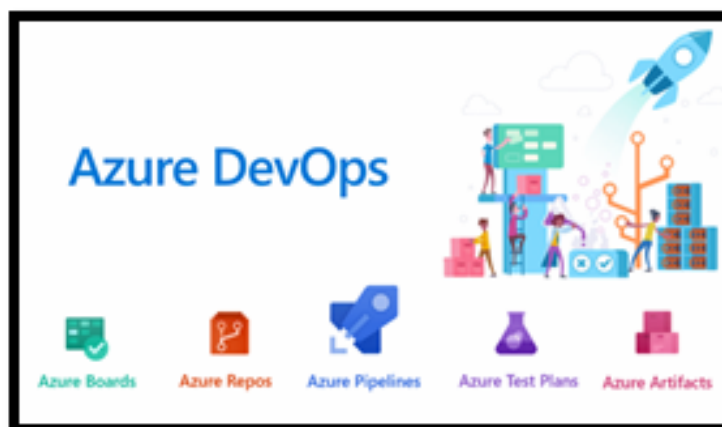
## 2.2 ¿QUÉ ES AWS CODEPIPELINE?

Es un servicio de Amazon Web Services que nos ayuda a automatizar los procesos de lanzamiento para obtener actualizaciones rápidas y confiables. AWS Codepipeline compila, prueba e implementa tu código cada vez que hay un cambio, en función a los modelos de proceso de los releases que hagas.



## 2.3 ¿QUÉ ES AZURE DEVOPS?

Es una plataforma de SaaS (software como servicio) de Microsoft que nos proporciona una cadena de herramientas DevOps de punto a punto para desarrollar e implementar software. También proporciona alojamiento Git privado ilimitado, compilación en la nube para la integración continua, planificación ágil y administración de versiones para la entrega continua a la nube y en las instalaciones. Incluye amplio soporte IDE.



## 2.4 COMPARATIVA ENTRE AMBOS: VENTAJAS

Se muestra una comparativa en ventajas entre AWS Codepipeline y Azure DevOps

AWS CodePipeline	Azure DevOps
Fácil de instalar	Completo, flexible e intuitivo
Servicio administrado	Gran ecosistema extendido
Integración con GitHub	Integración con Azure
Ejecución en paralelo	Integración con GitHub
Despliegue automático	Soporte Open Source y gratis para Stakeholders

## 2.5 COMPARATIVA ENTRE AMBOS: DESVENTAJAS

Se muestra una comparativa en desventajas entre AWS Codepipeline y Azure DevOps

AWS CodePipeline	Azure DevOps
No hay pizarras de proyecto	Sigue siendo dependiente de C#
No tiene integración con las herramientas de 365 de "Power"	Pobre integración de Jenkins
Más básico que Azure DevOps	No tiene todos los requerimientos de herramientas de administración
No puede configurar repositorios de código localmente	Capacidad de equipos multifuncionales no es visible

## 2.6 COMPAÑÍAS QUE USAN ESTAS HERRAMIENTAS:

Se muestran las compañías que utilizan las herramientas.

AWS CodePipeline	Azure DevOps
CRED	<u>Via Varejo</u>
Swingvy	Microsoft
Affirm	QRPoint
HENNGE	<u>Queue-it</u>

## 2.7 HERRAMIENTAS QUE PUEDEN SER INTEGRADAS:

Se muestran herramientas que puede ser integradas en un futuro

AWS CodePipeline	Azure DevOps
GitHub	GitHub
Jenkins	Git
Amazon EC2	Docker
Amazon S3	<u>Slack</u>

Figura 1: Ranorex

## 2.8 ALGUNAS BUENAS ALTERNATIVAS A AMBOS:

- AWS CodeDeploy
- Jenkins
- AWS CodeBuild
- TeamCity

- Bamboo

### 3 Conclusiones

#### CodePipeLine

- CodePipeline es una herramienta de entrega continua e integración continua muy flexible.
- Facilita un poco la implementación en el entorno de AWS. Hay una gran disponibilidad de datos no estructurados.
- Se usa bastante para gestionar la CI/CD (Integración continua e integración delivery)

#### Azure DevOps

- Es muy fácil de configurar y usar si tiene alguna experiencia con procesos ágiles.
- Facilita un poco la implementación en el entorno de AWS. Hay una gran disponibilidad de datos no estructurados.
- Las barreras de entrada iniciales son extremadamente bajas, ya que los primeros 5 usuarios pueden aprovechar la herramienta de forma gratuita. Encontré la característica / funcionalidad general más fácil de usar y más accesible que herramientas similares.
- Si ya es un usuario de git, esto se integra directamente con los repositorios de git, lo que facilita la transición.
- La herramienta también está integrada con muchos otros productos de Microsoft, por lo que si tiene una tienda centrada en Microsoft, puede aprovechar el ecosistema más amplio.

### 4 Recomendaciones

Aquí una serie de aspectos a tener en cuenta al momento de elegir una herramienta para la gestión de prueba

- La categoría de defectos
- El lenguaje de programación y entorno de desarrollo
- El proceso de configuración y gestión de datos de prueba
- El control de versiones y CI (Integración Continua)
- Los reportes
- Las plataformas compatibles y etiquetado

## Referencias

- [1] Espinosa, S. G. (2016 de 03 de 1). Personalización del proceso de pruebas unitarias empleando la herramienta NUnit. Obtenido de serie científica: <https://publicaciones.uci.cu/index.php/serie/article/view/811>
- [2] Ríos, J. R. (23 de 09 de 2016). Evaluación de los Frameworks en el Desarrollo de Aplicaciones Web con Python. Obtenido de revistas unla: <http://revistas.unla.edu.ar/software/article/view/1149>
- [3] sierra, f. (01 de marzo de 2017). Estudio y análisis de los framework en php basados en el modelo vista controlador para el desarrollo de software orientado a la web. Obtenido de Investigación y desarrollo en TIC: <http://revistas.unisimon.edu.co/index.php/identific/article/view/2480>
- [4] sierra, f. (01 de marzo de 2017). Estudio y análisis de los framework en php basados en el modelo vista controlador para el desarrollo de software orientado a la web. Obtenido de Investigación y desarrollo en TIC: <http://revistas.unisimon.edu.co/index.php/identific/article/view/2480>
- [5] Paloma Díaz, Susana Montero, Ignacio Aedo. (2005). Ingeniería de la web y patrones de diseño.
- [6] Villón Moreno, M. S. (abril de 2019). Plataforma tecnológica para contribuir a la planeación urbana en la ciudad de Guayaquil dirigido a la transportación enfocado al diseño de pruebas que permitan la verificación y validación del software usando herramientas de pruebas ágiles. Obtenido de repositorio institucional de la universidad de guayaquil: <http://repositorio.ug.edu.ec/handle/redug/39670>
- [7] Paloma D´iaz, Susana Montero, Ignacio Aedo. (2005). Ingeniería de la web
- [8] <https://www.trustradius.com/products/aws-codepipeline/reviews>
- [9] <https://www.trustradius.com/compare-products/aws-codepipeline-vs-azure-devops>
- [10] <https://www.infoworld.com/article/3271126/what-is-cicd-continuous-integration-and-continuous-delivery-explained.html>