

Internet de las Cosas y Big Data

Tecnología Big Data: Semana 01

Actividad 01

Responda de manera breve cada una de las preguntas propuestas:

1. ¿Qué es la Integración continua (CI)?

La Integración Continua es una práctica en la que los desarrolladores suben continuamente los avances de un proyecto a un repositorio compartido, con ayuda de procesos automatizados se realizan pruebas constantemente para hallar los errores de manera ágil.

2. ¿Qué es la Entrega continua (CD)?

Es una extensión de CI y también consiste en la revisión y prueba automatizada de un proyecto para encontrar y depurar errores. Pero esto con el objetivo de lanzar versiones funcionales del software, en caso de que surjan más fallas las siguientes versiones con las correcciones no demorarán mucho en ser lanzadas.

3. ¿Por qué el control de versiones es importante en los proyectos de CI/CD?

Ambas metodologías requieren hacer cambios frecuentemente en los proyectos, así como tener que regresar a versiones anteriores a los cambios. También se debe considerar que se requiere el trabajo en equipo y un sistema de versiones en una buena herramienta para esto.

4. ¿Qué es DevOps?

DevOps es un modelo de trabajo en el que se integran las tareas desempeñadas por los equipos de Desarrollo y Operaciones.

El equipo de desarrollo se encarga de programar, depurar y empaquetar el código para que este realice las tareas requeridas de forma segura en las infraestructuras que se hayan solicitado. Y el equipo de Operaciones se encarga de la implementación de hardware, sistema operativo, seguridad y estabilidad de las plataformas en las que se ejecutará el código.

DevOps empaqueta estas tareas bajo filosofías de trabajo colaborativo para que la implementación se realice de forma rápida manteniendo una buena comunicación entre los equipos.

Las fases de DevOps se consideran un ciclo:

- **Plan:** Acá definimos las labores, los requerimientos que se necesitan para implementar en nuestra plataforma y para esto usamos herramientas como issues o boards.
- **Create:** Es escribir el código necesario para resolver el problema de negocio que tenemos. Todo este código podemos tenerlo en un solo lugar para colaborar y se hace uso de repositorios, también branches o tags.
- **Verify:** Se corren nuestras pruebas automatizadas donde definimos las reglas a probar.

- **Package:** Empaquetamos nuestro código para correr en una infraestructura determinada. Normalmente se hace en un contenedor de Docker.
- **Release:** Una nueva versión de nuestro código y llega a producción.
- **Configure:** Se puede cambiar nuestro cluster de kubernetes, mandar instrucciones para manejar el nuevo estado de nuestra aplicación.
- **Monitor:** Cómo nuestro código está funcionando, qué tipo de performance ocurre en los dispositivos de nuestros clientes.

5. ¿Porque sería importante implementar DevOps en proyectos de IoT?

Los proyectos de IoT requieren implementar diversas tecnologías, tanto de software como hardware por lo que se requiere el trabajo multidisciplinario de distintos profesionales. DevOps brinda las herramientas necesarias para la correcta comunicación de los equipos, detección temprana de errores y lanzamientos óptimo que cumplan los requerimientos técnicos.