

1. Conceito de CI e CD

CI/CD (continuous integration/continuous delivery) são práticas combinadas a fim de entregar aplicações frequentemente aos clientes. Para que essa entrega seja possível, é utilizada, nas fases do desenvolvimento de aplicações, a automação. Os conceitos designados a esse método são: integração, entrega e implantação contínuas. O CI/CD, possibilita solucionar os problemas que a inclusão de novos códigos pode causar para os indivíduos da área de desenvolvimento e operações.

Ademais, o CI/CD realiza a prática de monitoramento e automação contínuos em todas as fases das aplicações, também as etapas de teste e integração, além da entrega e implantação. Unidas, essas práticas relacionadas são, em diversos momentos, chamadas de "pipeline de CI/CD", sendo compatíveis com o trabalho conjunto das áreas de operações e desenvolvimento com métodos ágeis.

O "CI" se refere à integração contínua, que é um processo de automação para desenvolvedores. Uma CI de sucesso é quando as novas modificações no código de uma aplicação são desenvolvidas, testadas e consolidadas regularmente em um repositório compartilhado, para evitar conflitos entre ramificações quando muitas aplicações são desenvolvidas ao mesmo tempo.

O "CD" se refere à entrega contínua e/ou à implantação. Sendo qualquer um dos casos (e/ou) se trata da automação de etapas avançadas do pipeline, porém em alguns momentos são usados separadamente, para ilustrar o nível de automação presente na aplicação.

2. Quadro Ferramentas

Fase	Conceito	Ferramenta de apoio
Operate	Nesta fase a versão está disponível para uso dos clientes	Kubernetes < kubernetes.io/ > Ansible < https://www.ansible.com/ >
Monitor	Nesta etapa, os produtos ou aplicativos entregues a um usuário foram monitorados para anotar quaisquer falhas de tempo de atividade e tempo de	Grafana < https://grafana.com/ > Graylog < https://www.graylog.org/ >

	inatividade, erros.	
Plan	Após o monitoramento, ele reúne todas as informações e comentários do cliente e planeja as alterações que precisam ser feitas para torná-lo melhor.	Openstack < https://www.graylog.org/ > Trello https < https://trello.com/ >
Code	Nesta etapa, os desenvolvedores escrevem o código em qualquer plataforma para desenvolver o produto para um cliente.	Git < https://git-scm.com/ > GitLab < https://gitlab.com/ >
Build	construir onde a versão básica do produto é criada usando uma linguagem de programação adequada.	Gradle < https://gradle.org/ > Sonatype Nexus < https://www.sonatype.com/products/nexus-repository >
Test	passo do teste, no qual os produtos construídos são testados usando as ferramentas de teste de automação	Azure < https://azure.microsoft.com/ > JUnit < https://junit.org/junit5/ >
Release	Esta etapa envolve planejar, programar e controlar o processo construído em um ambiente diferente.	DockerHub < https://hub.docker.com/ > Python Package Index < https://pypi.org/ >
Deploy	Aqui todos os produtos e arquivos de implantação são executados no servidor.	Docker < https://www.docker.com/ > aws < https://aws.amazon.com/ >

3. Referências:

Guia completo: DevOps e a cultura do código - Gaea

Ciclo de vida do DevOps | Fases e ciclo de vida

O que é CI/CD? (redhat.com)