



KUBERNETES  
**DAYS**

| 4-Linux



KUBERNETES  
**DAYS**

**4-Linux**

**Deploy no Kubernetes com  
GitLab-CI**



# Sobre a 4Linux

Fundada em 2001, a 4Linux é líder de mercado em cursos de Linux e open source com números impressionantes: mais de 200.000 alunos treinados, mais de 5.228 empresas atendidas e mais de 40 diferentes cursos altamente especializados. Somos uma das poucas escolas de TI que também atua em consultoria e isso traz inúmeros benefícios aos nossos alunos, com uma metodologia de ensino única.



**Somos a melhor e mais renomada escola brasileira de cursos de Linux e open software**

**+40**

diferentes cursos  
altamente especializados

**+200.000**

alunos  
treinados

**+5,228**

empresas  
atendidas

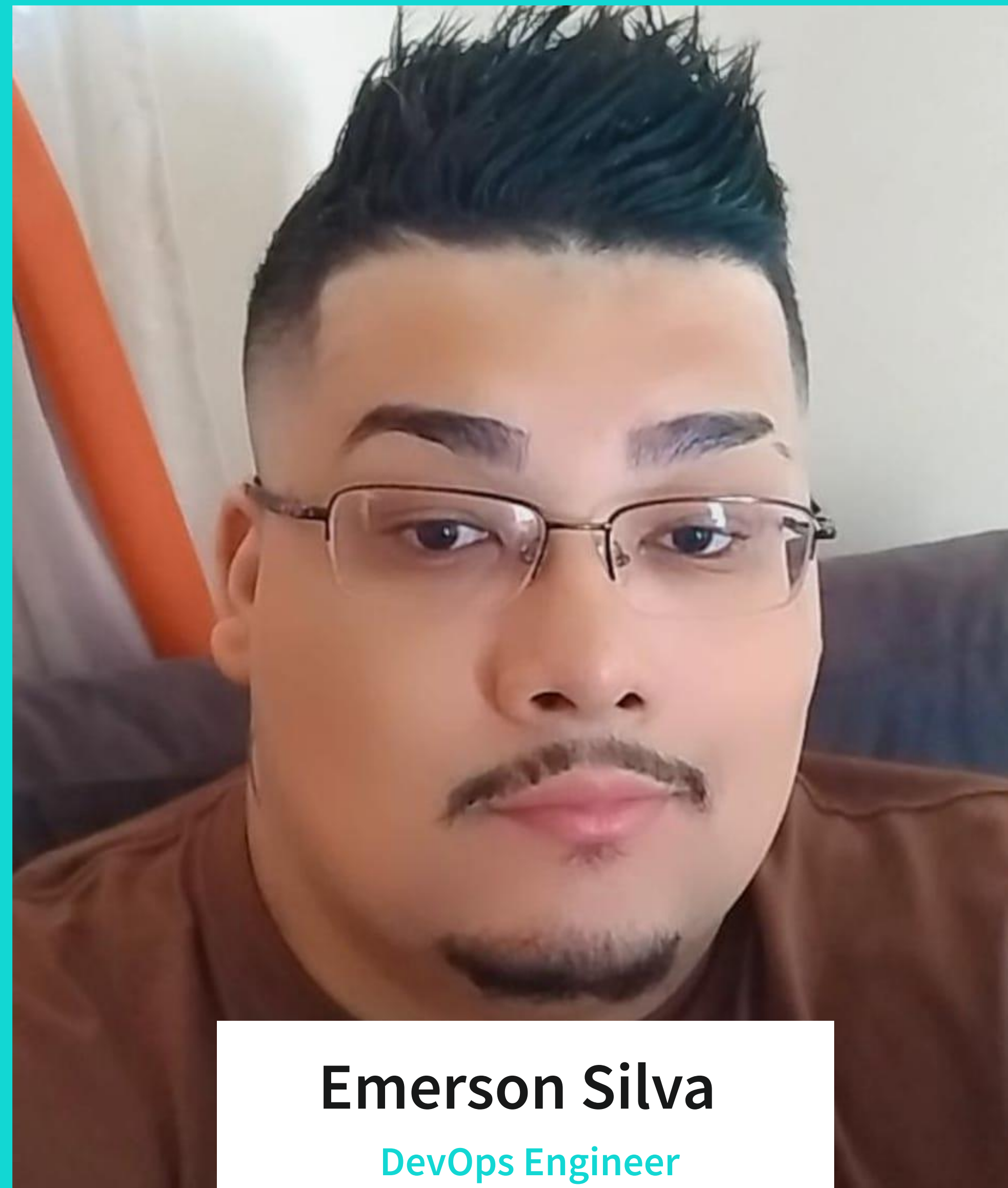


# Conheça o construtor

## MINI-BIO

Atua com foco em Linux, Monitoramento, Containers, Cloud, Automação e DevOps, possuindo mais de 6 anos de experiência profissional em Tecnologia da informação. Com expertise como instrutor de cursos voltados a Docker, Kubernetes, Terraform e CI/CD.

- GitLab Associate
- DevOps Essentials
- Scrum Essentials
- AWS CCP



**Emerson Silva**

DevOps Engineer

# Tema e objetivos da live

01

## O que é Kubernetes

Sei que você conhece

02

## O que é GitLab, Gitlab Runner

Conversa rápida

03

## Falando sobre CI/CD

O que é, para que serve, etc

04

## Sonarqube e Testes Unitários

resenha rápida

05

## Sobre o nosso Deploy

Vamos entender o que vamos fazer o deploy e como



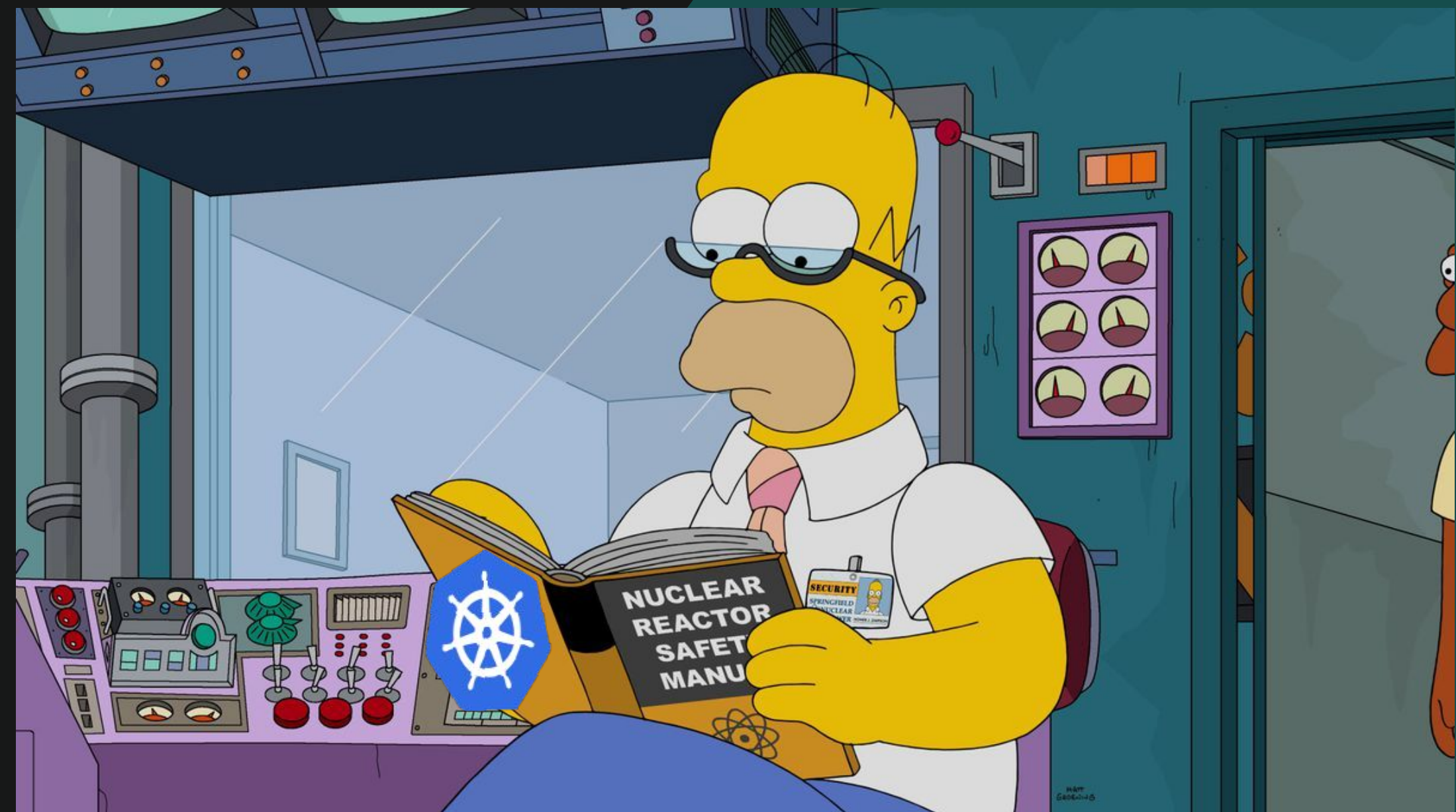
KUBERNETES  
**DAYS**

4.Linux



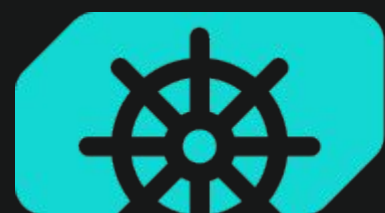
# O que é o Kubernetes

- Kubernetes é uma plataforma de código aberto, portátil e extensiva para o gerenciamento de cargas de trabalho e serviços distribuídos em contêineres, que facilita tanto a configuração declarativa quanto a automação
- O nome Kubernetes tem origem no Grego, significando timoneiro ou piloto. K8s é a abreviação derivada pela troca das oito letras "ubernete" por "8", se tornando K"8"s.



# O que é o GitLab

- O GitLab é uma plataforma de gerenciamento de código-fonte baseada em Git. Ele oferece recursos de controle de versão distribuído, permitindo que desenvolvedores trabalhem em equipe e colaborem em projetos de software.ação
- O GitLab fornece um ambiente completo para o desenvolvimento de software, que inclui recursos como rastreamento de problemas, integração contínua, entrega contínua e um sistema de registro de contêiner integrado.

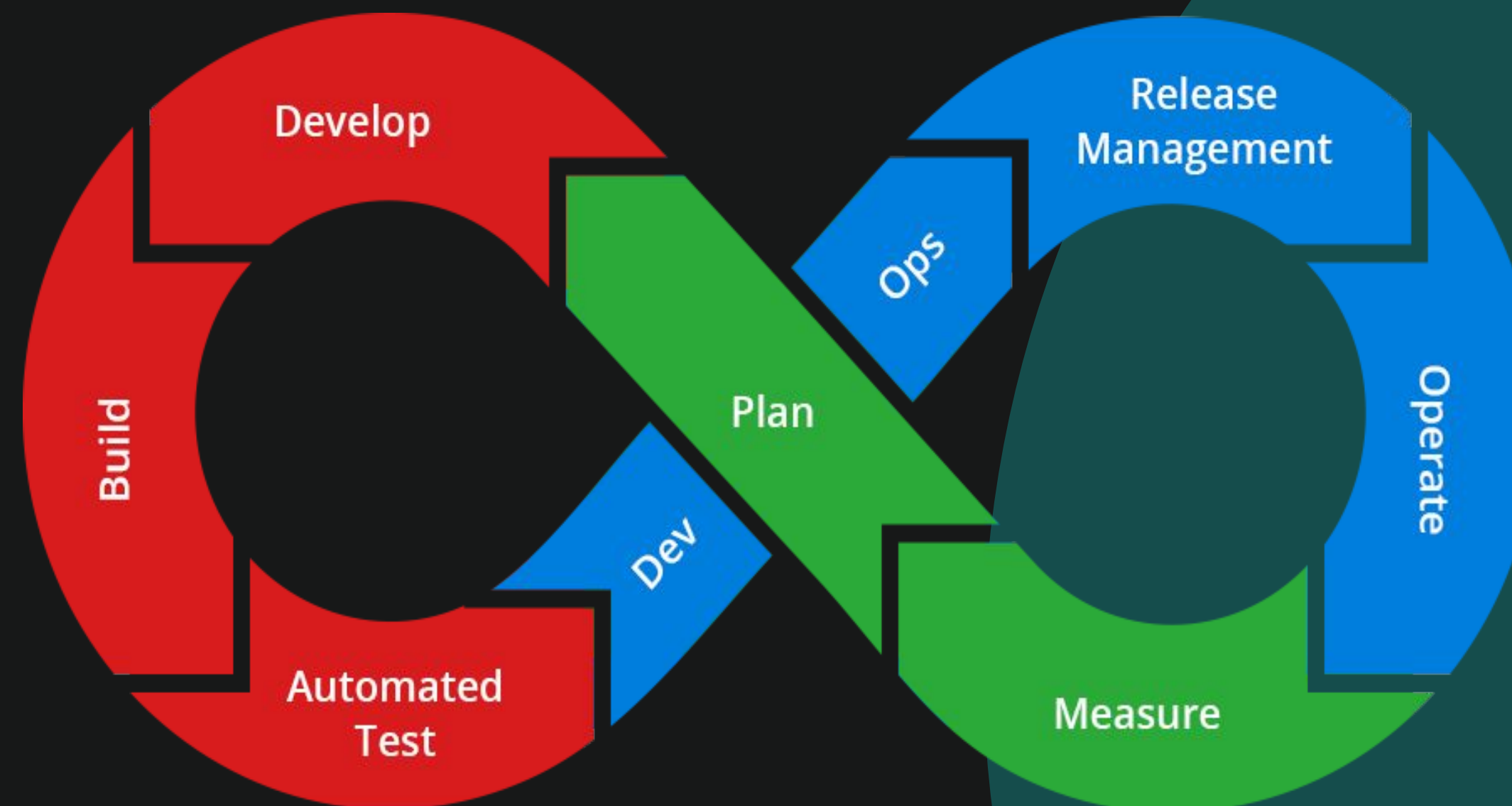


KUBERNETES  
**DAYS**

4.Linux

# CI (Continuous Integration) e CD (Continuous Delivery/Deployment)

- CI (Continuous Integration) e CD (Continuous Delivery/Deployment) são práticas de desenvolvimento de software que visam automatizar e agilizar o processo de construção, teste e implantação de aplicativos.



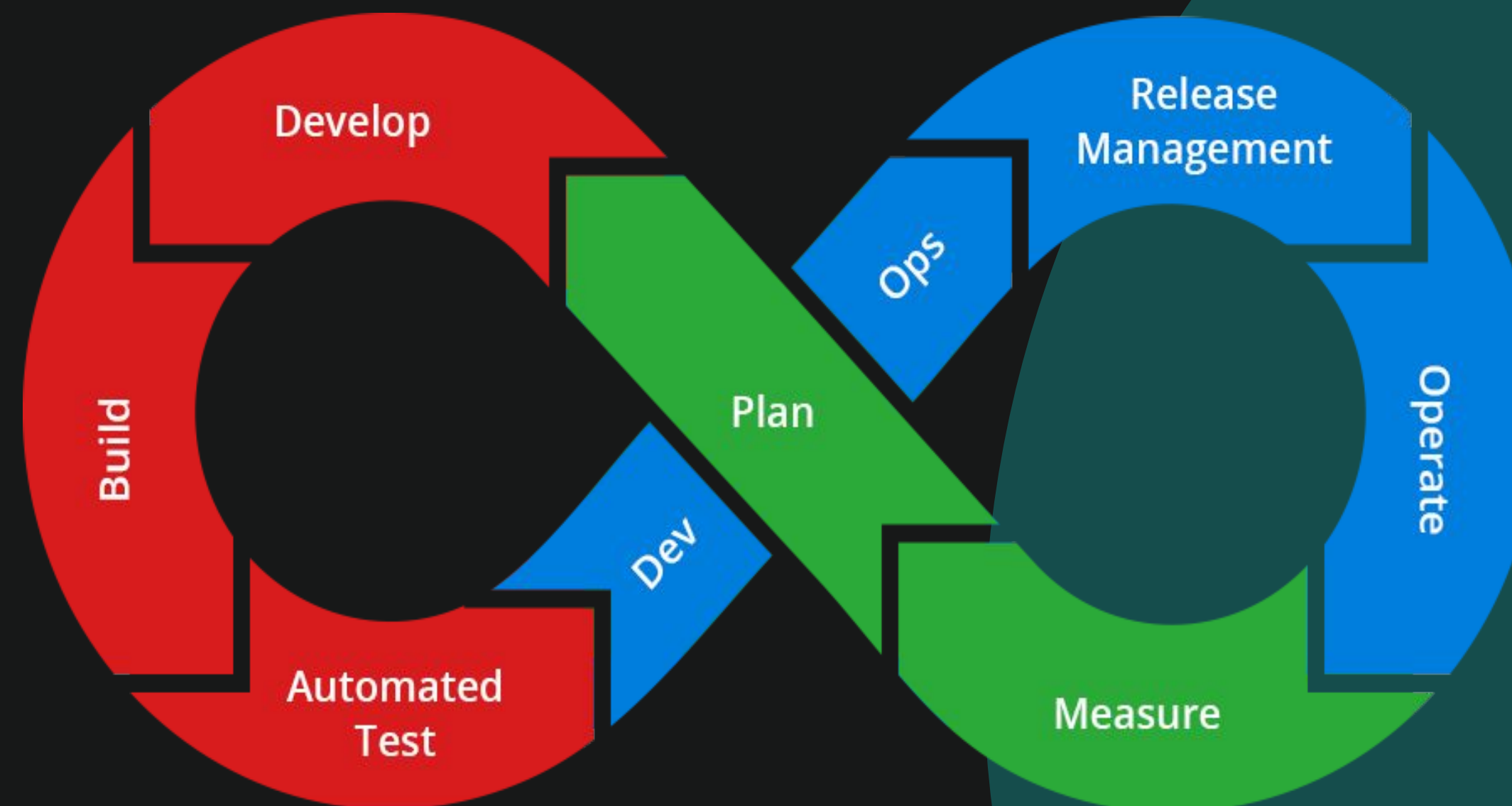
KUBERNETES  
**DAYS**

4.Linux



# Continuous Integration (Integração Contínua)

- A Integração Contínua envolve a integração frequente e automatizada do código-fonte de vários desenvolvedores em um repositório compartilhado. O objetivo é detectar problemas de integração o mais cedo possível, garantindo que o código seja sempre testado e integrado com sucesso

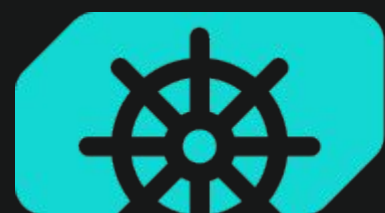
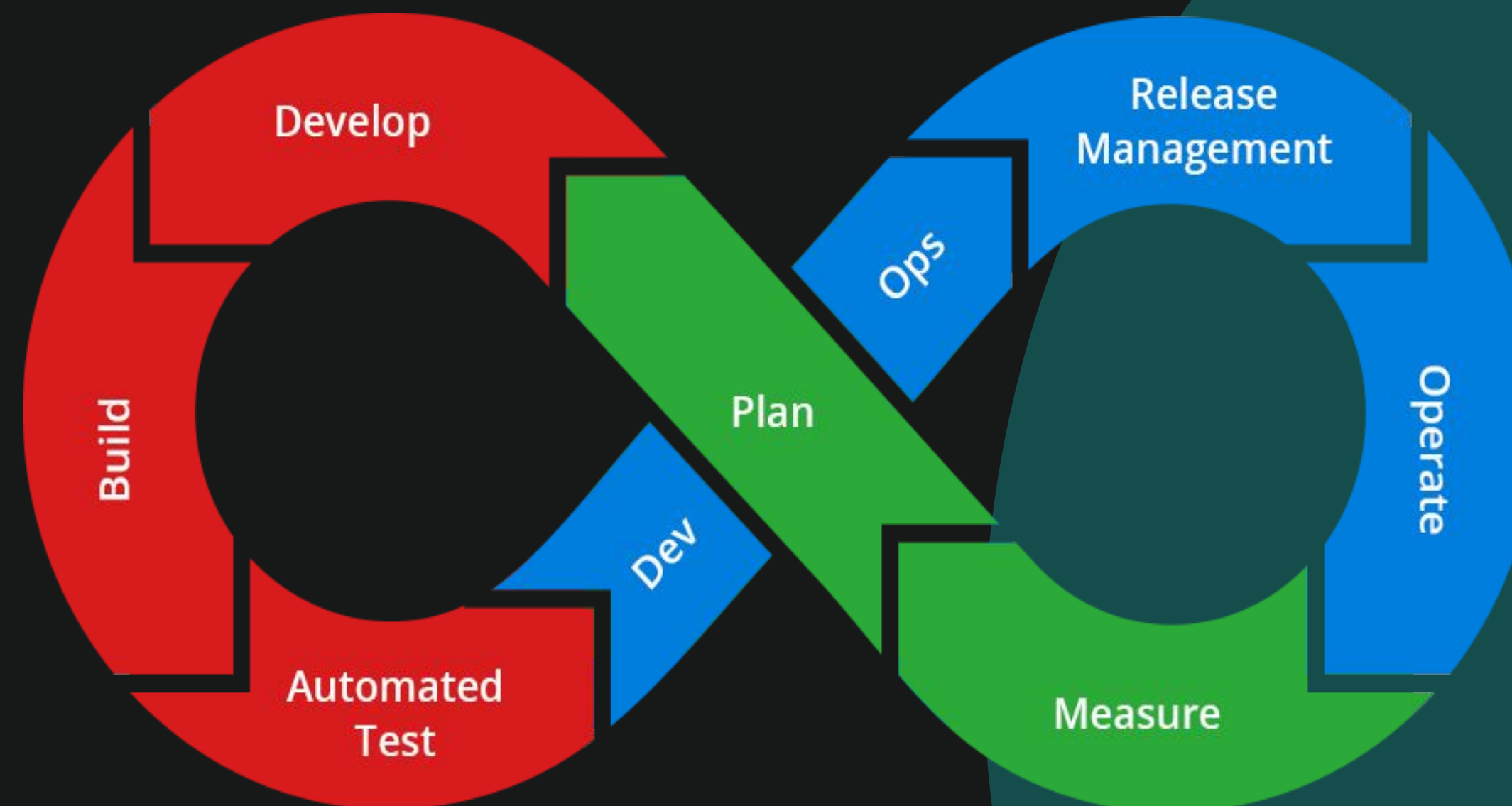


KUBERNETES  
**DAYS**

4.Linux

# Continuous Delivery (Entrega Contínua)

- A Implantação Contínua vai um passo além da Entrega Contínua, automatizando completamente o processo de implantação em produção. Com a Implantação Contínua, todas as etapas de build, testes e implantação são executadas automaticamente, e o software é implantado em produção sem intervenção manual. Essa abordagem permite que as mudanças sejam lançadas com mais rapidez, pois não há a necessidade de intervenção manual em cada implantação

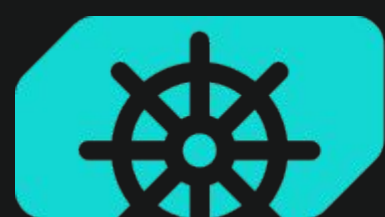
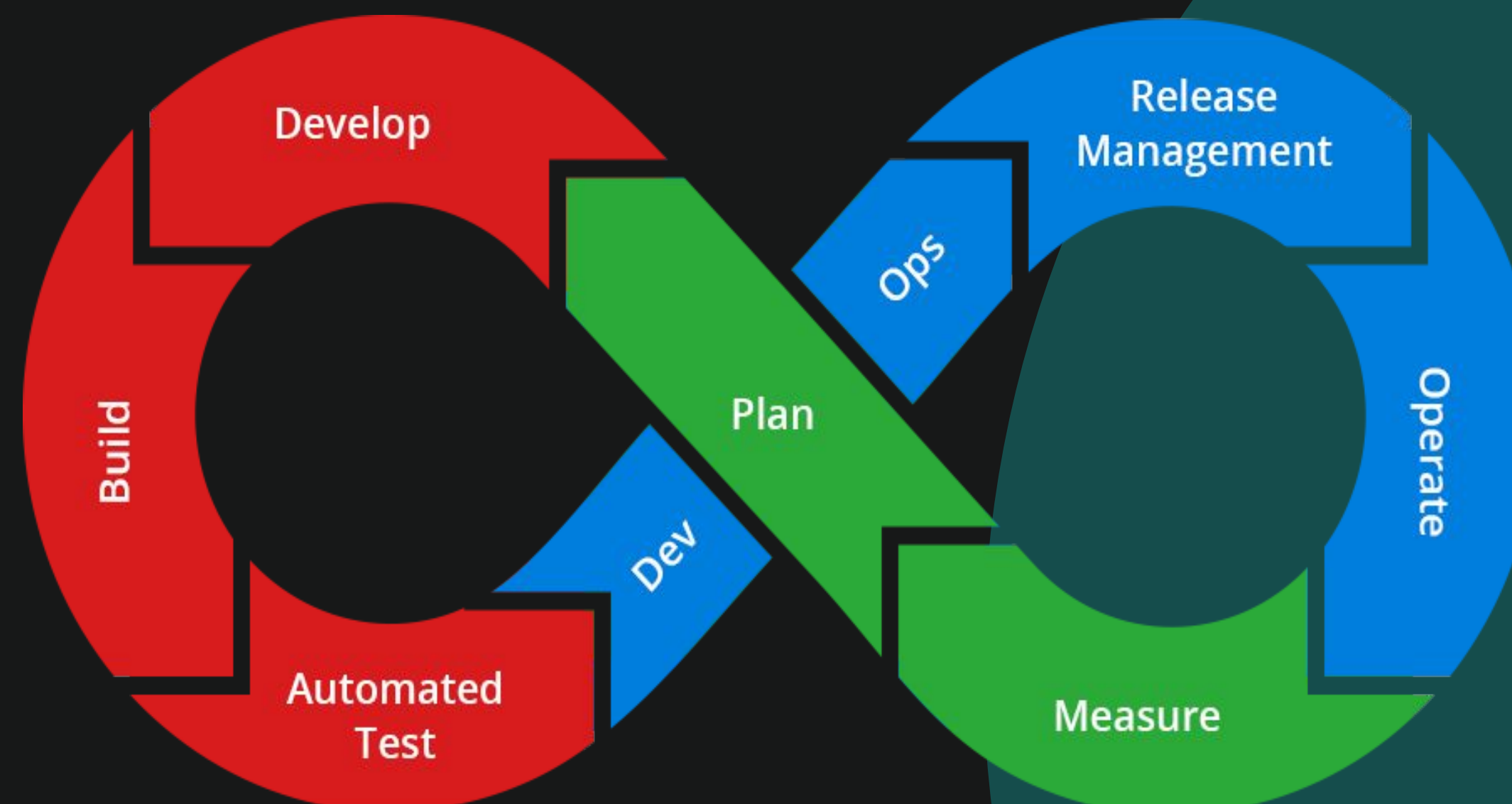


KUBERNETES  
**DAYS**

4.Linux

# Continuous Deployment (Implementacao Contínua)

- De fato, esse é o principal ponto de confusão entre os conceitos. O deployment contínuo é muito semelhante à entrega contínua, mas está à frente quando o assunto é automação. Na entrega contínua, estão prontas para deploy em produção todas as mudanças levadas para o repositório principal. Todavia, a interação humana ainda é muito importante no início desse processo.



KUBERNETES  
**DAYS**

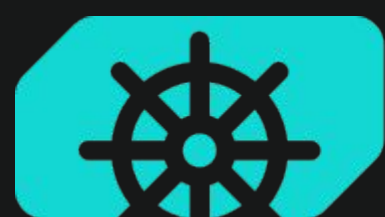
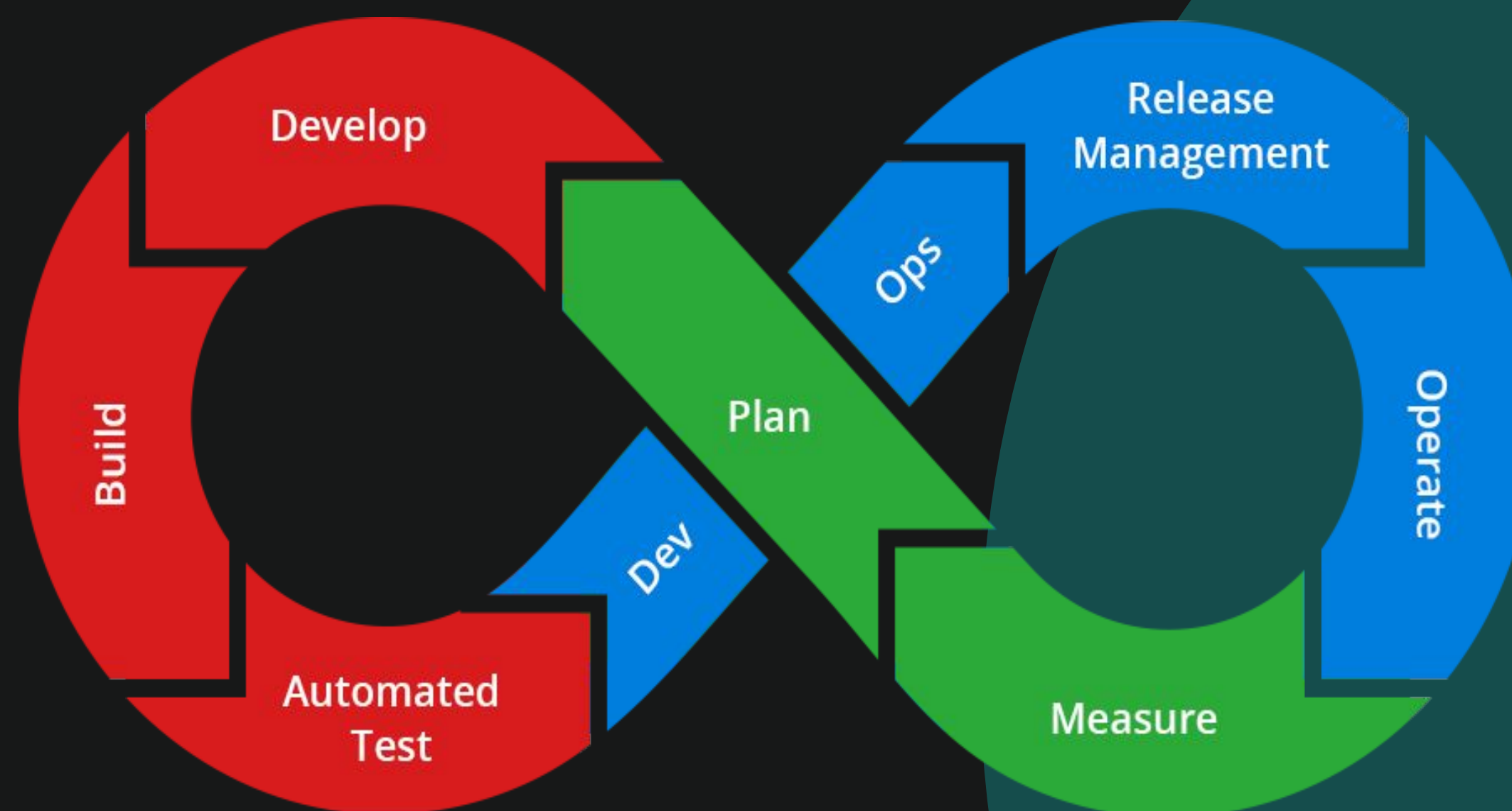
4.Linux



# O que é uma Pipeline

Um pipeline é composto por uma série de estágios, onde cada estágio executa uma tarefa específica no processo de desenvolvimento de software:

- Build
- Test
- Deploy
- Release

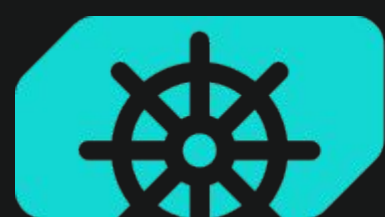


KUBERNETES  
**DAYS**

4.Linux

# SonarQube

SonarQube é uma plataforma de código aberto desenvolvida pela SonarSource para inspeção contínua da qualidade do código para realizar revisões automáticas com análise estática de código para detectar bugs e code smells em 29 linguagens de programação

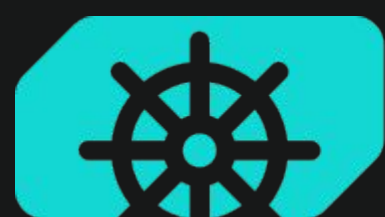


KUBERNETES  
**DAYS**

4.Linux

# Teste Unitários

Testes unitários são uma prática de teste de software em que partes individuais e isoladas de um programa, chamadas de unidades, são testadas de forma independente. O objetivo dos testes unitários é garantir que cada unidade do código (como uma função, método ou classe) funcione corretamente e produza os resultados esperados.




KUBERNETES  
**DAYS**

| 4.Linux



# Nossa aplicação


#KUBERNETESDAYS4LINUX

 KUBERNETES  
**DAYS**

**4.Linux** Open Software Specialists

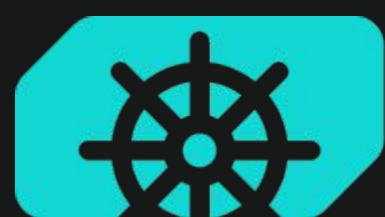
Username

Password

 Login

If you are not registry click [here](#)

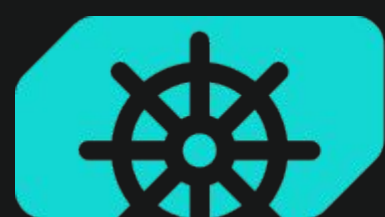
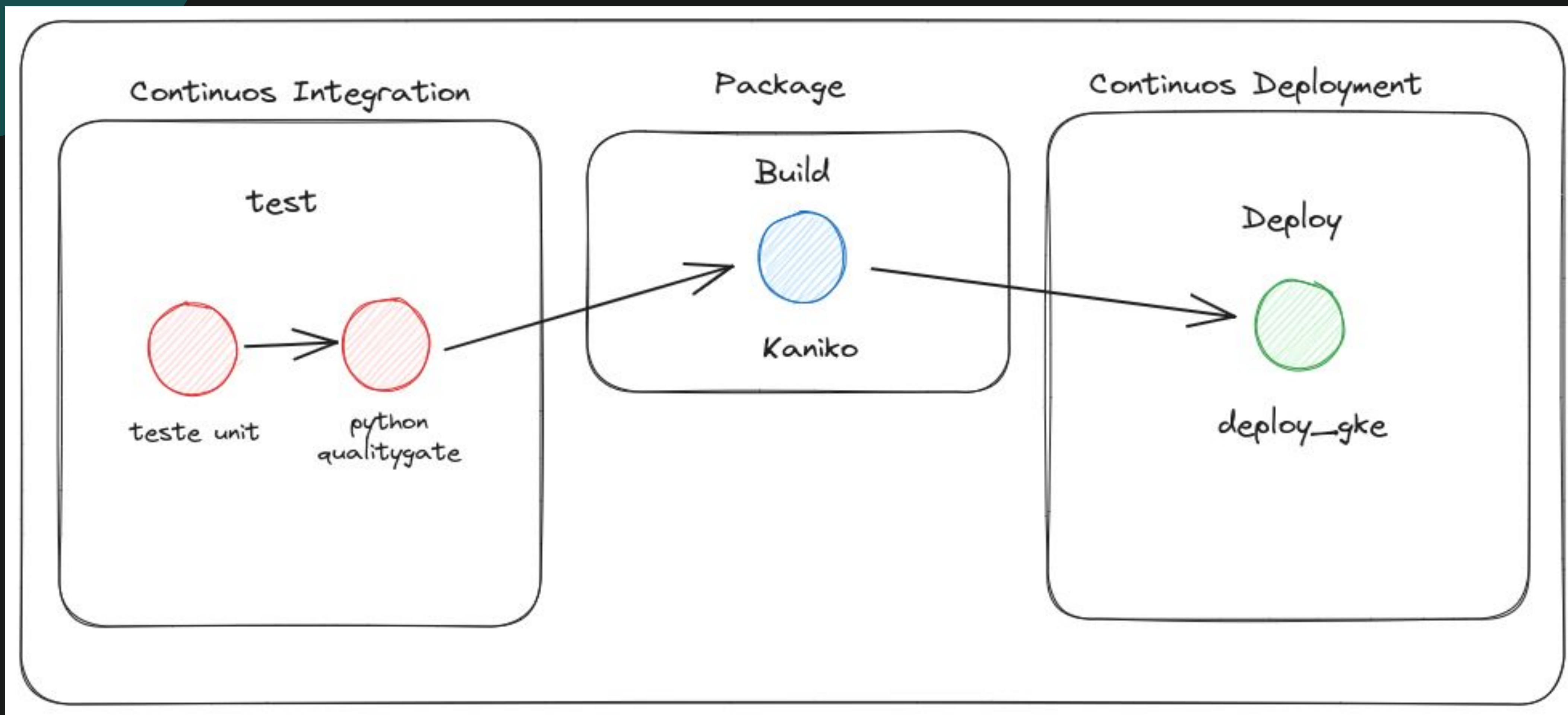
▶▶▶



KUBERNETES  
**DAYS**

**4.Linux**

# Fluxo do Deploy





# Obrigado

Por estar presente até o momento