



DevOps

DevOps:

<https://github.com/Cristianomeneses2008/devops>

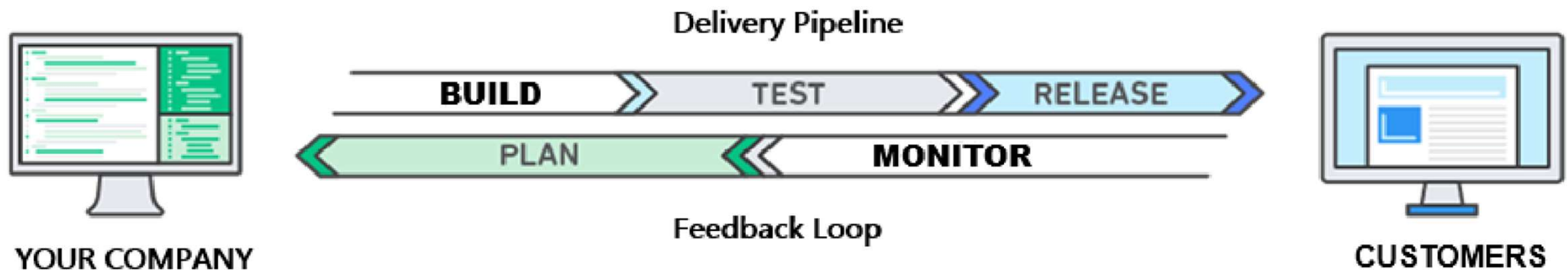
Palestra: Cristiano Meneses - 2025

O que é DevOps?

é uma metodologia que integra **desenvolvimento de software (Dev)** e **operações de TI (Ops)** para otimizar o ciclo de vida do software, desde o desenvolvimento até a produção. Em essência, o DevOps busca promover a colaboração, a comunicação e a automação entre as equipes, visando uma entrega mais rápida, confiável e eficiente de software.

Definição do modelo de DevOps

O DevOps é a combinação de filosofias culturais, práticas e ferramentas que aumentam a capacidade de uma empresa de distribuir aplicativos e serviços em alta velocidade: otimizando e aperfeiçoando produtos em um ritmo mais rápido do que o das empresas que usam processos tradicionais de desenvolvimento de software e gerenciamento de infraestrutura. Essa velocidade permite que as empresas atendam melhor aos seus clientes e compitam de modo mais eficaz no mercado.



Como funciona o DevOps

Com a implementação de um modelo de DevOps, as equipes de desenvolvimento e operações não ficam mais separadas. Às vezes, essas duas equipes são combinadas em uma só. Os engenheiros trabalham durante todo o ciclo de vida inteiro do aplicativo, da fase de desenvolvimento e testes à fase de implantação e operações, e desenvolvem várias qualificações não limitadas a uma única função.

Em alguns modelos de DevOps, as equipes de controle de qualidade e segurança também podem aumentar sua integração com o desenvolvimento, as operações e todo o ciclo de vida dos aplicativos. Quando a segurança é a prioridade de todos em uma equipe de DevOps, isso é denominado às vezes DevSecOps.

Essas equipes usam práticas para automatizar processos que historicamente sempre foram manuais e lentos. Eles usam uma pilha de tecnologia e ferramentas que os ajudam a operar e desenvolver aplicativos de modo rápido e confiável. Essas ferramentas também ajudam os engenheiros a realizar tarefas independentemente (por exemplo, implantação de código e provisionamento de infraestrutura) que normalmente exigiriam a ajuda de outras equipes, e isso aumenta ainda mais a velocidade da equipe.

Benefícios do DevOps

Velocidade

Opere em alta velocidade para que você possa trazer inovações para os seus clientes mais rapidamente, adaptar-se melhor a mercados dinâmicos e tornar-se mais eficiente na geração de resultados comerciais. O modelo de DevOps permite que as suas equipes de desenvolvedores e operações atinjam esses resultados. Por exemplo, os microsserviços e a entrega contínua permitem que as equipes assumam a responsabilidade sobre os serviços e, então, lancem atualizações para eles mais rapidamente.

Entrega rápida

Aumente a frequência e o ritmo de lançamentos para poder inovar e melhorar seu produto mais rapidamente. Quanto mais rápido você puder lançar novos recursos e corrigir erros, maior será a sua agilidade para responder às necessidades dos clientes e criar vantagem competitiva. A integração e a entrega contínuas são práticas que automatizam o processo de lançamento de software, da fase de criação à fase de implantação.



Confiabilidade

Garanta a qualidade das atualizações de aplicativos e alterações de infraestrutura para que você possa entregar com confiança em um ritmo mais rápido, sem deixar de manter uma experiência positiva para os usuários finais. Use práticas como a integração e a entrega contínuas para testar se cada uma das alterações funciona e é segura. As práticas de monitoramento e registro em log ajudam você a permanecer informado sobre a performance em tempo real.

Escala

Opere e gerencie seus processos de infraestrutura e desenvolvimento em escala. A automação e a constância ajudam você a gerenciar sistemas complexos ou dinâmicos com eficiência e risco reduzido. Por exemplo, a infraestrutura como código ajuda você a gerenciar seus ambientes de implantação, teste e produção de modo repetido e mais eficiente.



Colaboração melhorada

Crie equipes mais eficientes em um modelo cultural de DevOps, que enfatiza valores como propriedade e responsabilidade. As equipes de desenvolvedores e operações colaboram de perto, compartilham muitas responsabilidades e combinam seus fluxos de trabalho. Isso reduz ineficiências e economiza tempo (por exemplo, períodos de transferência reduzidos entre desenvolvedores e operações, escrevendo código que leva em consideração o ambiente em que está sendo executado).

Segurança

Opere rapidamente enquanto mantém o controle e preserva a conformidade. Você pode adotar o modelo de DevOps sem sacrificar a segurança usando políticas de conformidade automáticas, controles minuciosos e técnicas de gerenciamento de configuração. Por exemplo, usando a infraestrutura e a política como código, você pode definir e acompanhar a conformidade em escala.



Por que o DevOps é importante

O software e a Internet transformaram o mundo e seus mercados, do comércio ao entretenimento e ao sistema bancário. O software já não apenas sustenta uma atividade empresarial, na verdade ele tornou-se um componente integral de cada parte de uma empresa. As empresas interagem com seus clientes por meio de softwares disponibilizados como serviços ou aplicativos online, e em todos os tipos de dispositivos. Elas também usam o software para aumentar a eficiência operacional e transformar cada parte da cadeia de valor, como logística, comunicação e operações. Similarmente a como essas empresas de bens materiais transformaram como elas projetavam, criavam e disponibilizavam produtos usando a automação industrial durante o século 20, as empresas de hoje devem transformar como elas criam e disponibilizam software.



Como adotar um modelo de DevOps

Filosofia cultural do DevOps

A transição para o DevOps exige uma mudança de cultura e mentalidade. Em seu nível mais simples, o objetivo do DevOps é remover as barreiras entre duas equipes tradicionalmente separadas em silos: desenvolvimento e operações. Em algumas empresas, podem até não existir equipes de desenvolvimento e operações separadas, os engenheiros se encarregam de ambas. Com o DevOps, as duas equipes trabalham juntas para otimizar a produtividade dos desenvolvedores e a confiabilidade das operações. Elas se esforçam para manter a comunicação frequente, aumentar a eficiência e melhorar a qualidade dos serviços disponibilizados aos clientes. As equipes assumem a responsabilidade total sobre os serviços, geralmente indo além do escopo tradicional de seus cargos e títulos definidos, pensando sobre as necessidades do cliente final e como eles podem contribuir para resolver essas necessidades. As equipes de controle de qualidade e segurança também podem tornar-se altamente integradas a essas equipes. As empresas usando um modelo de DevOps, independentemente de sua estrutura organizacional, têm equipes que supervisionam o ciclo de vida completo de desenvolvimento e infraestrutura como parte das suas responsabilidades.

Explicação das práticas de DevOps

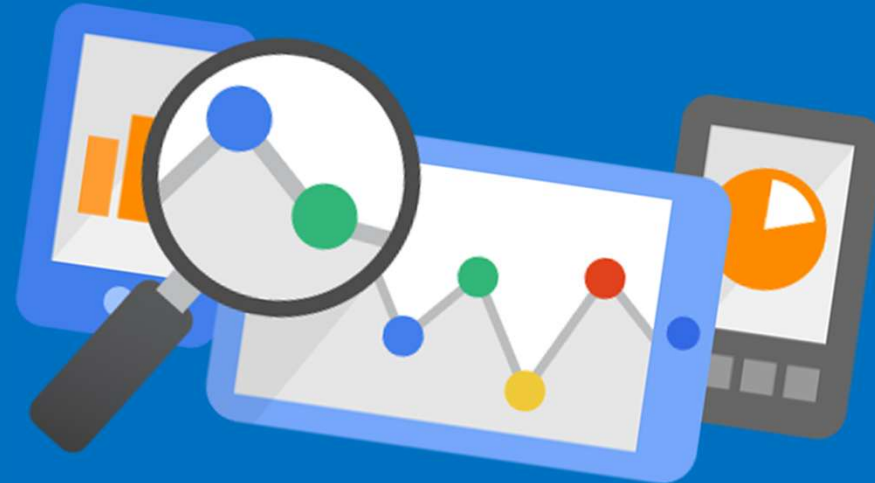
Existem algumas práticas essenciais que ajudam as empresas a inovar mais rapidamente por meio da automação e da simplificação dos processos de desenvolvimento de software e gerenciamento de infraestrutura. A maioria dessas práticas são realizadas com as ferramentas certas.

Uma prática fundamental é a execução de atualizações muito frequentes, porém pequenas. É assim que as empresas inovam mais rapidamente para os seus clientes. Geralmente, essas atualizações são mais incrementais por natureza do que as atualizações ocasionais executadas de acordo com as práticas de lançamento tradicionais. As atualizações frequentes, porém pequenas, diminuem o risco de cada implantação. Essas atualizações ajudam as equipes a analisar erros mais rapidamente, pois elas conseguem identificar a última implantação que provocou o erro. Embora o ritmo e o tamanho das atualizações sejam variáveis, as empresas que usam o modelo de DevOps implantam atualizações com uma frequência muito maior do que as que utilizam práticas de desenvolvimento de software tradicionais.

As empresas também podem usar uma arquitetura de microsserviços para tornar seus aplicativos mais flexíveis e viabilizar inovações mais rapidamente. A arquitetura de microsserviços desacopla sistemas grandes e complexos e os transforma em projetos simples e independentes. Os aplicativos são divididos em vários componentes individuais (serviços). Cada um desses serviços abrange uma finalidade ou função única, além de ser operado independentemente dos serviços de mesmo nível e do aplicativo como um todo. A arquitetura reduz a sobrecarga gerada pela coordenação da atualização de aplicativos e, quando cada serviço é combinado com equipes pequenas e ágeis que assumem a responsabilidade sobre cada serviço, as empresas conseguem operar mais rapidamente.

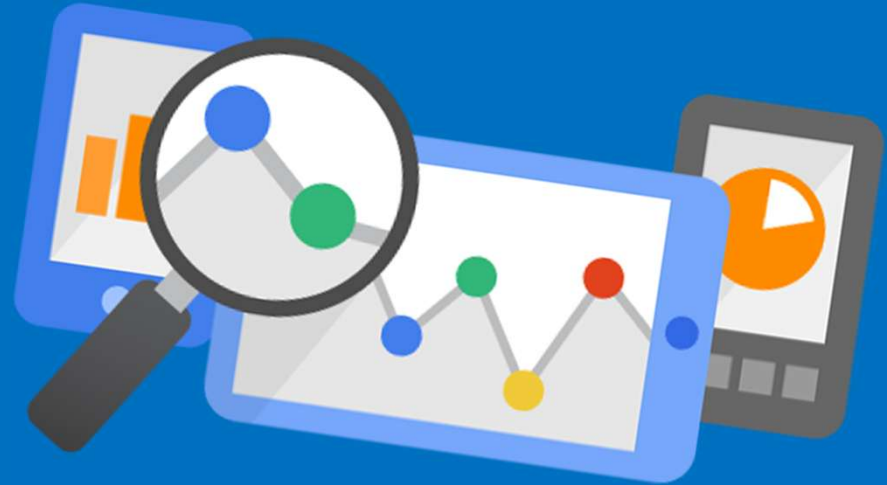
No entanto, a combinação de microsserviços e uma maior frequência de lançamentos resulta em um número significativamente maior de implantações que podem apresentar desafios operacionais. Portanto, as práticas de DevOps, como a integração e a entrega contínuas, resolvem esses problemas e permitem que as empresas mostrem resultados rapidamente, de modo seguro e confiável. As práticas de automação da infraestrutura, como a infraestrutura como código e o gerenciamento de configuração, ajudam a manter recursos computacionais elásticos e responsivos para as alterações frequentes. Além disso, o uso do monitoramento e do registro em log ajudam os engenheiros a rastrear o desempenho de aplicativos e da infraestrutura para que possam reagir rapidamente quando surgirem problemas.

Juntas, essas práticas ajudam as empresas a disponibilizar atualizações mais rápidas e confiáveis aos seus clientes. Aqui está uma visão geral sobre práticas importantes de DevOps.



Práticas de DevOps:

- Integração contínua;
- Distribuição contínua;
- Microserviços;
- Infraestrutura como código;
- Monitoramento e registro em log;
- Comunicação e colaboração;
- Ferramentas de DevOps.



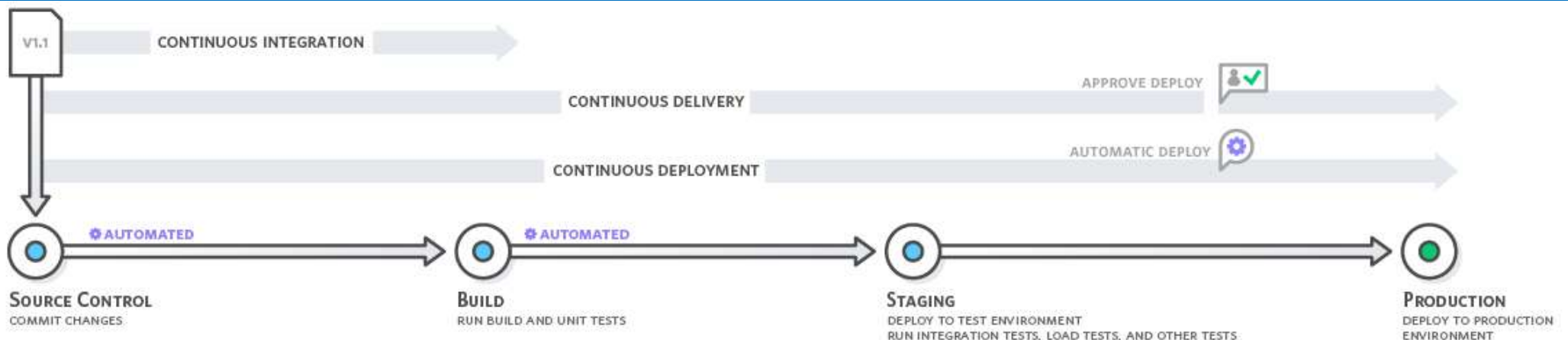
Integração contínua

A integração contínua é uma prática de desenvolvimento de software em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do software e reduzir o tempo necessário para validar e lançar novas atualizações de software.



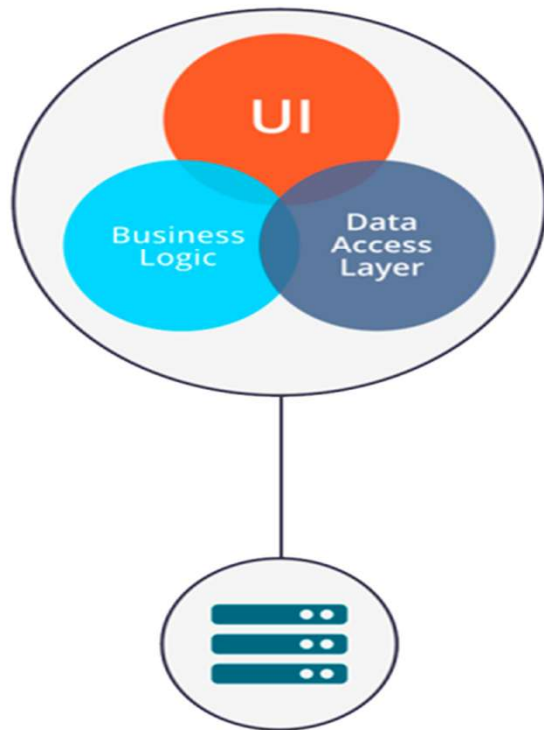
Distribuição contínua

A entrega contínua é uma prática de desenvolvimento de software em que alterações de código são criadas, testadas e preparadas automaticamente para liberação para produção. Ela expande com base na integração contínua, pela implantação de todas as alterações de código em um ambiente de teste e/ou ambiente de produção, após o estágio de criação. Quando a integração contínua for implementada adequadamente, os desenvolvedores sempre terão um artefato de criação pronto para ser implantado, e que passou por um processo de teste padronizado.

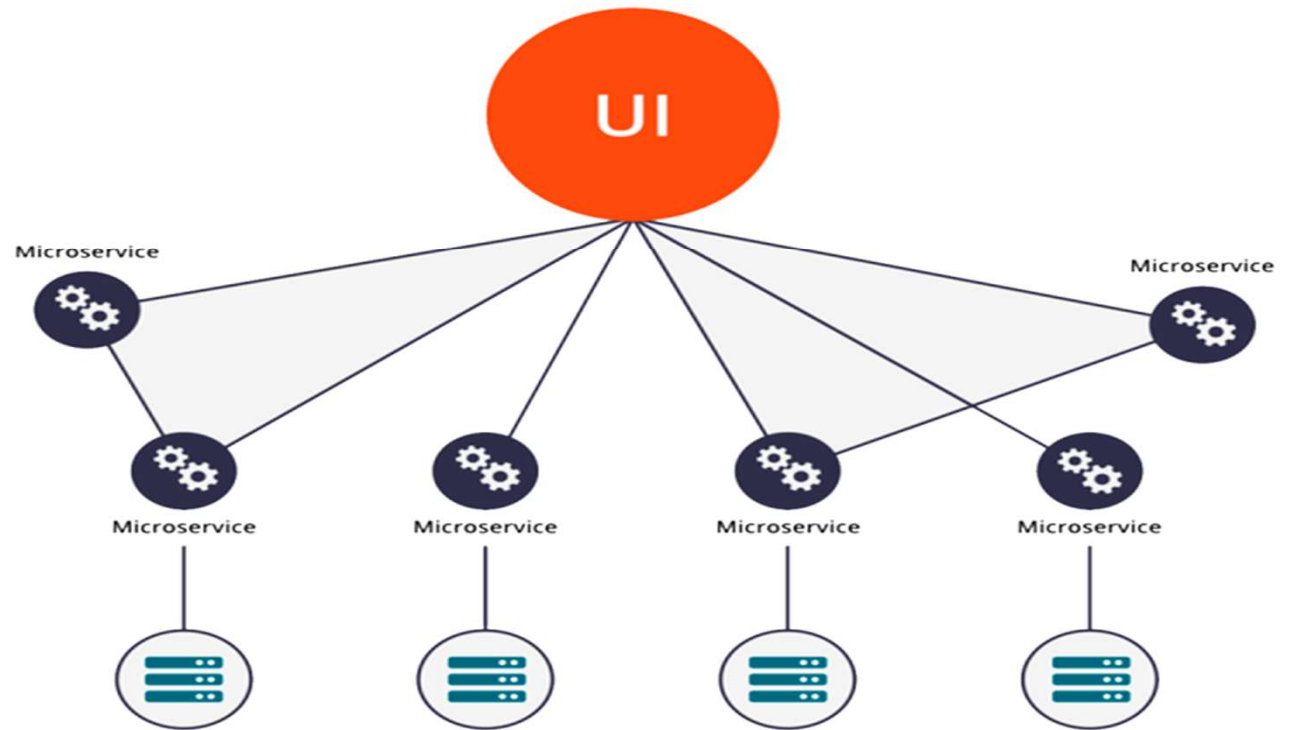


Microserviços

A arquitetura de microserviços é uma abordagem de projeto para a criação de um aplicativo único como um conjunto de pequenos serviços. Cada serviço é executado em seu próprio processo e se comunica com outros serviços por meio de uma interface bem definida usando um mecanismo leve, geralmente uma interface de programação de aplicativo (API) baseada em HTTP. Os microserviços são criados em torno dos recursos empresariais, e cada serviço tem uma finalidade única. Você pode usar estruturas ou linguagens de programação diferentes para gravar microserviços e implantá-los independentemente como um único serviço ou um grupo de serviços.



Arquitetura Monolítica



Arquitetura de Microserviços

Infraestrutura como código

é uma prática em que a infraestrutura é provisionada e gerenciada usando técnicas de desenvolvimento de código e software, como controle de versão e integração contínua. O modelo controlado por API da nuvem permite que desenvolvedores e administradores de sistema interajam com a infraestrutura de modo programático e em escala, em vez de precisarem instalar e configurar manualmente os recursos. Portanto, os engenheiros podem dialogar com a infraestrutura usando ferramentas baseadas em código e tratá-la de modo similar ao código do aplicativo. Como são definidos por código, infraestrutura e servidores podem ser implantados rapidamente usando padrões normativos, atualizados com os patches e as versões mais recentes ou duplicados de modos repetidos.

INFRASTRUCTURE AS CODE

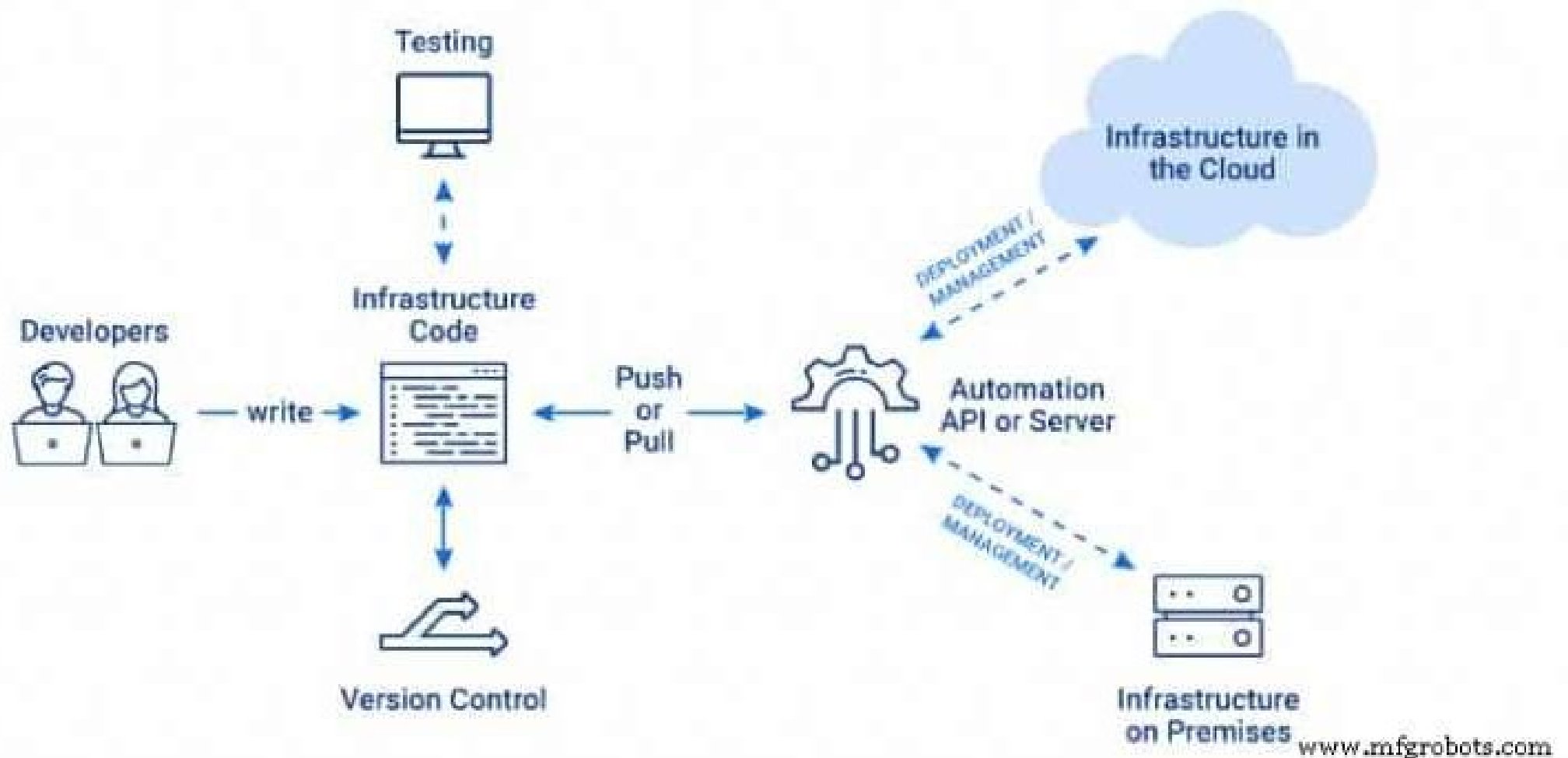
INFRASTRUCTURE
AUTOMATION

CONFIGURATION
MANAGEMENT

VERSION
CONTROL

AUTOMATED
TESTING

DEPLOYMENT
AUTOMATION



Infraestrutura como código

1. Gerenciamento de configuração

Os desenvolvedores e os administradores de sistema usam código para automatizar o sistema operacional e a configuração do host, as tarefas operacionais e muito mais. O uso do código torna as alterações de configuração repetidas e padronizadas. Isso isenta os desenvolvedores e os administradores de sistemas de ter que configurar manualmente sistemas operacionais, aplicativos de sistemas ou software do servidor.

Infraestrutura como código

2. Política como código

Com a infraestrutura e a sua configuração codificada com a nuvem, as empresas podem monitorar e aplicar a conformidade de modo dinâmico e em escala. Portanto, a infraestrutura, que é descrita pelo código, pode ser acompanhada, validada e reconfigurada de modo automático. Isso facilita para as empresas administrarem as alterações de recursos e garantirem que as medidas de segurança foram aplicadas adequadamente de modo distribuído (por exemplo, segurança de informações ou conformidade com PCI-DSS ou HIPAA). Isso permite que as equipes dentro de uma empresa operem em uma velocidade maior, pois recursos fora de conformidade podem ser sinalizados automaticamente para serem investigados em mais detalhes ou, até mesmo, ter sua conformidade restabelecida de modo automático.

Faixa Salarial:

- Média: R\$ 6.473,00 (em média).
- Júnior: R\$ 4.046,00 (mediana).
- Pleno: R\$ 7.000,00 (média).
- Sênior: R\$ 12.049,50 (mediana).
- Especialista: R\$ 15.500,00 (remuneração total mensal estimada).

No Brasil, um profissional DevOps pode ganhar, em média, R\$ 8.100 por mês, mas a remuneração varia bastante dependendo da experiência, nível de senioridade, tamanho da empresa e localização.

Ferramentas de DevOps

RANCHER vs OPENSIFT



- ✓ INSTALAÇÃO: 20 MINUTOS a 2 HORAS
- ✓ UPGRADES INCREMENTAIS E CONSISTENTES
- ✓ SUPORTE PLATAFORMA + CATÁLOGO
- ✓ PADRÃO DA INDÚSTRIA CNCF
- ✓ CUSTO: 1X



- INSTALAÇÃO: 3 DIAS A ALGUMAS SEMANAS
- REPORTE DE UPGRADES COM FALHAS E ROLLBACKS
- SUPORTE SOMENTE PLATAFORMA
- PRIORIZA PRÁTICAS NÃO-CNCF
- CUSTO: 5X-10X



FIM

Fontes de Pesquisa:

<https://aws.amazon.com/pt/devops/what-is-devops/>