



# DevOps e a melhoria no fluxo de desenvolvimento

DevOps Fundamentals



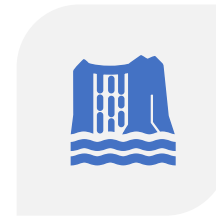
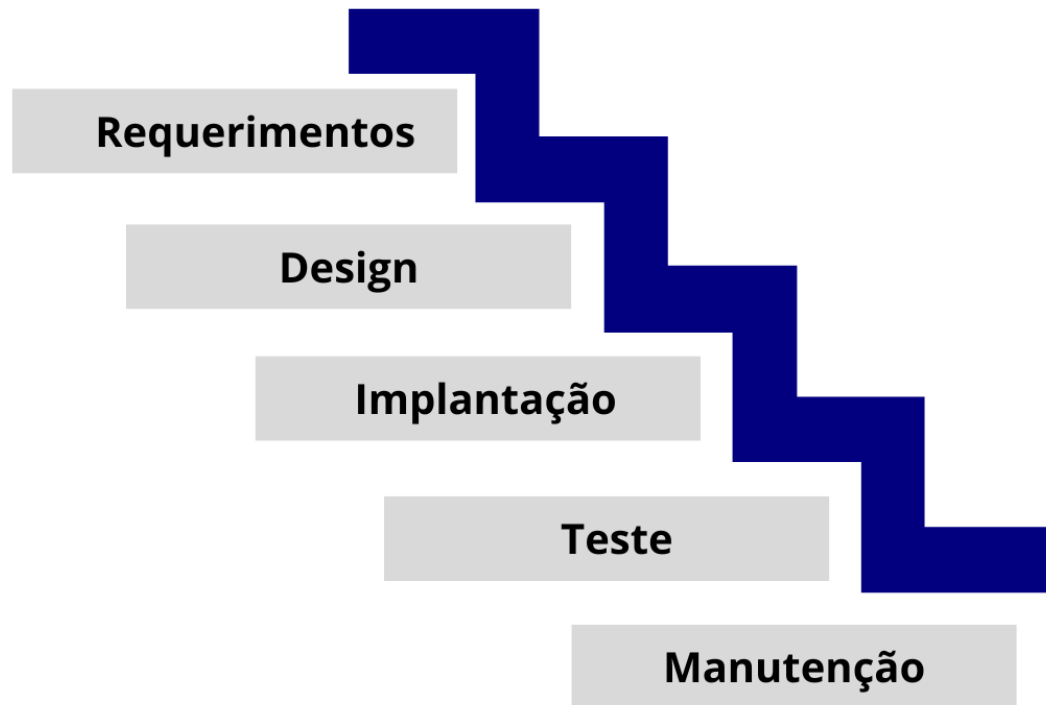


# DevOps e a melhoria no fluxo de desenvolvimento

Atualmente temos diversas metodologias de desenvolvimento de software e que foram criadas para ajuda a melhorar o fluxo de desenvolvimento. Algumas delas são:

- Modelo Cascata (Wartefall)
- Scrum
- Kanban
- TDD – Test-Driven-Development
- DevOps

# DevOps e a melhoria no fluxo de desenvolvimento



**MODELO CASCATA  
(WATERFALL)**



ESTE É UM DOS MODELOS DE DESENVOLVIMENTO DE SOFTWARE MAIS ANTIGO E MAIS UTILIZADO. ESTA METODOLOGIA CONSISTE EM UMA SEQUÊNCIA DE FASES, ONDE CADA FASE É COMPLETADA ANTES DE INICIAR A PRÓXIMA. AS FASES INCLUEM ANÁLISE DE REQUISITOS, DESIGN, IMPLEMENTAÇÃO TESTES E MANUTENÇÃO. EMBORA ESTE MODELO SEJA FÁCIL DE ENTENDER E IMPLEMENTAR, ELE PODE SER INFLEXÍVEL EM RELAÇÃO ÀS MUDANÇAS NO MEIO DO PROCESSO.

# DevOps e a melhoria no fluxo de desenvolvimento

## Scrum

O Scrum é um framework ágil, simples e leve ou poderíamos dizer que é uma metodologia ágil de desenvolvimento de software, muito utilizada!

O Scrum divide o desenvolvimento em iterações curtas, conhecidas como sprints, que geralmente duram de 2 a 4 semanas. Ele também incrementa valores para entregar valor com frequência e com isto reduzir os riscos do projeto.

Ele também é adaptável a mudança e ajuda a manter a equipe focada e produtiva.



# DevOps e a melhoria no fluxo de desenvolvimento



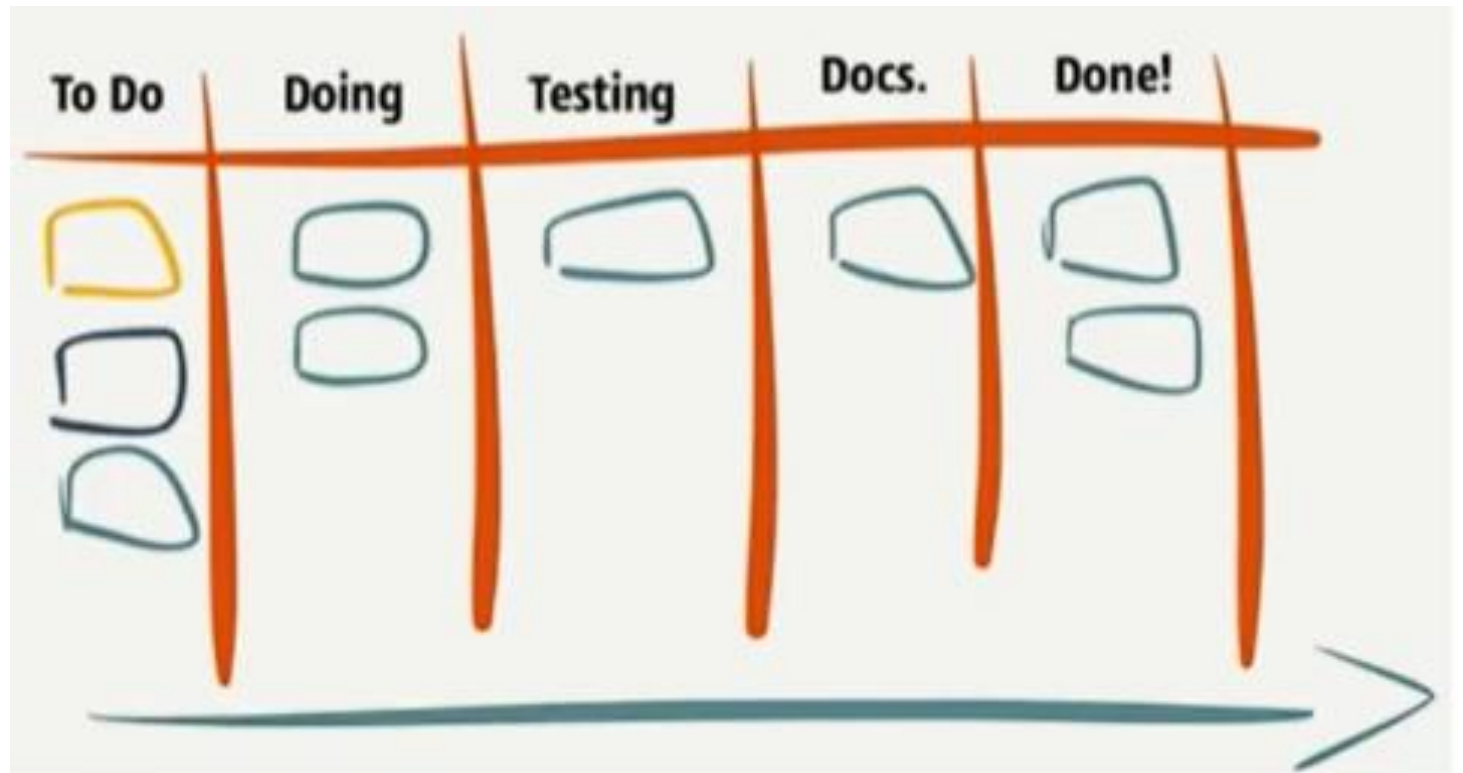
## Kanban

É uma metodologia ágil que se concentra na entrega contínua de pequenos incrementos de software. Ou seja, nada mais é do que um quadro com todas as tarefas que permite visualizar onde o trabalho está fluindo bem e onde está na fila ou interrompido.

A palavra **kanban** vem do japonês e significa listas ou cartões.

Esta metodologia foi inventada nas linhas de produção da **Toyota**.

# DevOps e a melhoria no fluxo de desenvolvimento



## Kanban

No kanban nós temos 3 listas **TO DO**, **DOING** e **DONE**.

Em **TO DO** - colocamos as tarefas que estão por fazer.

Em **DOING** - colocamos as tarefas que estamos fazendo naquele exato momento.

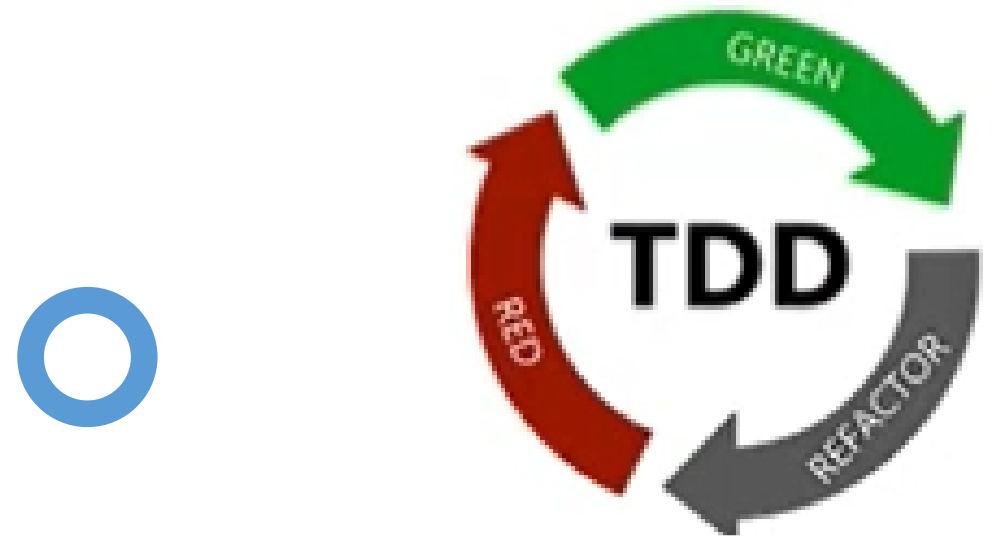
Em **DONE** - colocamos as tarefas que foram concluídas.

# DevOps e a melhoria no fluxo de desenvolvimento

## TDD – Test-Driven-Development

Ele não é uma forma de escrever testes e na verdade ele é uma forma de desenvolver código, ou seja, eu crio meu teste e em cima deste teste eu desenvolvo o código somente para passar naquele teste.

O objetivo do TDD é garantir que o código seja testado continuamente durante o desenvolvimento.

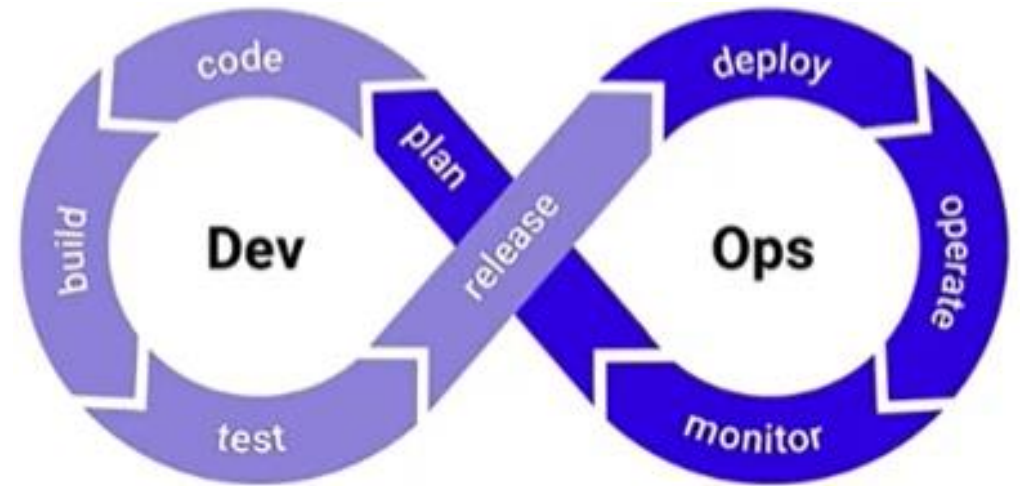


# DevOps e a melhoria no fluxo de desenvolvimento

## DevOps

O DevOps é uma abordagem que se concentra na colaboração entre as equipes de desenvolvimento e operações de TI.

O objetivo do DevOps é integrar o desenvolvimento e operação em um processo contínuo e automatizado, o que ajuda a reduzir o tempo de lançamento de software e melhorar a qualidade do software.





# DevOps e a melhoria no fluxo de desenvolvimento

Dentre todas o DevOps é uma metodologia de colaboração entre equipes de desenvolvimento e operações. E tem como objetivo a entrega rápida e confiável de software.

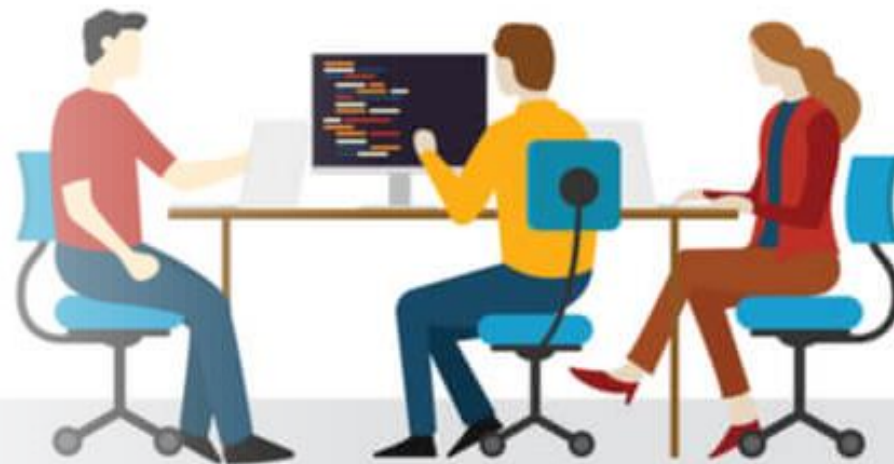
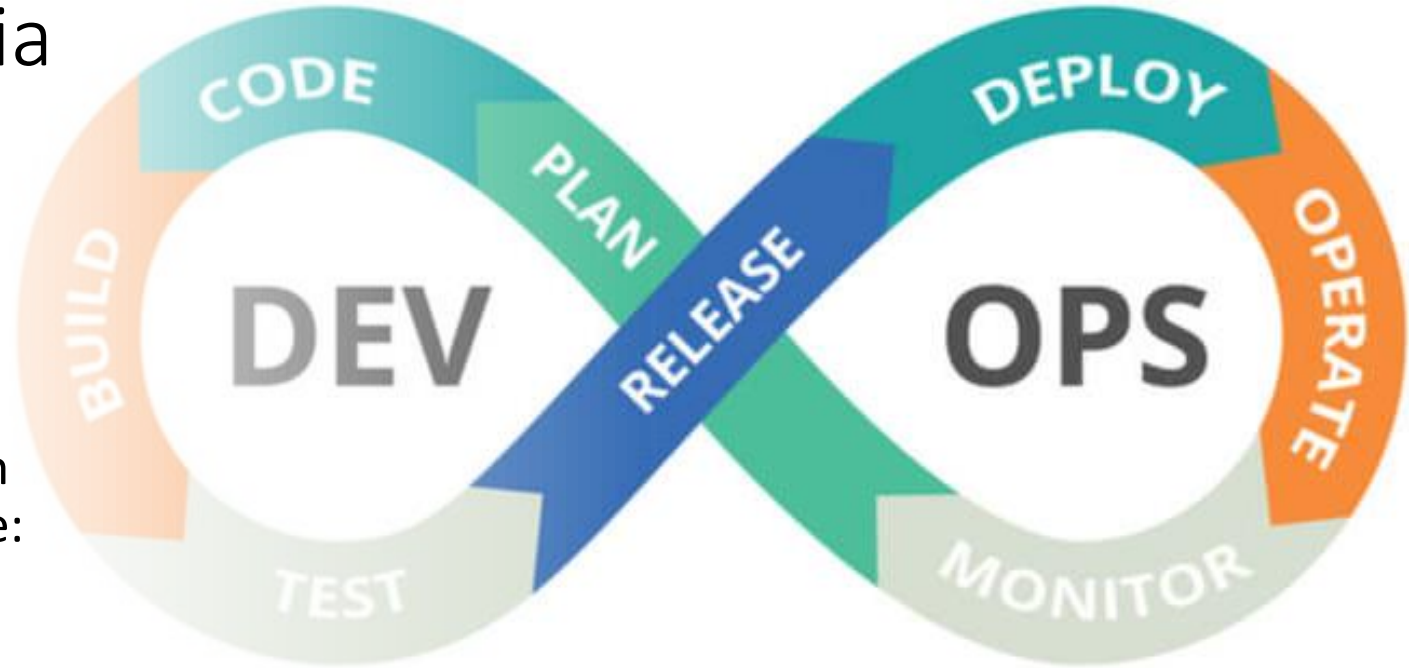
Os DevOps aborda mais a automação de processos, a colaboração entre as equipes e a monitoração constante do software em produção. Por isto, a sua adoção tem crescido e vem se tornando mais adequado para equipes que desejam melhorar a eficiência, a qualidade e agilidade no processo de desenvolvimento e entrega de software.



# DevOps e a melhoria no fluxo de desenvolvimento

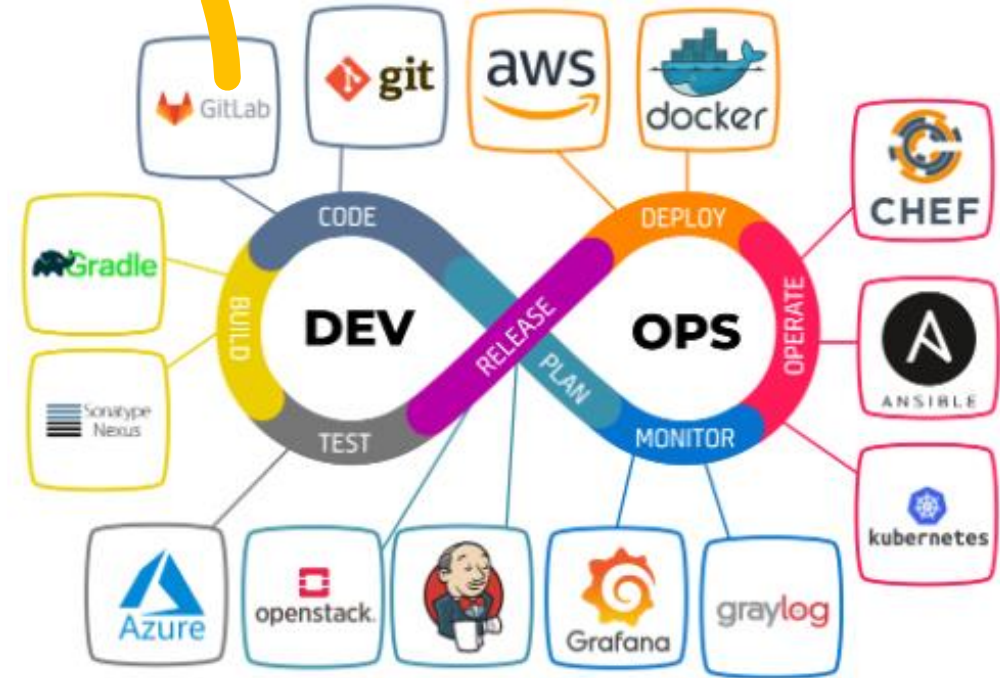
Vejamos a seguir, algumas das maneiras que os DevOps ajudam no desenvolvimento de software:

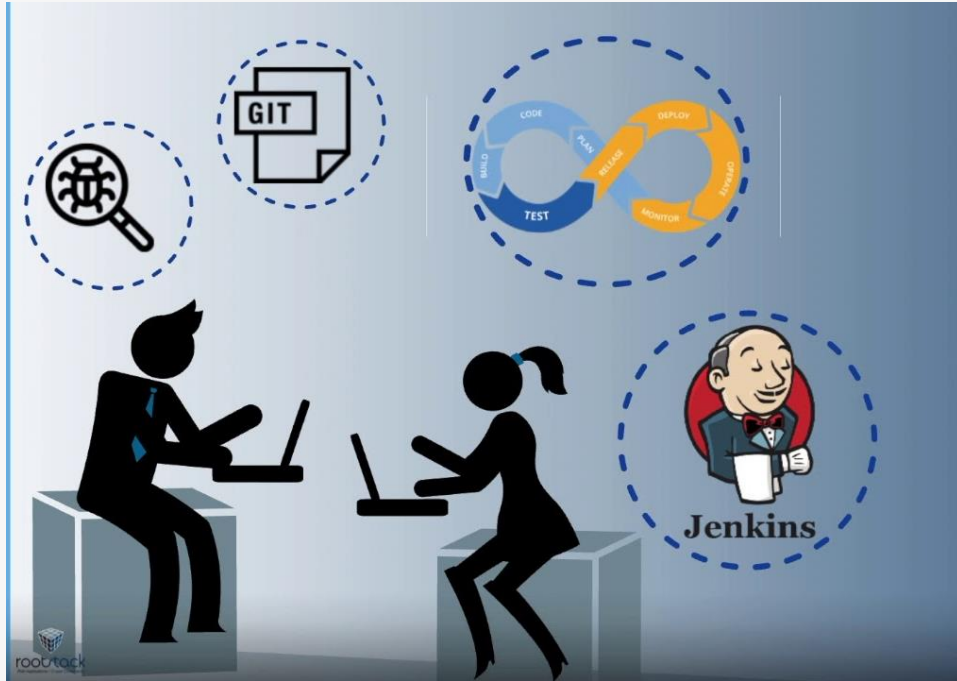
1. A colaboração entre equipes: O DevOps incentiva a colaboração e a comunicação constante entre as equipes de desenvolvimento e operações, permitindo que elas trabalhem juntas de forma mais eficiente e eficaz.



# DevOps e a melhoria no fluxo de desenvolvimento

2. Automatização de processos: o DevOps enfatiza a automação de processos, desde a integração contínua e entrega contínua (CI/CD), passando por testes, até a implantação e monitoramento de software. Ajudando a reduzir o tempo de lançamento no mercado e minimiza a possibilidade de erros humanos.





3. Melhoria contínua: o DevOps promove a cultura de melhoria contínua, em que a equipe de desenvolvimento está sempre procurando maneiras de melhorar o processo de desenvolvimento de software, a qualidade do software e a experiência do usuário.

4. Agilidade: o DevOps ajuda as equipes a se tornarem mais ágeis no desenvolvimento de software, permitindo que elas respondam rapidamente às mudanças nas necessidades do cliente e do mercado.

---

## DevOps e a melhoria no fluxo de desenvolvimento




## DevOps e a melhoria no fluxo de desenvolvimento

5. Monitoramento constante: o DevOps promove a monitoração constante do software em produção, permitindo que a equipe do desenvolvimento responda rapidamente a quaisquer problemas e reduza o tempo de inatividade do sistema.

No geral, o DevOps é uma abordagem que ajuda as equipes de desenvolvimento de software a melhorar a eficiência, a qualidade e a agilidade no processo de desenvolvimento e entrega de software.





Equipe de  
Desenvolvimento X  
Equipe de  
Infraestrutura



---

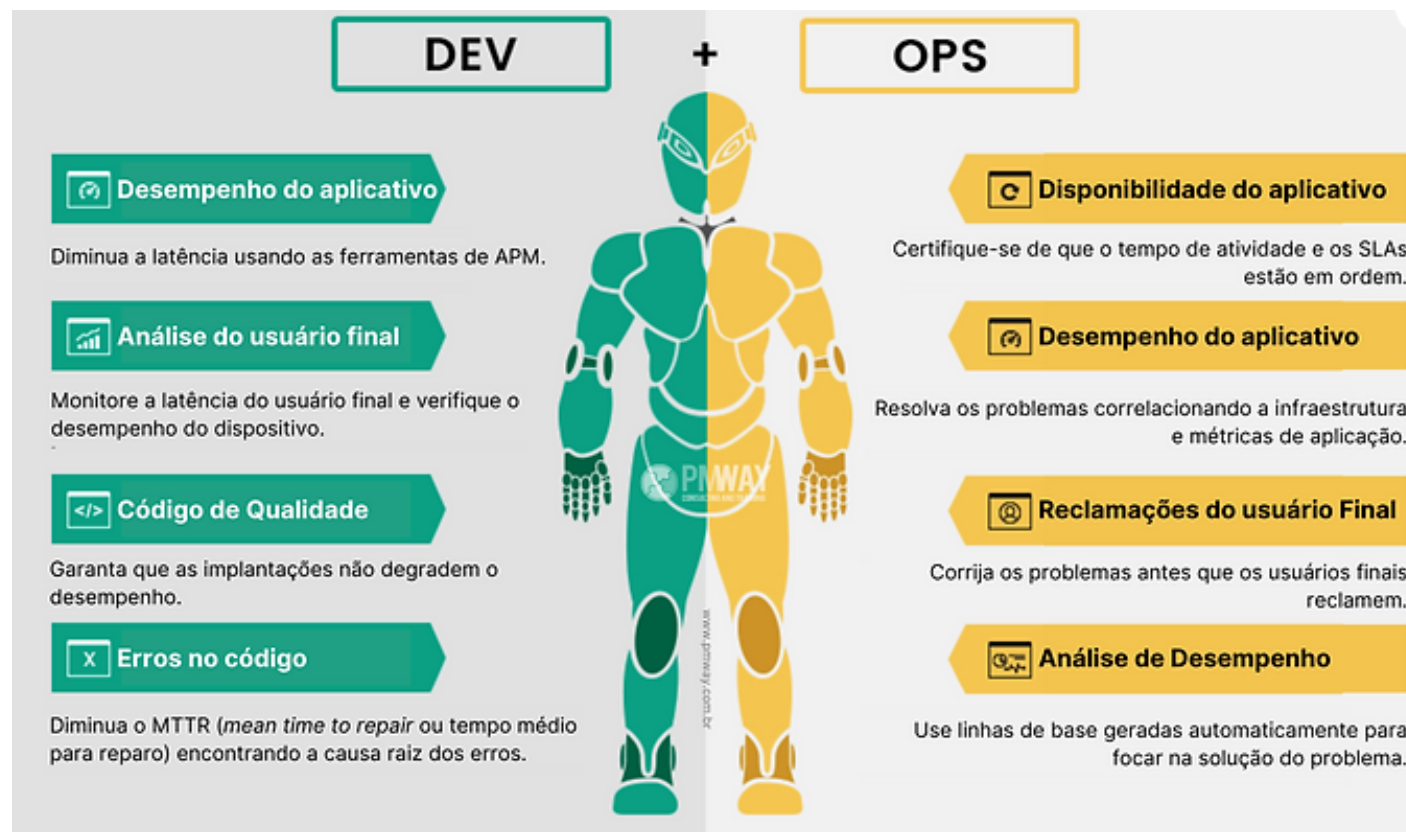
## Equipe de Desenvolvimento X Equipe de Infraestrutura

Durante muitos anos, a equipe de desenvolvimento e a equipe de operações têm enfrentado dificuldades de relacionamento.

Os desenvolvedores e administradores de sistema possuem visões diferentes, mas ambos concordam que os clientes do setor corporativo frequentemente exigem mudanças rápidas, novos recursos e serviços atualizados para gerar receitas.

Ao mesmo tempo, esses usuários também desejam um sistema estável e confiável, sem interrupções ou falhas, o que gera um conflito.

# Equipe de Desenvolvimento X Equipe de Infraestrutura



Encontrar um equilíbrio entre essas demandas tem sido um grande desafio para ambas as equipes.

Equipes que não trabalham em sinergia tendem a ter um rendimento ruim e apresentar muitas falhas durante um projeto de software. Isto acontece porque a falta de sinergia prejudica a comunicação e a preocupação entre uma equipe em ajudar a outra.

Quanto mais próximas as equipes trabalharem, maior tende a ser a produtividade em prol de um único objetivo, que é de entregar o projeto com a qualidade esperada, dentro dos prazos e custos estimados.



# Equipe de Desenvolvimento X Equipe de Infraestrutura



Geralmente os **profissionais de infraestrutura** (redes, servidores, segurança e etc.) escolhem a área de operações para “correr” da programação.



E os **profissionais de desenvolvimento** escolhem a área por não gostar de mexer com periféricos, redes, servidores e etc. Estes profissionais focam mais em código.



Observaram como são áreas e profissionais distintos?



“Não mexa no meu código que eu não mexo no seu servidor!”



Equipe de  
Desenvolvimento X  
Equipe de Infraestrutura

Quando DevOps houve uma aproximação desses profissionais  
e com isto houve o que ninguém esperava

# Equipe de Desenvolvimento X Equipe de Infraestrutura

O **profissional Dev** - precisou aprender um pouco sobre as operações e entender como funciona os servidores, redes, segurança e etc. Por que com o DevOps e ele irá automatizar ou criar a IAC (infraestrutura como código) e com isto precisará saber como as coisas se integram.



O **profissional Ops** - precisou entender um pouco sobre desenvolvimento e entender como funciona a IDE, como compilar o código, fazer testes unitários e como fazer um deploy nos servidores.



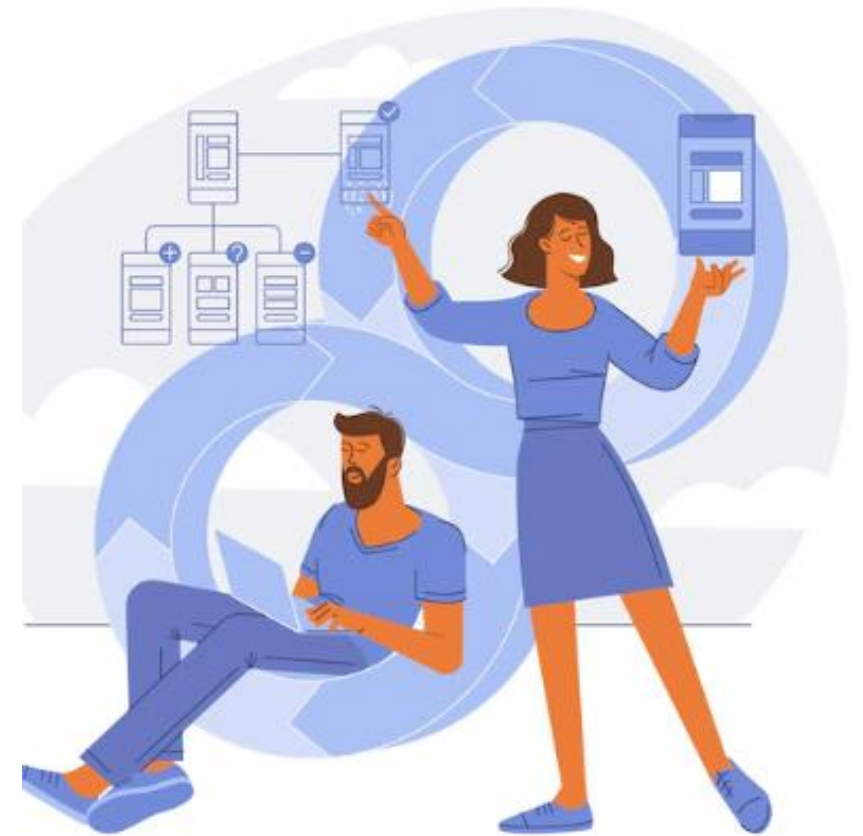
# Equipe de Desenvolvimento X Equipe de Infraestrutura

---

O DevOps não só une times de áreas específicas de TI, mas também, promove uma ruptura nas barreiras existentes com o negócio.

Promove, ainda, uma atuação sinérgica entre quem desenvolve e permite uma maior comunicação com quem define as funcionalidades para otimizar os negócios, já que a visão passa a ser fim a fim e o propósito de entregar valor ao cliente.

O DevOps também traz muitos benefícios que auxiliam times com veremos ver a seguir...



# Equipe de Desenvolvimento X Equipe de Infraestrutura



## Benefícios do DevOps:

1. Entrega ágil de software: permite A Entrega rápida de software de alta qualidade, atendendo as necessidades dos clientes e as mudanças no mercado de forma mais ágil.
2. Redução de erros e problemas operacionais: ao automatizar processos e monitorar constantemente, o DevOps ajuda a reduzir erros humanos e problemas operacionais, garantindo maior eficiência.

# Equipe de Desenvolvimento X Equipe de Infraestrutura

3. Melhoria na colaboração: promove a cultura de colaboração entre as equipes de desenvolvimento e operações, melhorando a comunicação e a troca de conhecimentos.

4. Aumento da qualidade do software: por meio de práticas como testes automatizados e integração contínua, o DevOps ajuda a identificar e corrigir problemas precocemente, garantindo um software de melhor qualidade.




# Equipe de Desenvolvimento X Equipe de Infraestrutura



5. Aumento da satisfação do cliente: com as entregas mais rápidas e de melhor qualidade, o DevOps pode contribuir para aumentar a satisfação do cliente, o que é fundamental para o sucesso de uma empresa.

Resumindo, o DevOps traz inúmeros benefícios às empresas que o adotam, desde a redução de custos e aumento da eficiência até a melhoria da qualidade do software e a satisfação do cliente.



A 3D rendering of a puzzle with one red piece standing out among many white pieces. The puzzle is set against a background with a blue-to-orange gradient. The red piece is positioned in the center-right of the frame, and the white pieces are arranged around it, some of which are slightly offset, creating a sense of depth and focus on the red piece.

# Estrutura Organizacional para DevOps



# Estrutura Organizacional para DevOps

Este é um ponto muito interessante, porque existem muitas informações sobre DevOps na internet e ao mesmo tempo com estas informações não conseguimos entender o que é o DevOps e de fato.

**Cultura:** há quem diga e defenda que é uma cultura de desenvolvimento de software e que as empresas adotam para melhorar o fluxo de desenvolvimento de software.

**Profissional:** há outros que dizem que o DevOps é o profissional que fica transitando entre as áreas de desenvolvimento infraestrutura



# Estrutura Organizacional para DevOps

Todas as duas abordagens estão corretas, porém e há quem **concorde** e há quem **não concorde**, mas uma coisa é certa: **quem manda é o mercado!**

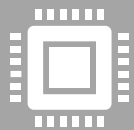


# Estrutura Organizacional para DevOps

---



Hoje vemos mais a procura de profissionais com título DevOps em seu LinkedIn ou cv do que alguém que tenha conhecimento geral da cultura em si. Ter o conhecimento, a base e o fundamento é bom, porém isto não é regra para o mercado.



Implementar o DevOps pode ser um desafio para algumas empresas. É importante que as empresas avaliam cuidadosamente as suas necessidades de negócios, cultura organizacional e capacidades de TI antes de adotar o devops.

# Estrutura Organizacional para DevOps

Veremos a seguir algumas maneiras de como as empresas pensam com relação ao DevOps.

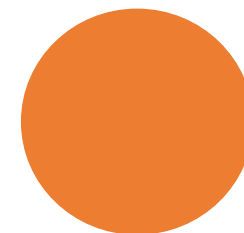


# Estrutura Organizacional para DevOps

A implementação do DevOps pode variar de empresa para empresa, mas algumas práticas comuns que podem ser aplicadas.

A **Target**, por exemplo, criou uma equipe específica de DevOps para lidar com o gerenciamento de seus aplicativos e sistemas, o que ajudou a melhorar a colaboração entre as equipes de desenvolvimento e operações.

Já a **Walmart** investiu em uma cultura de automação, usando ferramentas de automação de testes e integração contínua para acelerar o processo de desenvolvimento de software e aumentar a eficiência.



# Estrutura Organizacional para DevOps

**Etsy**, por sua vez, criou uma cultura de experimentação, incentivando sua equipe a testar novas ideias e tecnologias regularmente. Isso ajudou a empresa a inovar rapidamente e a se adaptar às necessidades em constante mudança do mercado.

Independentemente da abordagem escolhida, é importante que as empresas adotem uma **mentalidade de colaboração e transparência** entre as equipes de desenvolvimento e operações. Além disso, a automação e monitoramento contínuo são fundamentais para garantir que o processo de desenvolvimento de software seja eficiente e de alta qualidade.





# Estudo de caso – Exemplo prático de um projeto DevOps




# Estudo de caso – Exemplo prático de um projeto DevOps

Suponhamos que uma empresa esteja desenvolvendo um aplicativo de comércio eletrônico que precise ser atualizado regularmente para corrigir bugs, adicionar novos recursos e melhorar a performance.

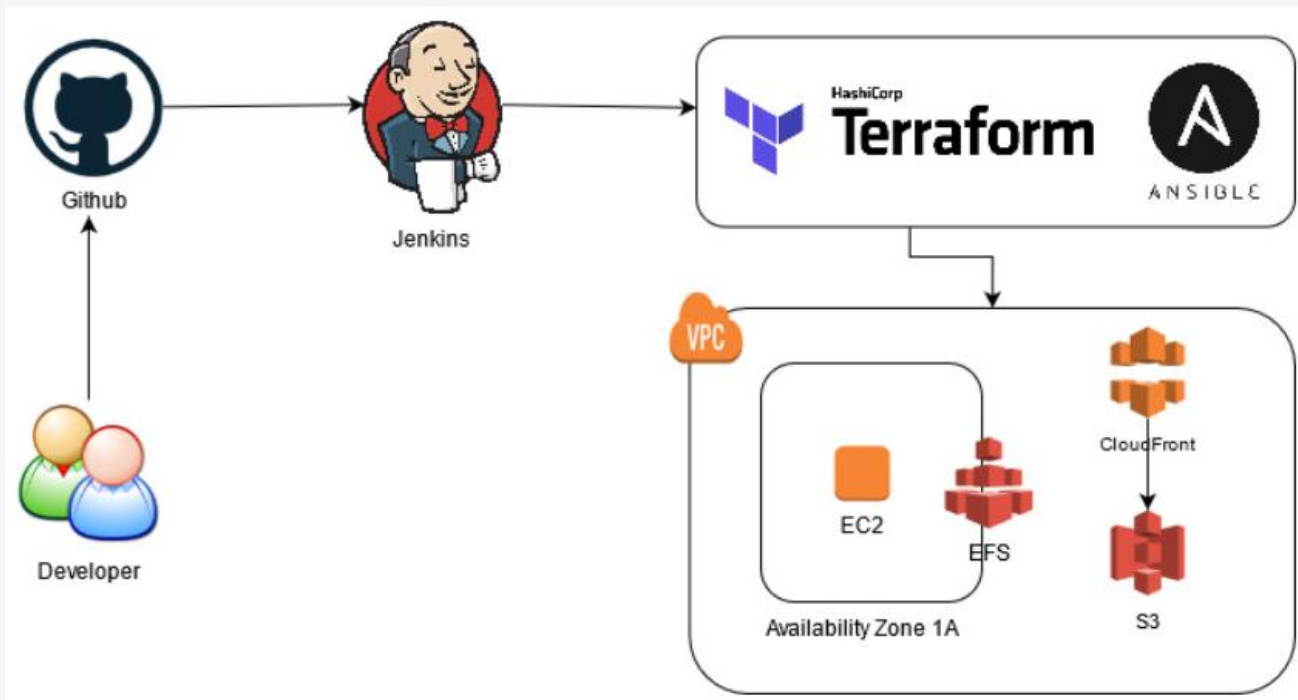


Para garantir a entrega rápida e eficiente dessas atualizações, a empresa pode adotar a cultura DevOps.



As equipes de desenvolvimento e operações trabalhariam juntas para automatizar o processo de integração e entrega contínua.





## Estudo de caso – Exemplo prático de um projeto DevOps

Eles usariam ferramentas como:

- ✓ Github para o controle de versão de código
- ✓ Jenkins para automatizar a compilação e testes de código
- ✓ Ferramentas de deployment automatizado como o Ansible ou Terraform

# Estudo de caso – Exemplo prático de um projeto DevOps

---

Além disso, eles criaram uma infraestrutura escalável e flexível, usando tecnologias de nuvem como o Amazon Web Services (AWS) ou Microsoft Azure.

---

Isso permitiria que eles escalassem os recursos do servidor automaticamente para lidar com Picos de tráfego durante os períodos de alta demanda.

---

A equipe de operações também implementaria ferramentas de monitoramento contínuo para garantir que o aplicativo esteja sempre disponível e funcionando corretamente.

## Estudo de caso – Exemplo prático de um projeto DevOps

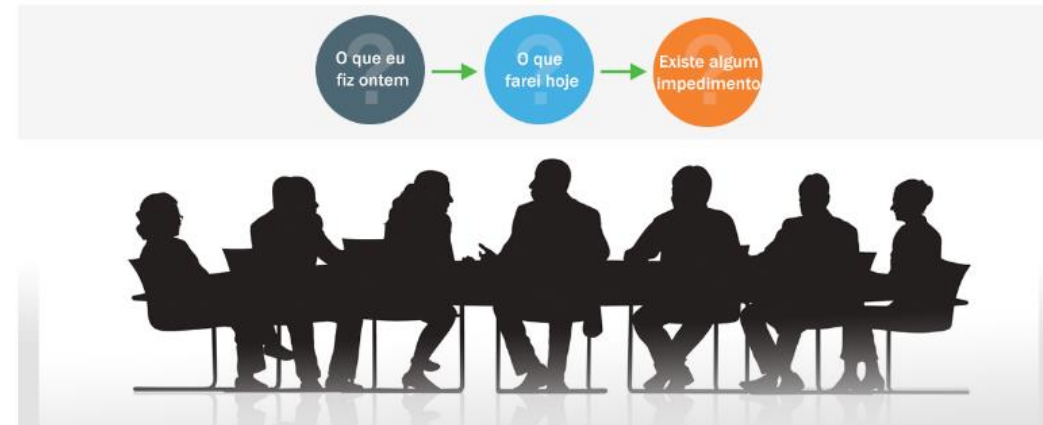
Para a monitoração, eles poderiam usar ferramentas como:

- Nagios, Zabbix ou Prometheus para monitorar o desempenho do aplicativo
- Grafana para gerar Dashboards que ajudam a identificar problemas e tomar medidas corretivas rapidamente



# Estudo de caso – Exemplo prático de um projeto DevOps

Por fim, a equipe de desenvolvimento e operações trabalhariam em conjunto para criar uma cultura de colaboração e transparência. Eles realizariam reuniões regulares para discutir os problemas e progressos, compartilharia métricas e dados de desempenho para garantir que todos estivessem na mesma página, eu trabalharei um juntos para solucionar problemas rapidamente.



## Estudo de caso – Exemplo prático de um projeto DevOps

---

Com essas práticas em ação, a equipe de desenvolvimento pode entregar atualizações de software para o aplicativo de comércio eletrônico de maneira rápida, eficiente e segura . Além disso, a equipe pode iterar rapidamente e obter feedback dos usuários, permitindo que eles melhorem continuamente o aplicativo com novos recursos e funcionalidades.



# Estudo de caso – Exemplo prático de um projeto DevOps

## Conhecendo a Arquitetura

