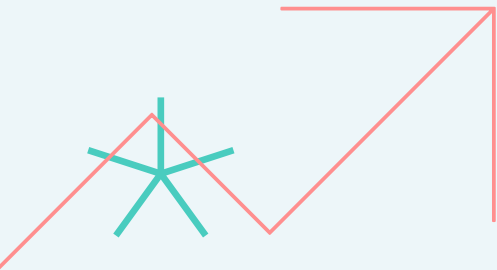
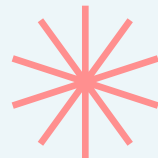
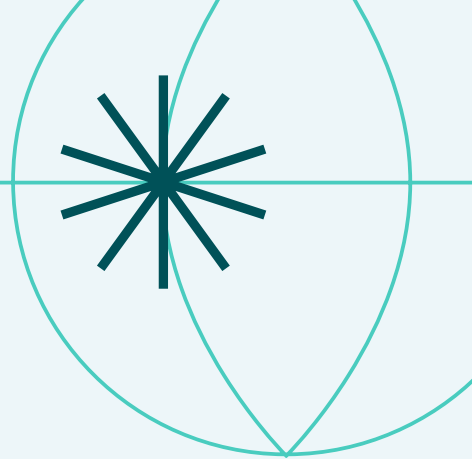




DevOps

0 que danado é isso?!



Quem sou eu?

- Me chamo Raissa (Rai);
- Sou gateira. Tutora de Ronaldo, Lara e Laila;



- Apaixonada por fotografia;
- Viajante sempre que dá 🐼;
- Pernambucana, bairrista de carteirinha;
- Trabalho atualmente no **Picpay**;
- Formação em Design de Interiores e Sistemas de Informação pela UFRPE.



Índice



01

O que é DevOps e DevSecOps?

02

CI/CD: continuous integration e continuous delivery

03

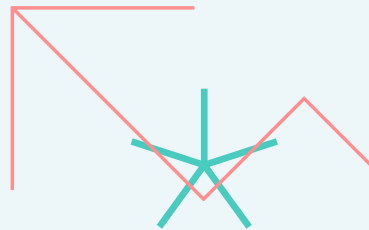
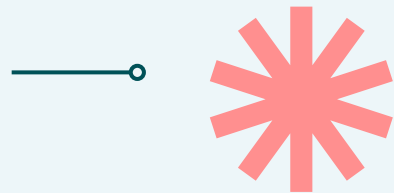
Componentes

04

Serviços e ferramenta

05

Hands on!





01

O que é DevOps?

cultura, automação e plataforma





Origem:

2008 na Agile Conference em Toronto - Canadá;

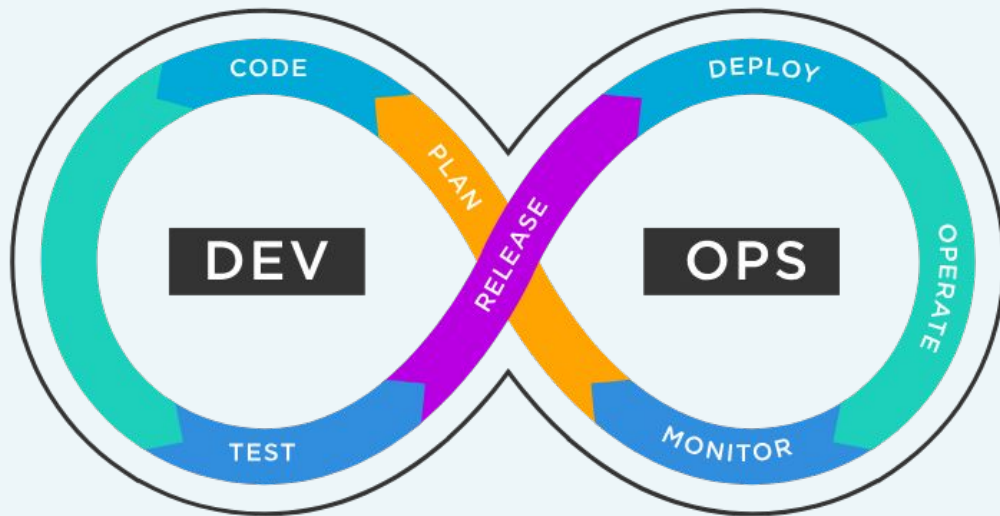
2009, na O'Reilly Velocity Conference e DevOpsDay, em Ghent, na Bélgica;

2011, Gartner publicou um relatório no qual afirmava que, até o final de 2015, DevOps se tornaria a principal estratégia em 20% das organizações mundiais. E de lá para cá, diversos gigantes do mercado passaram a investir pesado em DevOps, como IBM, HP, Amazon, Netflix, Walmart, Adobe e outras.



Definição:

É uma abordagem de cultura, automação e design de plataforma que tem como objetivo agregar mais valor aos negócios e aumentar a capacidade de resposta às mudanças por meio de entregas de serviços rápidas e de alta qualidade.





Quais são as características?

01

Automação

04

Métricas e análise de dados

02

Escalabilidade

05

Compartilhamento/colaboração
/Aprendizado

03

Segurança e qualidade

06

Métricas e análise de dados

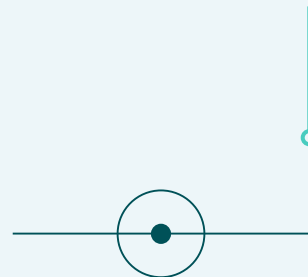
A metodologia descreve abordagens que ajudam a acelerar os processos necessários para levar uma ideia do desenvolvimento à implantação em um ambiente de produção no qual ela seja capaz de gerar valor para o usuário.

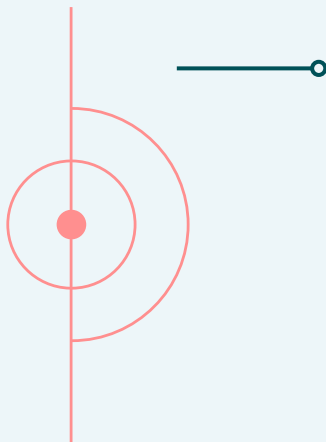




Desafíos

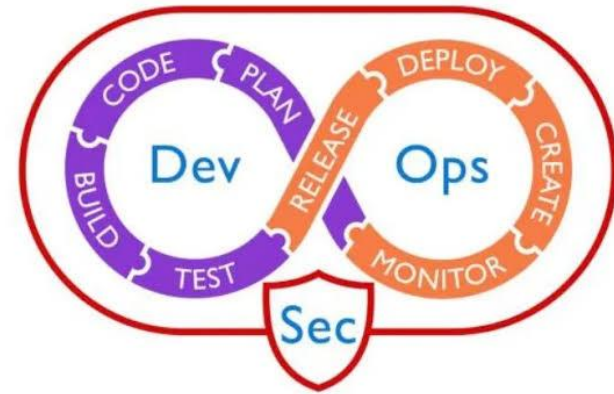
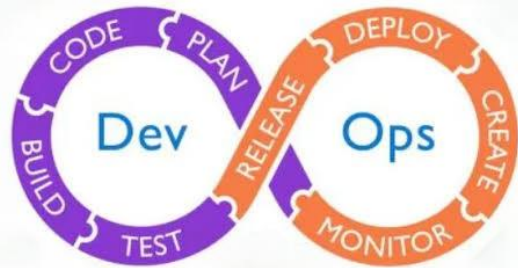
- Resistência à mudanças
- Comunicação
- Escolha das Ferramentas
- Integração de Ferramentas
- Automação e implantação de Testes e guardrails
- Segurança e Gerenciamento de Segredos
- Segurança de dados
- Migração de dados
- Visibilidade
- Resposta a Incidentes
- Consistência de Ambientes
- Ambientes Dinâmicos
- Gerenciamento de dependências
- Retorno sobre Investimento
- Regulamentações e Conformidade
- Metrificar





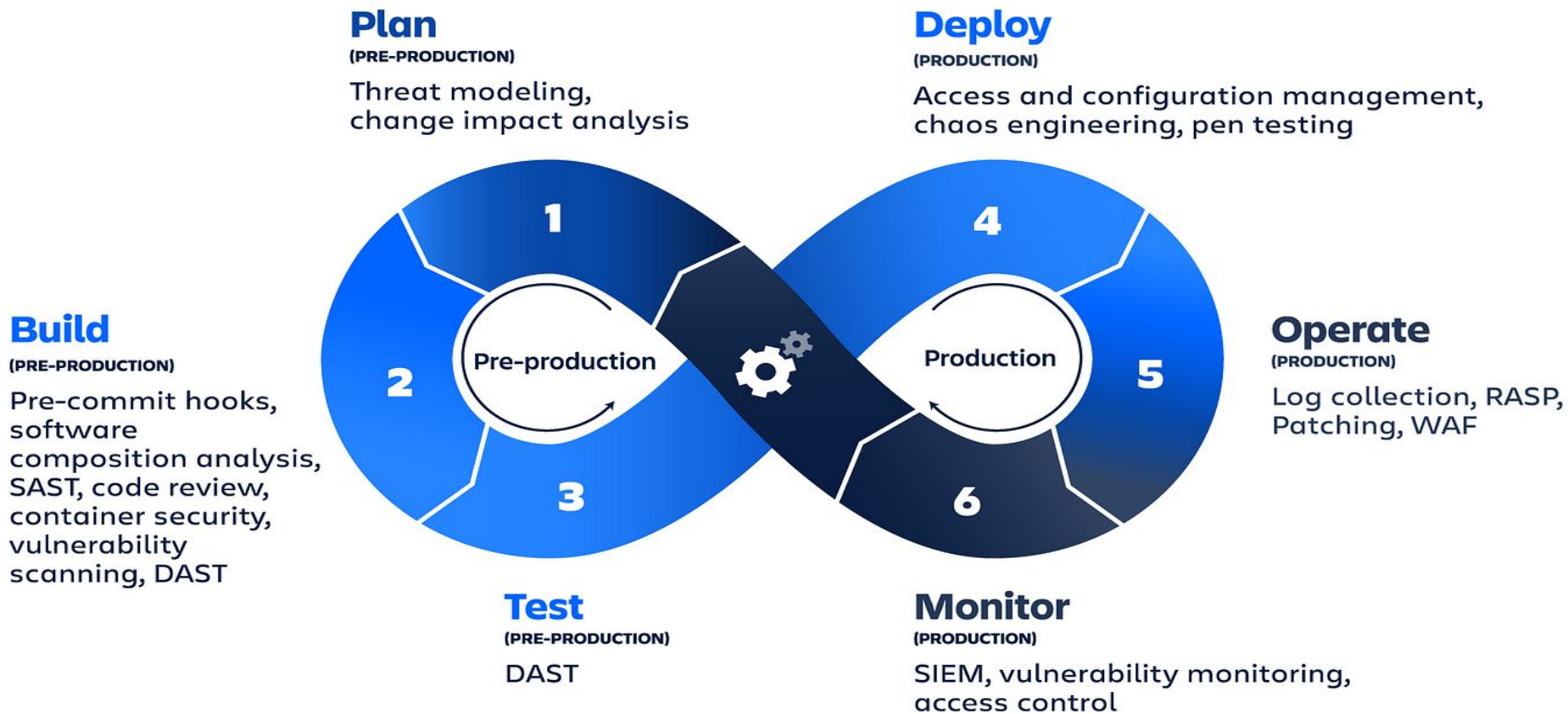
E o DevSecOps?





DevSecOps

DevSecOps







02

CI/CD: continuous integration, continuous delivery e continuous deployment

Do desenvolvimento à implantação em produção





Build

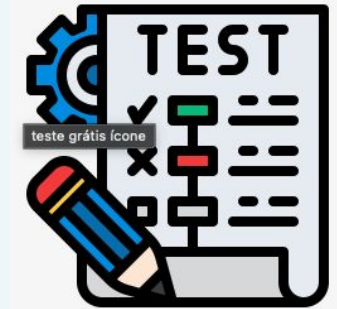
É a compilação do código, no caso da esteira de CI esta compilação vai ocorrer sempre em um ambiente configurado de forma **padrão** e com todas as dependências da aplicação, evitando assim o "na minha máquina funciona".





Testes

Servem para verificar se o software consegue entregar corretamente tudo que ele propõe, são importantes para garantir o funcionamento do software. Que cada uma das unidades de configurações consiga atuar sozinha e em conjunto, em busca da melhor experiência possível para os usuários. Nesse sentido, o teste de software é uma ferramenta fundamental tanto para que o software funcione quanto para que ele seja seguro. Esses testes podem ser realizados em qualquer uma das partes do software, desde a unidade pequena até seu funcionamento como um todo, analisando também o número de dados e a sua segurança.





Versionamento

É um processo para gestão e controle de mudanças de código. Ele envolve a atribuição de números ou rótulos, geralmente chamados de **versões** ou **releases**, a medida que o software evolui.

Dessa forma, o versionamento

- Desenvolvedores e usuários acompanhem as mudanças;
- Identificar versões estáveis;
- Compreender as alterações introduzidas;
- Gerenciar compatibilidade entre diferentes versões;
- Possibilidade de reverter para versões anteriores se necessário.

permite

1

major
Incompatibilidade



12

minor
novas
funcionalidades

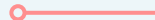


7

patch
correções de bugs



que:

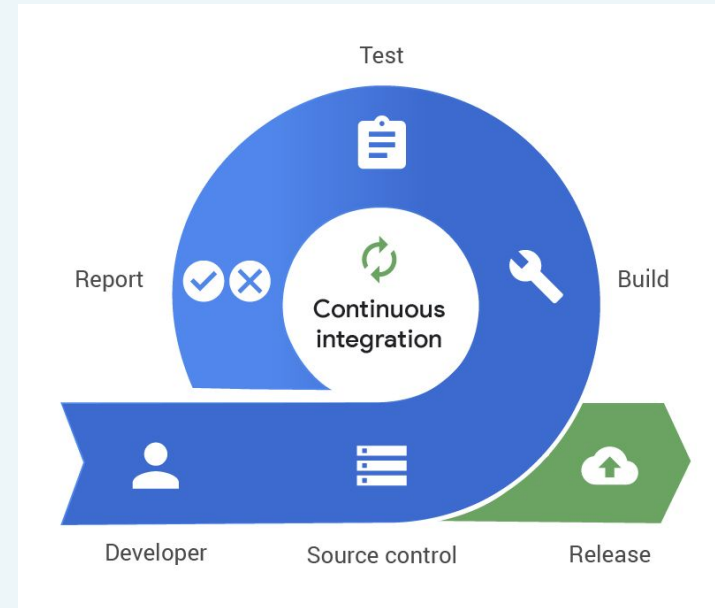




Integração Contínua (Continuous Integration – CI)

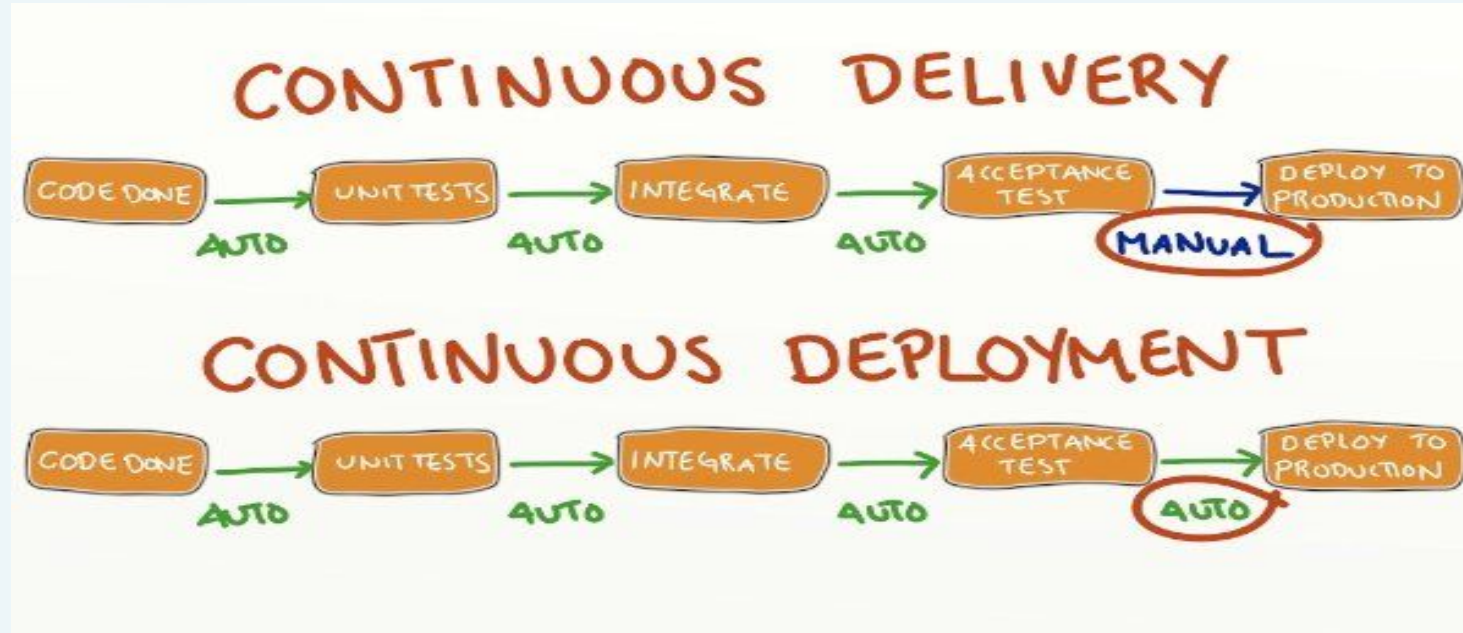
É uma prática de automatizar a **integração** de **alterações** de código de vários contribuidores em um único projeto de software permitindo que os desenvolvedores mesclam com frequência as alterações de código em um repositório central onde **builds**, **análises** e **testes** são executados.

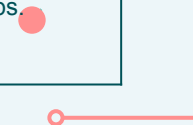



Ferramentas automatizadas são usadas para afirmar a correção do novo código antes da integração no código principal.






Continuous Delivery e Continuous Deployment (CD)





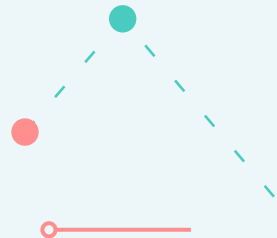
	Entrega Contínua (Continuous Delivery)	Deploy Contínuo (Continuous Deployment)
	Alterações de códigos são automaticamente preparadas para implantação	Alterações de códigos são liberadas automaticamente no ambiente de produção
Benefícios	<ul style="list-style-type: none">- Maior produtividade;- Automação completa do processo de entrega;- Bugs são encontrados e podem ser solucionados mais cedo;- Novas atualizações lançadas com mais frequência.	<ul style="list-style-type: none">- Processo de liberação mais rápido;- Não há necessidade de intervenções humanas;- É fácil encontrar e consertar problemas;- Novas atualizações lançadas com mais frequência.
Desafios	<ul style="list-style-type: none">- Requer um CI sólido;- Requer uma boa suíte de testes;- Precisa de um processo de implantação automatizado;- Uso adequado de features flags.	<ul style="list-style-type: none">- Requer um excelente conjunto de testes;- Documentação deve permanecer atualizada;- Features flags se torna uma parte inerente ao processo DevOps.





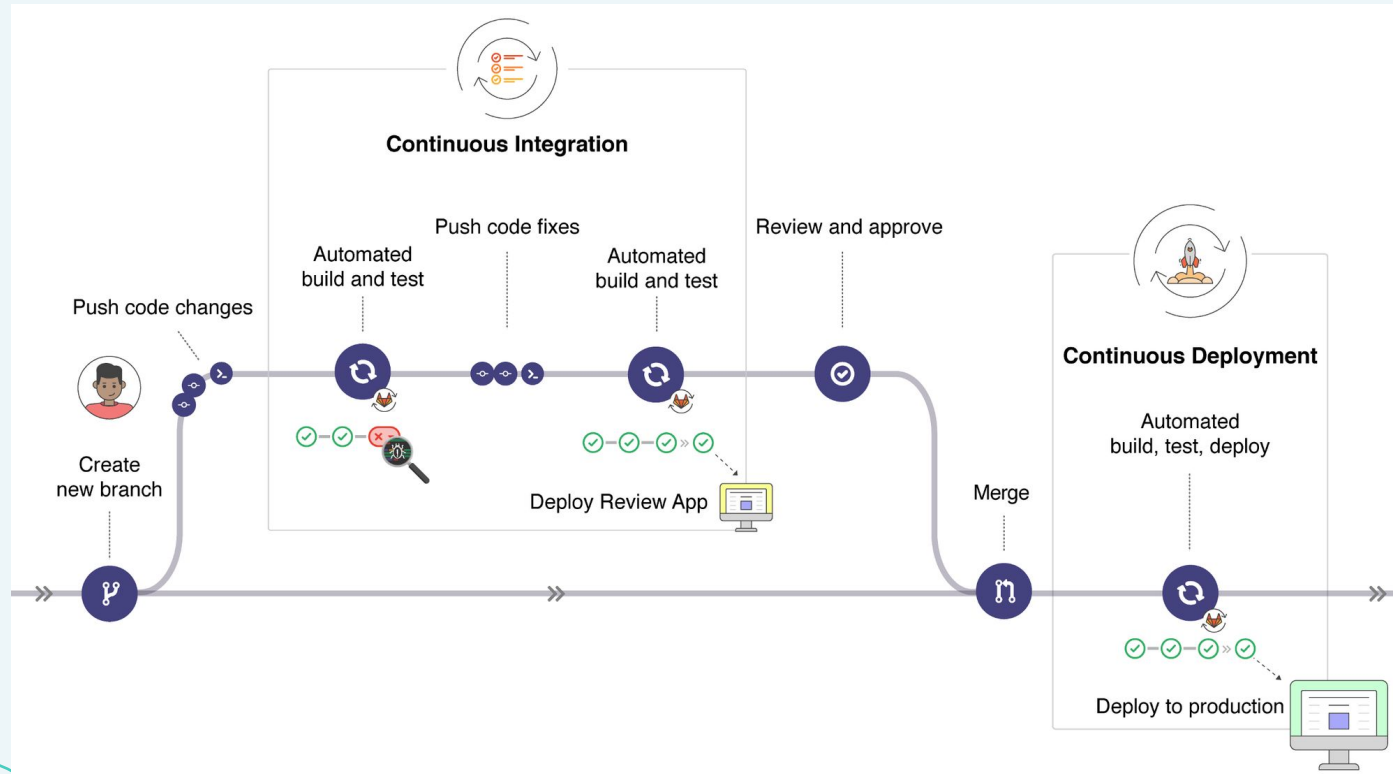
Resumindo...

Integração Continua	Entrega Continua	Deploy Continuo
<p>Refere-se à integração, construção e teste de código dentro do ambiente de desenvolvimento de software</p> <p>promove a prática de compartilhamento de código na comunidade de desenvolvimento de software, mesclando alterações em um repositório de código compartilhado centralmente</p>	<p>Significa que o software que passou pelo build pode ser liberado para produção</p> <p>Significa que um deploy pode acontecer com uma certa frequência, mas cada versão é decidida e liberada caso a caso</p>	<p>Significa que as mudanças, através do pipeline, são automaticamente colocados em produção</p> <p>Habilita que múltiplos deploys possa ser realizados por dia</p> <p>Requer que a Entrega Contínua esteja sendo feita</p>



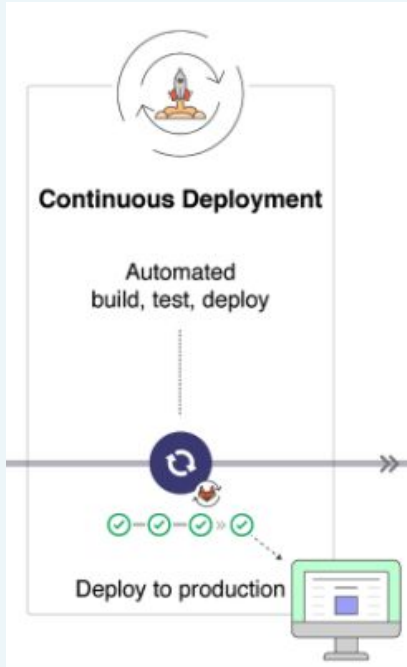


Fluxo de CI/CD da Aplicação





Fluxo de CD da Aplicação



Build para produção

Ofuscação

Versionamento

Testes

- Feature Flags
- Unitários
- Shadow
- End to End

Análise estática de código

Sonar

GuardRails

Distribuição

Canary

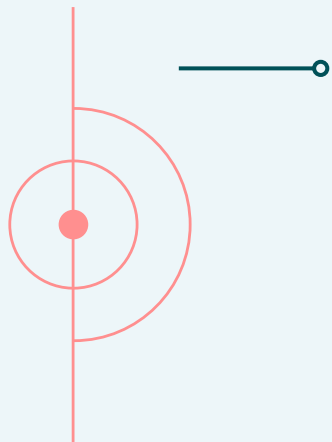




Boas práticas

- Integração contínua;
- Manter os repositórios de códigos bem estruturado e organizado;
- Configurar regras de proteção de Branch;
- Manter um ambiente estável, escalável e atualizado;
- Planejamento e automação para deploy, falhas e rollback;
- Monitoramento;
- Rastreabilidade;
- Feature flag;
- Arquitetura bem estruturada (ex: módulos);
- Tempo de build (custos);
- Tamanho da aplicação (custos);
- Segurança;





03

Componentes





Monitoramento

É o processo de observar, medir e analisar o desempenho, a disponibilidade e a funcionalidade de sistemas, aplicações e infraestrutura de TI, para com isso garantir que os sistemas estejam operando conforme o esperado, podendo identificar problemas rapidamente e proporcionar dados valiosos para melhorias contínuas.

- Desempenho: Aplicações e servidores;
- Infraestrutura;
- Segurança: Logs, Vulnerabilidades;
- Banco de dados;
- Experiência de usuário.



Grafana



Prometheus



AWS CloudWatch



dynatrace



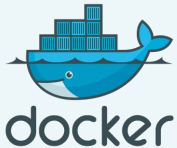
new relic





Containers

Containers são pacotes leves e portáteis que contêm tudo o que uma aplicação precisa para ser executada, incluindo o código, runtime, bibliotecas e dependências.



...





Orquestração de Containers

A orquestração de containers é uma prática que envolve a automação do gerenciamento, escalonamento e operação de containers. Ela facilita a gestão de um grande número de containers, especialmente em ambientes de produção complexos.

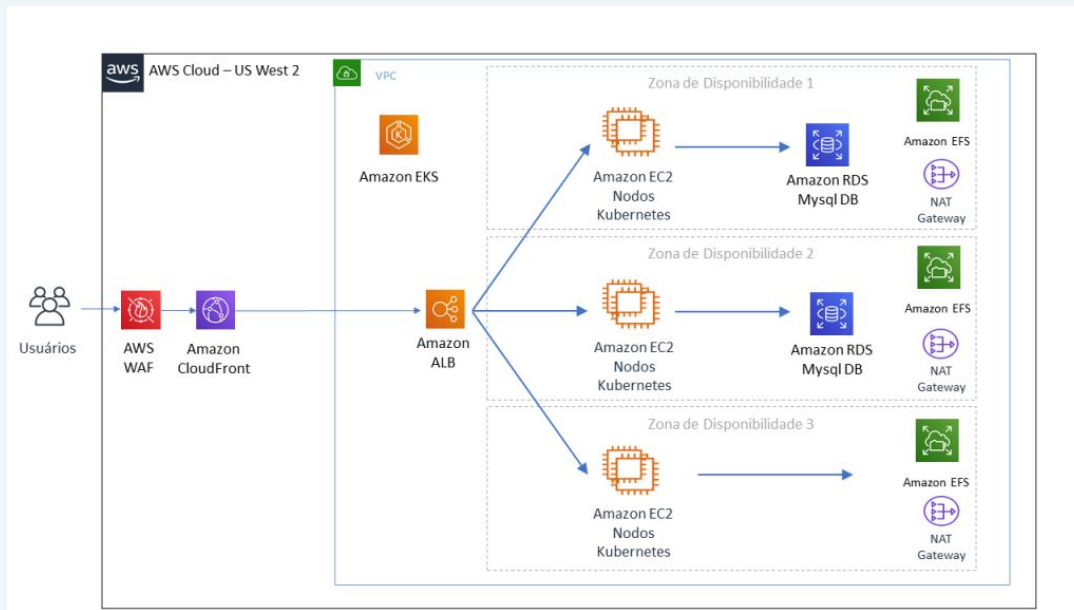


...





Arquitetura de Infraestrutura



Devido à pandemia, COVID-19, o Governo do Chile declarou estado de quarentena em várias regiões do país. Milhões de cidadãos foram alcançados por esta medida, com o objetivo de controlar a pandemia, o governo ordenou o controle da circulação de cidadãos através da concessão de licenças temporárias. A fim de otimizar a emissão de tais licenças, foi decidido que elas deveriam ser concedidas através de uma plataforma web.

Link do case: [Evoluindo diante de uma pandemia, como escalar para atender um país inteiro usando tecnologias sem servidor na AWS](#)





04

Serviços e Ferramentas





Serviços

PRODUCT	aws	Microsoft Azure	Google Cloud Platform
Virtual Servers	Instances	VMs	VM Instances
Platform-as-a-Service	Elastic Beanstalk	Cloud Services	App Engine
Serverless Computing	Lambda	Azure Functions	Cloud Functions
Docker Management	ECS	Container Service	Container Engine
Kubernetes Management	EKS	Kubernetes Service	Kubernetes Engine
Object Storage	S3	Block Blob	Cloud Storage
Archive Storage	Glacier	Archive Storage	Coldline
File Storage	EFS	Azure Files	ZFS / Avere
Global Content Delivery	CloudFront	Delivery Network	Cloud CDN
Managed Data Warehouse	Redshift	SQL Warehouse	Big Query

Guia de Equivalência de Serviços em Diferentes Nuvens

www.chapeudealuminio.com.br

@chapeu.aluminio



aws	Azure	Google Cloud	ORACLE CLOUD
Elastic Compute Cloud (EC2)	Virtual Machine	Compute Engine	Virtual Machine
Elastic Kubernetes Service (EKS)	Azure Kubernetes Service (AKS)	Google Kubernetes Engine (GKE)	Oracle Container Engine
Lambda	Azure Functions	Cloud Functions	OCI Functions
Simple Storage Service (S3)	Blob Storage	Cloud Storage	Object Storage
Elastic Block Store	Managed Disk	Persistent Disk	Persistent Volume
Elastic File System	File Storage	File Store	File Storage
Virtual Private Cloud	Virtual Network	Virtual Private Cloud	Virtual Cloud Network
Route 53	DNS	Cloud DNS	DNS
Elastic Load Balancing	Load Balancer	Cloud Load Balancing	Load Balancer
Web Application Firewall	Web Application Firewall	Cloud Armor	Web Application Firewall
RDS	SQL Database	Cloud SQL	ATP
DynamoDB	Cosmos DB	Firestore	NoSQL Database
Redshift	Synapse Analytics	BigQuery	Autonomous Data Warehouse
Elastic MapReduce	HDInsight	Dataproc	Big Data
Kinesis	Streaming Analytics	Dataflow	Streaming
SageMaker	Machine Learning	Vertex AI	Data Science
Glue	Data Factory	Data Fusion	Data Integration
EventBridge	Event Grid	Eventarc	Events
Simple Queuing Service	Storage Queues	Pub/Sub	Streaming
Simple Notification Service	Service Bus	Firestore Cloud Messaging	Notifications
CloudWatch	Monitor	Cloud Monitoring	Monitoring
CloudFormation	Resource Manager	Deployment Manager	Resource Manager
IAM	Active Directory	Cloud Identity	IAM
KMS	Key Vault	Cloud KMS	Vault



Serviços (AWS)

Serviços de computação –

- Amazon EC2
- AWS Lambda
- AWS Elastic Beanstalk
- Amazon EC2 Auto Scaling
- Amazon ECS
- Amazon EKS
- Amazon ECR
- AWS Fargate



Serviços de segurança, identidade e conformidade –

- AWS IAM
- Amazon Cognito
- AWS Shield
- AWS Artifact
- AWS Key Management Service (KMS)



Serviços de armazenamento –

- Amazon S3
- Amazon S3 Glacier
- Amazon EFS
- Amazon EBS



Serviços de banco de dados –

- Amazon RDS
- Amazon DynamoDB
- Amazon Redshift
- Amazon Aurora



Serviços de redes e Entrega de Conteúdo –

- Amazon VPC
- Amazon Route 53
- Amazon CloudFront
- Elastic Load Balancing



Serviços de gerenciamento e Governança –

- AWS Trusted Advisor
- AWS CloudWatch
- AWS CloudTrail
- AWS Well-Architected Tool
- AWS Auto Scaling
- Interface da linha de comando da AWS
- AWS Config
- Console de Gerenciamento da AWS
- AWS Organizations



Gerenciamento de custos da AWS –

- Relatório de custos e uso da AWS
- Orçamentos da AWS
- AWS Cost Explorer





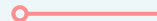
Ferramentas de CI/CD

- Backstage (Catálogo de serviços);
- Github;
- Kubernetes;
- Helm charts (gerenciador de pacotes do Kubernetes);
- Tekton;
- Helm;
- Sonarcloud;
- Harness;
- ArgoCD;
- AWS (ECR, EC2, S3 e etc);

...

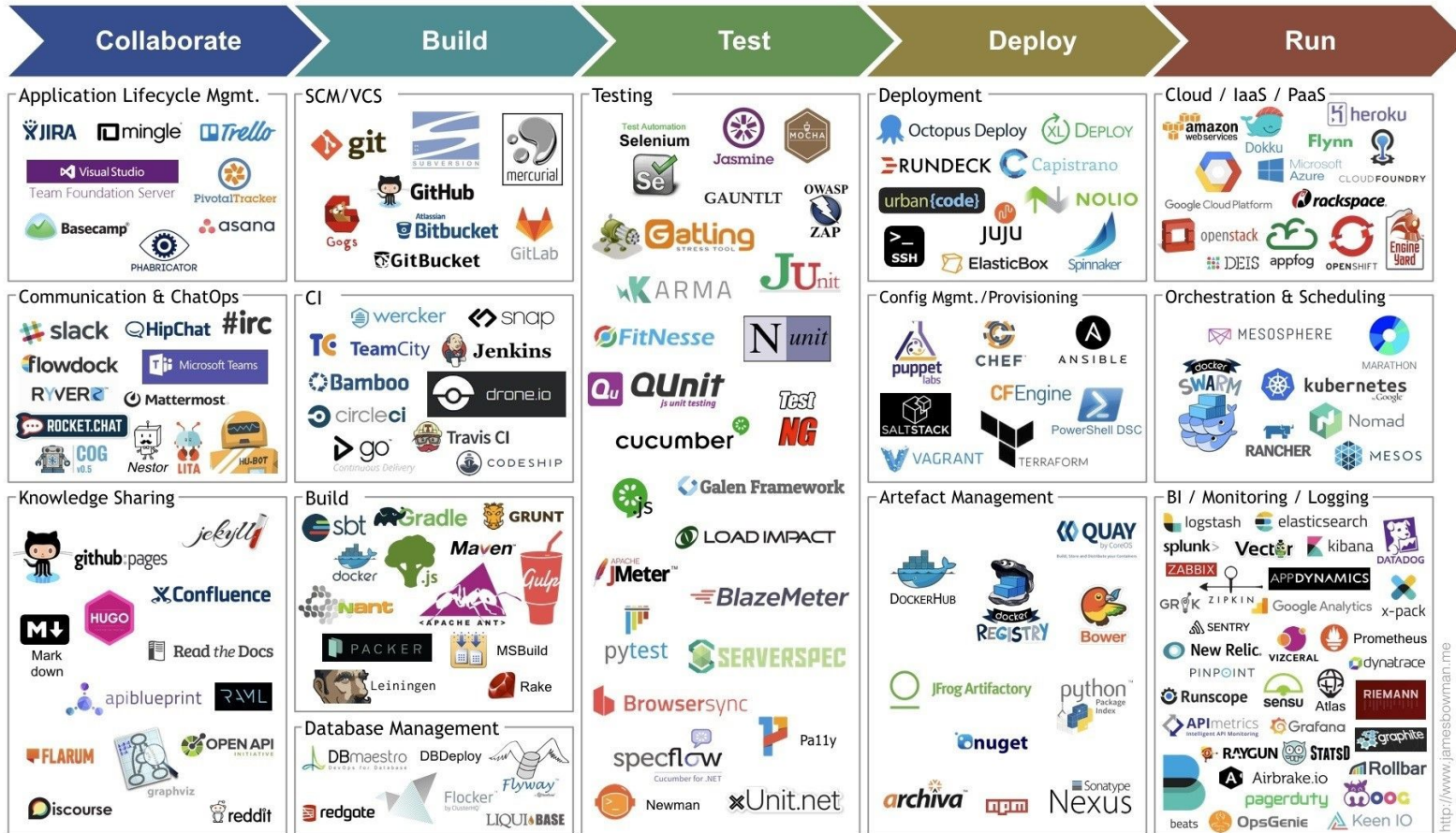
Ahh, mas só isso?

Calma!
Next



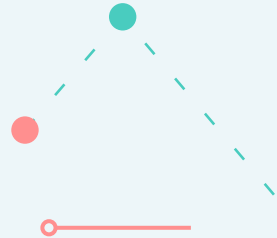


Ferramentas



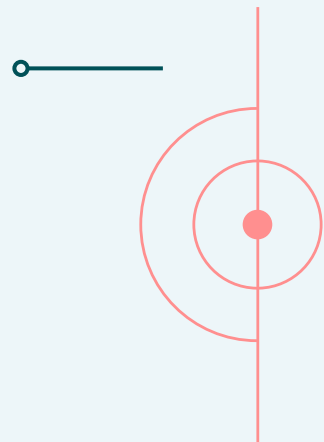


Certificação





05



Hands on!

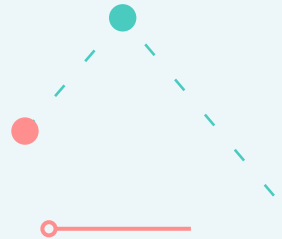
Exemplo prático utilizando GitActions





Requisitos:

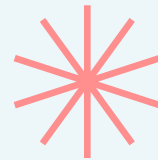
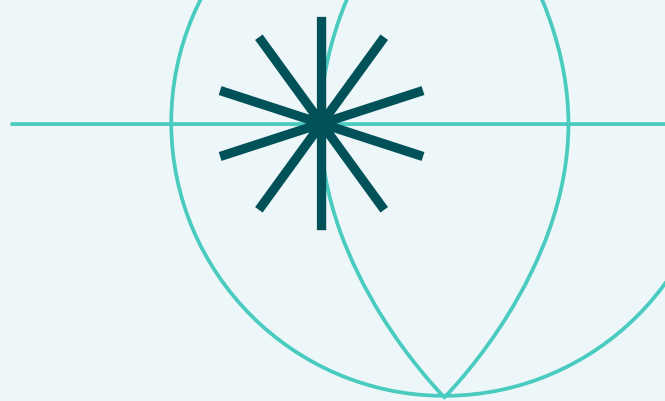
- Eu, como pessoa desenvolvedora, ...





Obrigada!

Perguntas???



<https://www.linkedin.com/in/raissa-brizeno/>

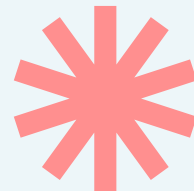
CRÉDITOS: Este modelo de apresentação foi criado pelo **Slidesgo**, e inclui ícones da **Flaticon**, infográficos e imagens da **Freepik** e conteúdo de **Sandra Medina**.



Referências



- <https://www.redhat.com/pt-br/topics/devops>
- <https://www.redhat.com/pt-br/topics/devops/what-is-ci-cd>
- <https://gaea.com.br/devops-na-pratica-dia-a-dia-do-desenvolvedor/>
- <https://backstage.io/>
- <https://helm.sh/docs/>
- <https://tekton.dev/docs/>
- <https://www.objective.com.br/insights/testes-de-software/#:~:text=O%20que%20C%3A9%20um%20teste,corretamente%20tudo%20que%20ele%20prop%3%B5e.>
- <https://gaea.com.br/conheca-a-incrivel-historia-do-devops/#:~:text=O%20embri%3A3o%20da%20hist%C3%B3ria%20do,desenvolvimento%20e%20opera%C3%A7%C3%B5es%20de%20TI.>
- <https://www.locaweb.com.br/blog/temas/codigo-aberto/versionamento-de-software-importancia/>



Indicação

