



CURSO
525

INFRAESTRUTURA ÁGIL
COM PRÁTICAS DEVOPS





Cultura DevOps

A História do DevOps

Anotações

1 O que é a Cultura DevOps?

2 Manifesto Ágil

3 Velocity Conference

4 DevOpsDays

Anotações

DevOps (Development + Operation)

É um processo de desenvolvimento e entrega de software que enfatiza a comunicação entre os profissionais de desenvolvimento e operacional (infraestrutura, rede e suporte).

O Termo DevOps tem tomado uma grande importância no mundo todo. Empresas com Amazon, Netflix, Facebook, Wallmart, Spotify, entre tantas outras, estão investindo em funcionários, ferramentas e ambientes que sigam as práticas DevOps, para melhorar a entrega e qualidade de seus produtos.

Hoje a palavra DevOps também está sendo utilizada para classificar funcionários, departamentos e times dentro do ambiente empresarial. Funcionários que possuem o perfil DevOps, têm qualidade para trabalhar tanto na área de Desenvolvimento de Software como na área de Administração de Sistema.

Objetivos de um ambiente com práticas DevOps:

- ✓ Melhorar a frequência dos deploys;
- ✓ Automatizar processos;
- ✓ Diminuir a ocorrência de erros em novas versões;
- ✓ Curtos períodos de tempo para mudanças e melhorias;
- ✓ Recuperação rápida em caso de falhas no ambiente;
- ✓ Padronização nos processos de configuração e servidores.

Anotações



Anotações

A História do DevOps – Manifesto Ágil

O **Manifesto ágil** foi criado em 2001 por um grupo formado de 17 desenvolvedores. Os pontos discutidos foram:

- ✓ Indivíduos e interações, mais que processos e ferramentas;
- ✓ Software em funcionamento, mais que documentação abrangente;
- ✓ Colaboração com o cliente, mais que negociação de contratos;
- ✓ Responder a mudanças, mais que seguir um plano.

A ideia do desenvolvimento ágil surgiu entre 1960 e 1990, com diversas metodologias para desenvolvimento, criadas durante o período. Todas com o intuito de auxiliar o desenvolvedor a entregar seus projetos o mais rápido possível, e que fossem de fácil resposta a mudanças. Porém, somente em 2001 que um grupo de 17 desenvolvedores se reuniram em um resort para realmente discutir quais os pontos positivos em diversas metodologias que utilizavam.

A partir dessa discussão, surgiu o Manifesto para Desenvolvimento Ágil de Softwares, que ficou popularmente conhecido como Manifesto Ágil. O Manifesto Ágil afirma que melhores resultados no desenvolvimento de software podem ser obtidos através da valorização de:

- ✓ Indivíduos e interações, mais que processos e ferramentas;
- ✓ Software em funcionamento, mais que documentação abrangente;
- ✓ Colaboração com o cliente, mais que negociação de contratos;
- ✓ Responder a mudanças, mais que seguir um plano.

A História do DevOps – Agile Conference

Entre 2001 e 2008, diversas discussões começaram sobre como agilizar as entregas, sendo uma delas a palestra **Infraestrutura Ágil de Andrew Schafer**.

Foi durante essa palestra que Andrew conheceu Patrick Debois, e juntos criaram o grupo **"Agile System Administrator"** no Google Docs.



O termo DevOps surgiu em 2009, porém a ideia por trás já existia anos antes. A primeira vez que houve uma discussão sobre o assunto foi durante a Agile Conference de 2008, onde Andrew Schafer apresentou sua palestra de Infraestrutura Ágil para uma plateia de apenas uma única pessoa – Patrick Debois.

Após a palestra, ambos discutiram diversos assuntos relacionados, sendo que posteriormente iriam criar um grupo sobre o assunto chamado "Agile System Administration" no Google Groups para difundir o assunto e convidar mais pessoas para colaborar com o assunto.

O'REILLY

Velocity

Em 2009, durante a Velocity Conference da O'Reilly, John Allspaw e Paul Hammond deram a famosa palestra chamada **"10+ Deploys Per Day: Dev and Ops Cooperation at Flickr"**.

Durante a palestra foi discutido assuntos como a interação entre os Desenvolvedores e a equipe de Operações, como conseguir um aumento dos deploys com ferramentas e mudanças culturais.

No ano seguinte, ocorreu a famosa palestra do Flickr, intitulada 10 Deploys a Day: Dev and Ops Cooperation at Flickr. Debois, comentou no Twitter que lamentava não estar presente na palestra, recebendo do Flickr a seguinte resposta: "Porque não organizar sua própria conferência de Velocity na Bélgica?".

Motivado pela palestra dos engenheiros da Flickr, Patrick decidiu criar sua própria conferência de DevOps em Gant, Bélgica. A conferência ficou conhecida como DevOpsDays e foi um sucesso.

Para lembrar o dia, foi criado a tag #DevOps no Twitter, que posteriormente seria usado como o nome da cultura.



Anotações

Desde então, cada vez mais pessoas estão se tornando adeptas da cultura DevOps. Alguns dos acontecimentos mais recentes que favoreceram isso foram:

O Projeto Fenix – Uma fábula que conta como um analista de TI salvou o departamento de uma companhia com a cultura DevOps;



Surgimento de diversas tecnologias de apoio a essa ideologia.

Anotações

- 1 O que é a Cultura DevOps?
- 2 Manifesto Ágil
- 3 Velocity Conference
- 4 DevOpsDays

Anotações



Cultura DevOps

Infraestrutura como Código

Anotações

1 Infraestrutura como Código

2 Entendendo Desenvolvimento Ágil

3 Metodologias Ágil

4 Ambientes Transitórios

Anotações

Infraestrutura como código (Infrastructure as Code), é uma forma de construir e gerenciar uma infraestrutura da mesma maneira que ocorre no desenvolvimento de software.



Infraestrutura como código (Infrastructure as Code - IaC) é um processo de gerenciamento e provisionamento de infraestruturas, onde todas as atividades e passos para sua criação e configuração são registrados em scripts, utilizados por ferramentas de automação (como Ansible e Puppet), e passam pelo mesmo processo de versionamento que um software.

Com esse processo, torna-se muito mais fácil a gestão de ambientes, pois garante que novos serviços que forem incluídos, possuam as mesmas configurações e versões de pacotes. Outro ponto positivo é que o profissional de operação para de se preocupar em configurar cada servidor, e passa a se preocupar em codificar o ambiente.

Objetivos de uma **Infraestrutura como Código**:

- ✓ Aproveitar o tempo com execução valiosas, ao invés de desperdiçá-las em tarefas rotineiras e repetitivas;
- ✓ Recuperar rapidamente o ambiente, em caso de falha ou perda total;
- ✓ Possibilitar mudanças na infraestrutura de forma fácil e automatizada;
- ✓ Garantir ambientes idênticos, baseados na mesma configuração;
- ✓ Testar as mudanças, antes que entrem em produção.

Anotações

Desenvolvimento Ágil de software ou Método Ágil, é um conjunto de metodologias de desenvolvimento de software. Alguns princípios da metodologia ágil são:

- ✓ Garantir a satisfação do cliente, entregando rapidamente e continuamente;
- ✓ Rápida adaptação a mudanças;
- ✓ Simplicidade;
- ✓ O design do software deve prezar pela excelência técnica;
- ✓ Cooperação constante entre pessoas que entendem do negócio e desenvolvedores.

Anotações

Dashboard KanbanFlow

The screenshot shows a Kanban-style dashboard titled "Project X". The interface is divided into four main columns:

- To-do:** Contains 10 items, each with a small user icon and a brief description. Examples include "Allow user to change picture", "Prepare speech comparing JavaScript and CoffeeScript", and "Use SASS for stylesheets".
- Do today:** Contains 5 items, also with user icons and descriptions. Examples include "Produce financial report", "Write blog entry", and "Review security guidelines".
- In progress:** Contains 3 items, labeled "3 / 3". Each item has a checklist. Examples include "Advertise our new product" (with "Make pressrelease" checked) and "Implement CRM integration".
- Done:** Contains 4 items, each with a small user icon and a brief description. Examples include "Meeting with Acme" and "Create a page on Google+".

At the top right, there are links for "Administration" and "John Smith". Below the columns, specific dates are listed: "Today", "Monday, 25 March", "Monday, 30 January", and "Sunday, 29 January".

Anotações

A Virtualização e Computação em Nuvem (Cloud Computing), também disponibilizam a possibilidade de construir ambientes transitórios. **Ambientes transitórios** são ambientes de curta duração que são terminados com uma certa frequência.

**Ambientes baseados
em Scripts**

São completamente baseados em scripts, atribuídos em versões e testados.

**Ambientes de
Self-service**

Qualquer pessoa autorizada pode ativar um novo ambiente.

Término Automático

Os ambientes são automaticamente finalizados, de acordo com a política determinada na empresa.

Anotações

1 Infraestrutura como Código

2 Entendendo Desenvolvimento Ágil

3 Metodologias Ágil

4 Ambientes Transitórios

Anotações



Cultura DevOps

Conhecendo as Ferramentas

Anotações

1

Visão Geral

2

Topologia

3

Conhecendo as Ferramentas

Anotações



Em nosso laboratório, utilizaremos diversas ferramentas para práticas DevOps:

Vagrant

Ansible

Gogs

Rundeck

Docker

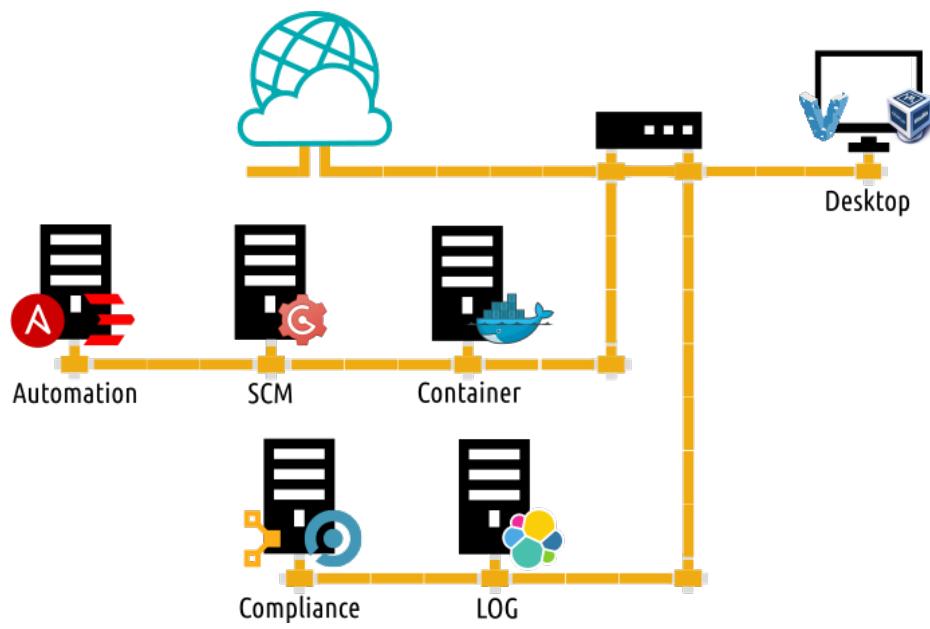
Puppet

Elastic Stack

Rocket Chat



Anotações



Anotações



Inicialmente, vamos realizar a construção do nosso ambiente como código.

Através desse código será feito o provisionamento de todas as máquinas virtuais usando o Vagrant, um construtor de ambientes virtuais produzido pela HashiCorp.

Anotações

A parte da instalação de serviços será delegado para o Ansible, uma ferramenta de provisionamento e gerencia de configuração, desenvolvido pela Ansible Inc. em 2012 e adquirida pela Red Hat em 2015.



ANSIBLE

Anotações



Gogs

A painless self-hosted Git service.

O próximo passo será criar uma maneira de versionarmos o código de nossa infraestrutura.

Para esse versionamento, será usado a ferramenta Gogs (Go Git Service) que é uma aplicação escrita em GoLang para gerenciamento de repositórios GIT, permitindo assim, colaboração de outras pessoas no projeto.

Anotações

Para automatização da nossa infraestrutura e deploy, utilizaremos o Rundeck que é uma plataforma Open-Source de gerenciamento de operações que conecta processos e ferramentas. Através do mesmo, iremos interligar as ferramentas a fim de entregar nosso ambiente provisionado de maneira automática.



Anotações

Para manter esse ambiente, será necessário uma ferramenta de gerenciamento de configurações. Neste caso, vamos utilizar o Puppet com o código sendo versionado e validado da mesma maneira que fizemos com o Ansible.



Anotações



Através do Docker, podemos provisionar aplicações em contêineres para sustentação da nossa infraestrutura DevOps. Atualmente, o Docker é a ferramenta de contêineres mais utilizada no mercado.

Anotações

Quando se tem um parque muito grande de máquinas, é necessário o uso de uma ferramenta para centralizar o armazenamento e visualização de logs. Para isso, será usada a solução Elastic Stack (Elasticsearch + Logstash + Kibana) da Elastic, dessa forma será permitido concentrar os logs de todas as máquinas em um único local para análise.



elasticsearch



logstash



kibana

Anotações



Utilizaremos o Rocketchat como ferramenta de colaboração. Vamos integrar para que as ações realizadas no Gogs e no Rundeck sejam refletidas no Chat de Operações. Também utilizaremos o Hubot para disparar jobs do Rundeck, através da metodologia ChatOps.

Anotações

- 1 Visão Geral
- 2 Topologia
- 3 Conhecendo as Ferramentas

Anotações
