



DEVOPS AVANÇADO

Cássio Trindade - Aula 01

Professores

CÁSSIO TRINDADE

Professor Convidado

Profissional da área de TI, trabalhando há mais de uma década com a formação de profissionais, dando aulas no Instituto Federal do Rio Grande do Sul, na Faculdade Dom Bosco, Universidade Luterana do Brasil (ULBRA), Pontifícia Universidade Católica do RS (PUCRS) e na TargetTrust. Atualmente atuando como Arquiteto de Software na PUCRS, sendo responsável pela condução e elaboração de mais de 90 projetos diretamente com alunos do curso de Engenharia de Software, trabalhando com as mais variadas tecnologias. Mais de 30 anos de experiência nas áreas de desenvolvimento de software, aplicativos para celulares e sistemas corporativos e para internet desde projetos de e-commerce para o Sonae Portugal e site de classificados digitais do Grupo RBS a dezenas de aplicativos mobiles.

MARCELO VEIGA NEVES

Professor PUCRS

Possui doutorado em Ciência da Computação pela Pontifícia Universidade Católica do Rio Grande do Sul (2015), mestrado em Ciência da Computação na Universidade Federal do Rio Grande do Sul (2009) e graduação em Ciência da Computação pela Universidade Federal de Santa Maria (2005). Tem experiência na área de Ciência da Computação, atuando principalmente nos seguintes temas: redes de computadores, processamento de alto desempenho e sistemas embarcados.

Ementa da disciplina

Estudo sobre entrega contínua (CD), uso de contêineres, orquestração e monitoramento. Experimentação de ferramentas: GitHub Actions, Docker Compose e Kubernetes e ferramentas de monitoração.

DEVOPS AVANÇADO

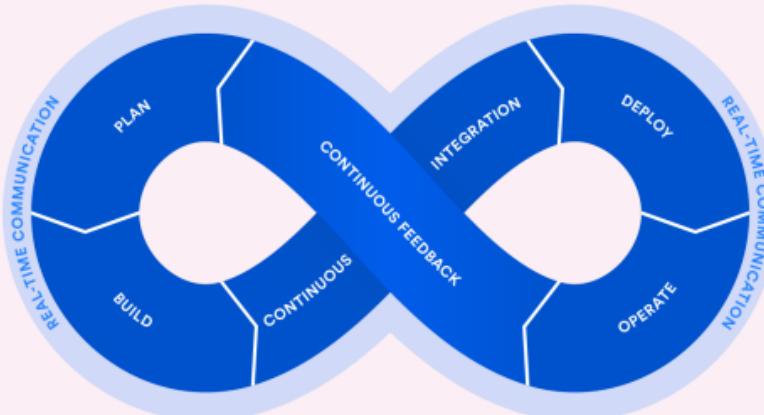
Cássio A. W. Trindade

- Mestre em Ciência da Computação
- Pós graduado em Engenheira de Software
- Especialista em Gestão da Tecnologia da Informação.
- Mais 30 anos de experiência em desenvolvimento de software, trabalhando em Sistemas Corporativos, Soluções digitais para Internet, Aplicativos Móveis, Arquitetura de Software.
- A 10 anos atuado na área de educação, atualmente Coordenador de Engenharia na JuntosSomos + (Triider/Habitissimo) na área de MatCon



**Cássio
Trindade**

AGENDA



01

Introdução
Planejamento
Build /
Construção

02

CI/CD
Continuous
Integration
Continuous
Delivery

03

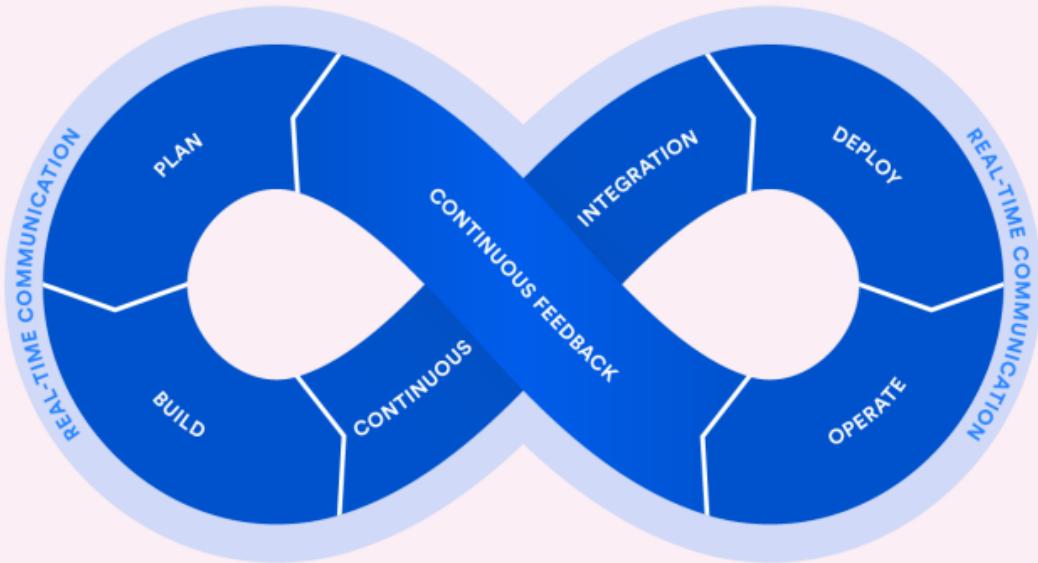
Deploy
Operate /
Operação

04

Continuous
Feedback

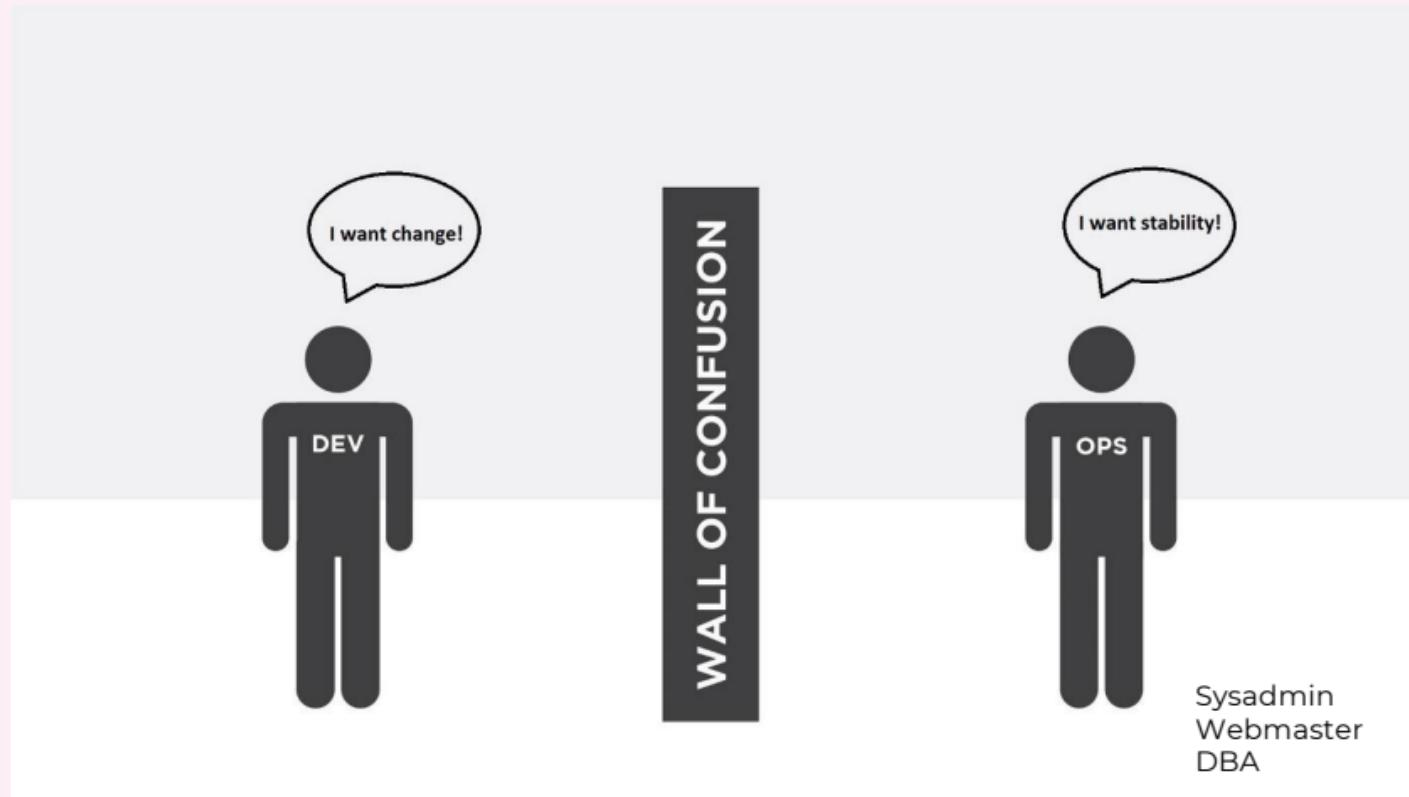
01

Introdução



O que NÃO é DevOps

- DevOps não é um Método.
- DevOps não é Cargo.
- DevOps não é Dev + Ops.
- DevOps não é uma Equipe.
- DevOps não é um Processo.
- DevOps não é uma Ferramenta.
- DevOps não é sinônimo de CI ou CD.
- DevOps não é exclusivo p/Métodos Ágeis.
- DevOps não é automação do processo de build e deploy



O que é DevOps

- Cultura de entrega continua de produtos digitais, com valor, que combina as áreas de "desenvolvimento" (development) e "operações" (operations). Que faz uso das melhores metodologias, práticas e ferramentas da área da TI.

CULTURA

Principais práticas

- Integração contínua (CI)
- Distribuição/implantação contínua (CD)
- Virtualização (IaaS) e containers
- Infraestrutura como código
- Gerenciamento de configuração
- Comunicação e colaboração

Inicio

10 deploys per day
Dev & ops cooperation at Flickr

John Allspaw & Paul Hammond
Velocity 2009

2014

The slide is a promotional graphic for AWS re:Invent 2014. At the top left is a video thumbnail showing a man in a dark t-shirt and jeans standing on a stage with a blue geometric background. To the right of the thumbnail, the AWS re:Invent logo is displayed in white and orange text, flanked by two decorative circular icons. The main title "AWS re:Invent" is centered in large white letters. Below it, the session identifier "ARC318" is shown. The main headline reads "Continuous Delivery at a Rate of 500 Deployments a Day!" in large white text. Underneath the headline, the speaker's name "Elias Torres, CTO at Driftt (Previously HubSpot VP of Engineer)" is listed. The date "November 14, 2014 | Las Vegas, NV" is at the bottom left. The Amazon Web Services logo is at the bottom center, and the AWS logo is at the bottom right.

AWS re:Invent

ARC318

Continuous Delivery at a Rate of
500 Deployments a Day!

Elias Torres, CTO at Driftt (Previously HubSpot VP of Engineer)

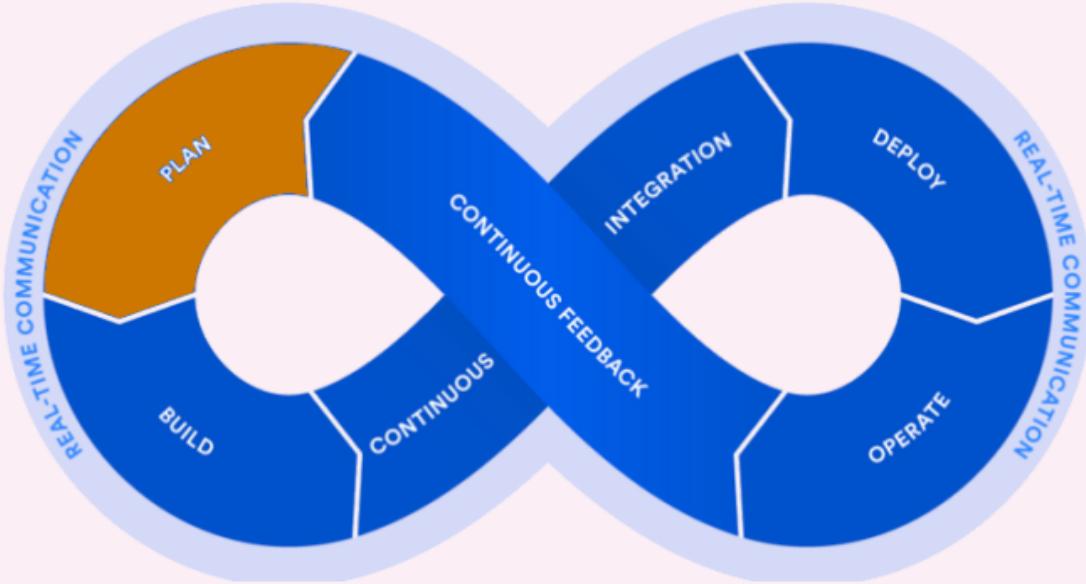
November 14, 2014 | Las Vegas, NV

© 2014 Amazon.com, Inc. and its affiliates. All rights reserved. May not be copied, modified, or distributed in whole or in part without the express consent of Amazon.com, Inc.

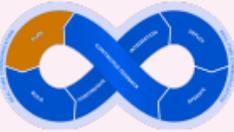
aws

01

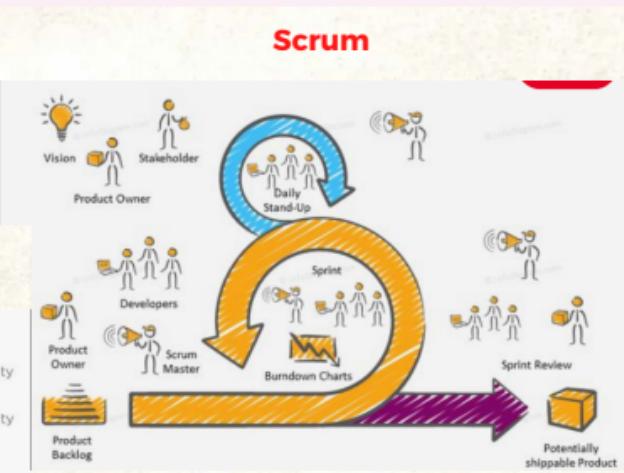
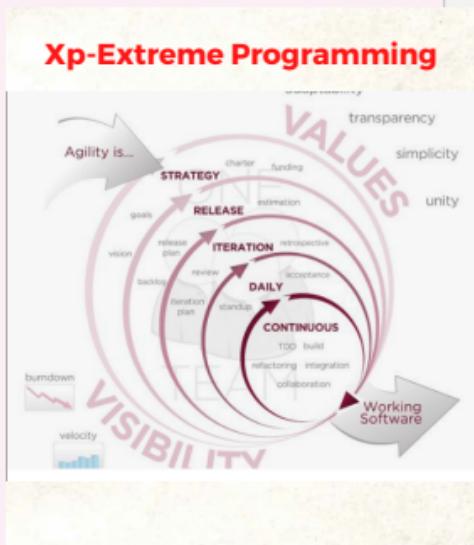
Planejamento



A cultura DevOps nasceu da necessidade de maior colaboração entre Dev (desenvolvedores) e Ops (operadores) de sistemas de Tecnologia da Informação.



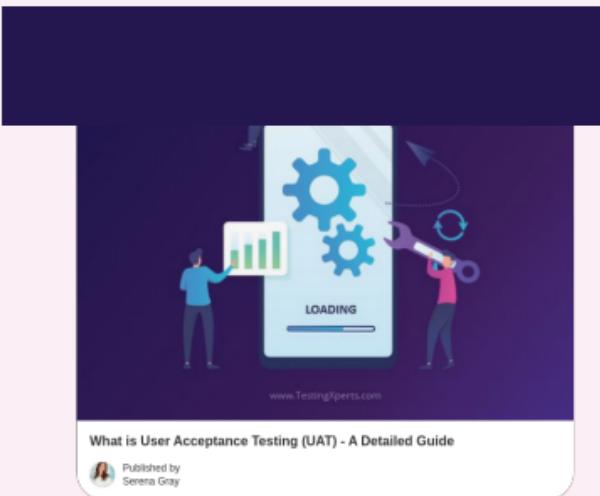
E esse alinhamento entre os setores surgiu na esteira do **desenvolvimento ágil**, empregando uma série de ferramentas.





O **planejamento** do DevOps geralmente é o primeiro estágio do DevOps. No entanto na prática, as equipes de software modernas trabalham em ciclos e os estágios tende a se repetir nos ciclos com mais ou menos intensidade de trabalho.

- Metas e Objetivos
 - Levantamento das dificuldades
 - Definição do objetivo
 - Definição de metas (SMART)*
- Equipe Multidisciplinar
 - Integração e treinamento
- Método / Ferramentas
 - Scrum, Kanban e Extreme Programming (XP)
 - Git, AWS, Azure, Jenkins, etc..

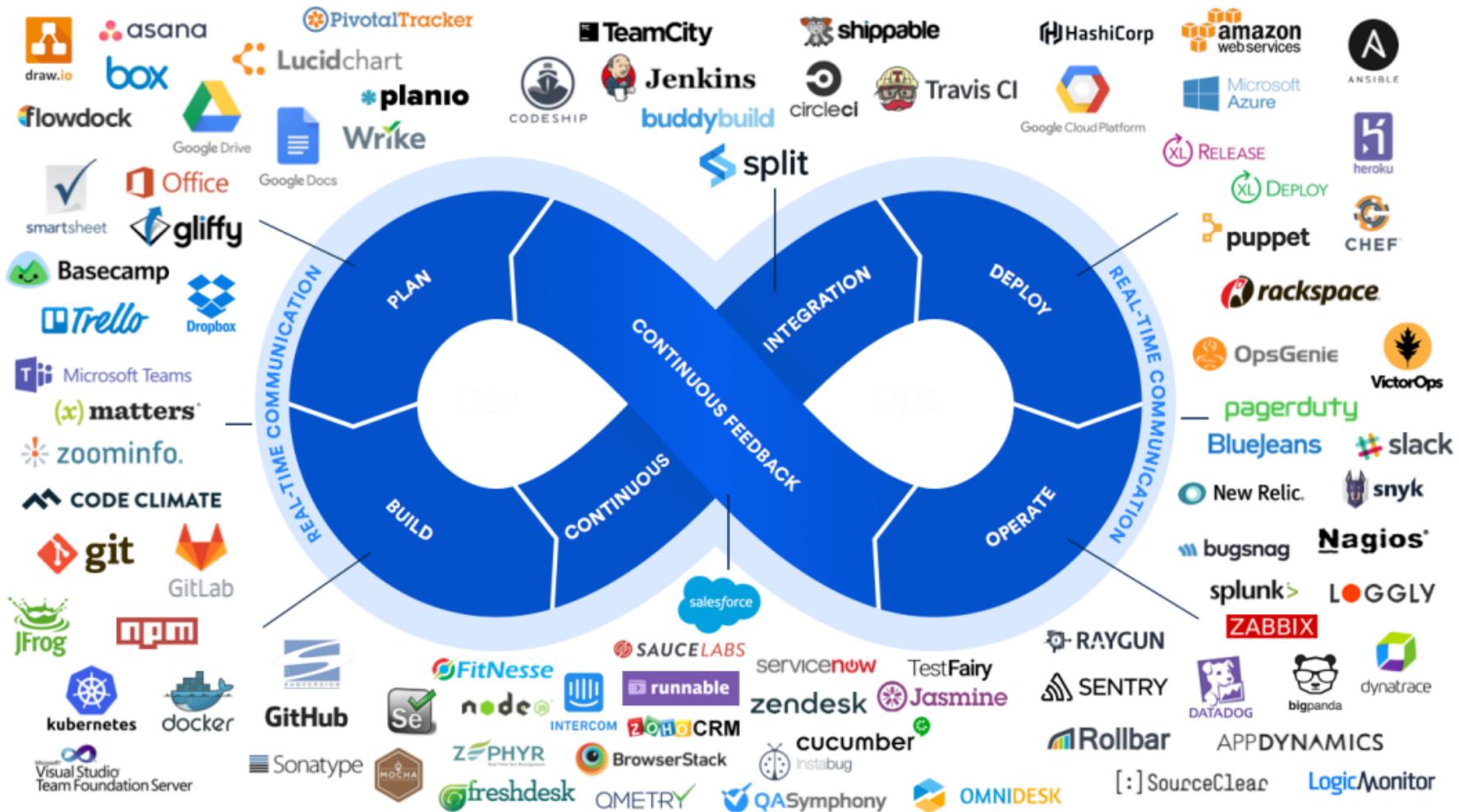


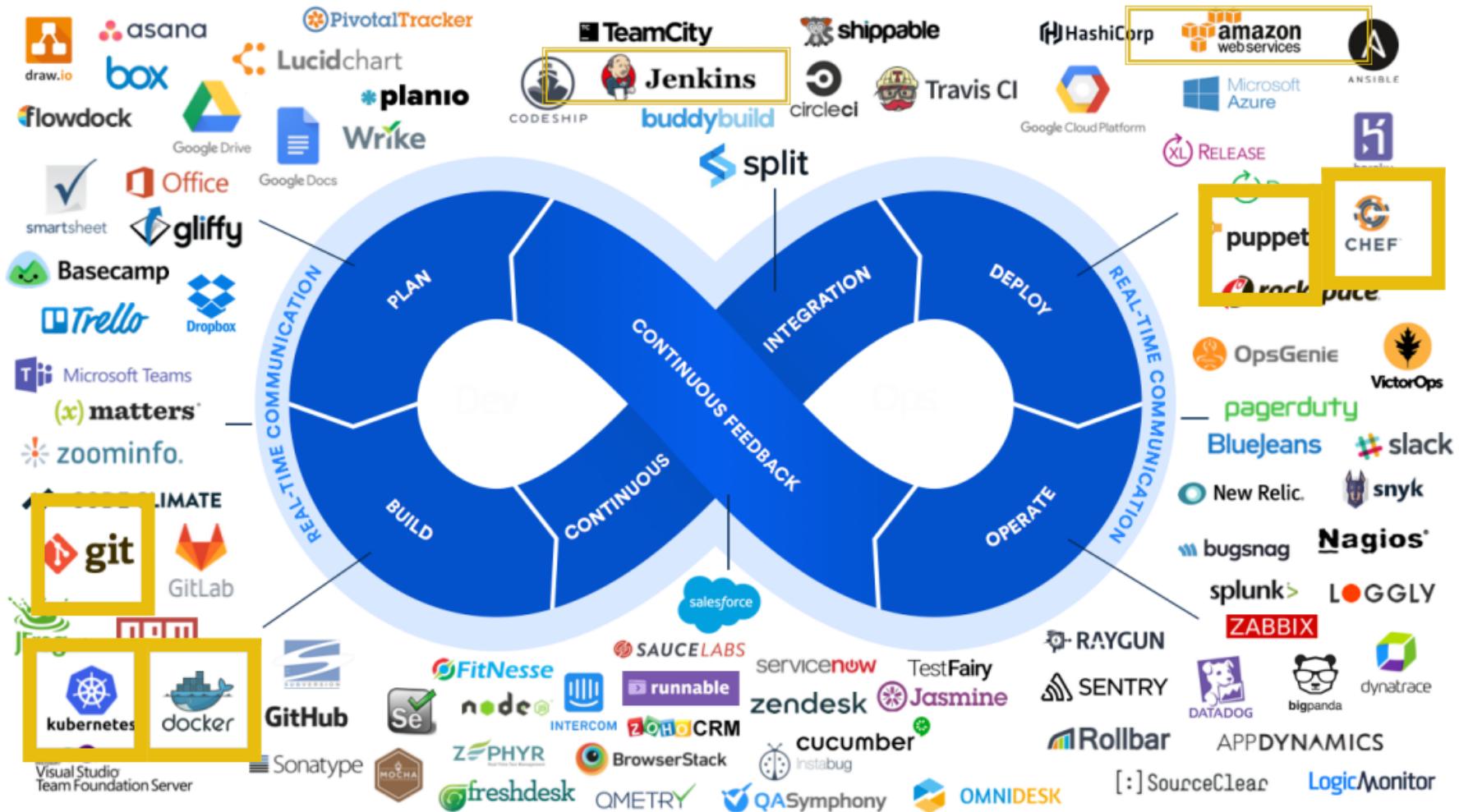
* S (Especificas), M (Mensuráveis), A (Atingíveis), R (Relevantes) e T (Temporais).



Segundo a Microsoft, os seguintes itens são importantes e ajudam a implementação de DEVOPS:

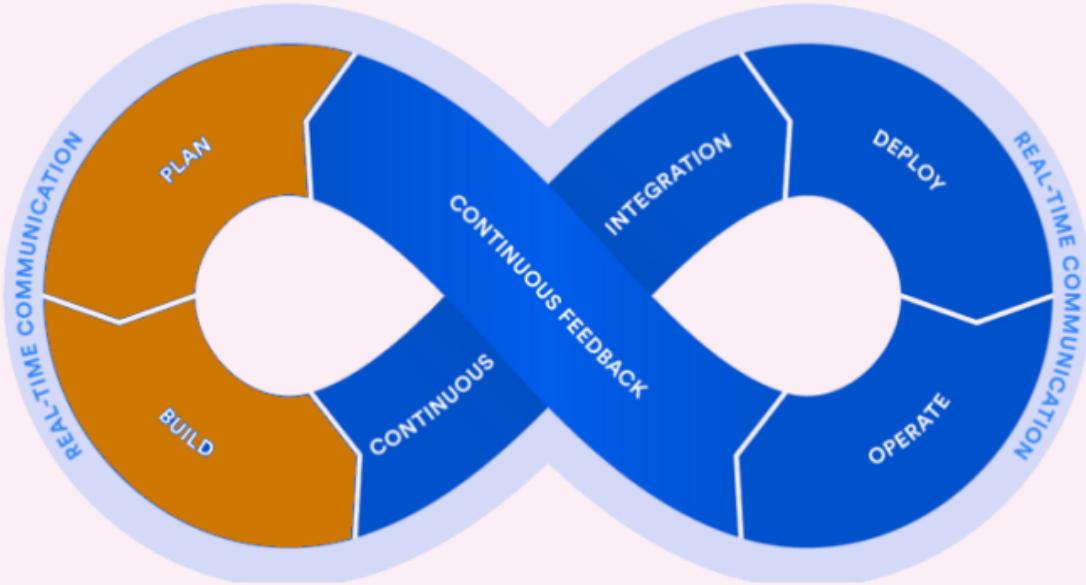
- Promover o alinhamento cultural e a autonomia.
- Alterar o foco dos indivíduos para as equipes.
- Criar novas estratégias de planejamento e aprendizagem.
- Implementar um modelo de trabalho com várias equipes.
- Aprimorar as práticas de qualidade de código.
- Promover transparência e responsabilidade.

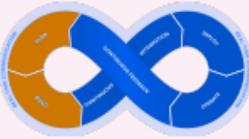




02

Build/Construção





Cloud Computing

Centros de dados disponíveis para muitos utilizadores pela Internet.

Nuvens em grande escala, predominantes hoje em dia, geralmente têm funções distribuídas em vários locais dos servidores centrais



Cloud



Infraestrutura

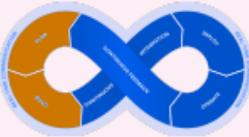
aws O que é a AWS?



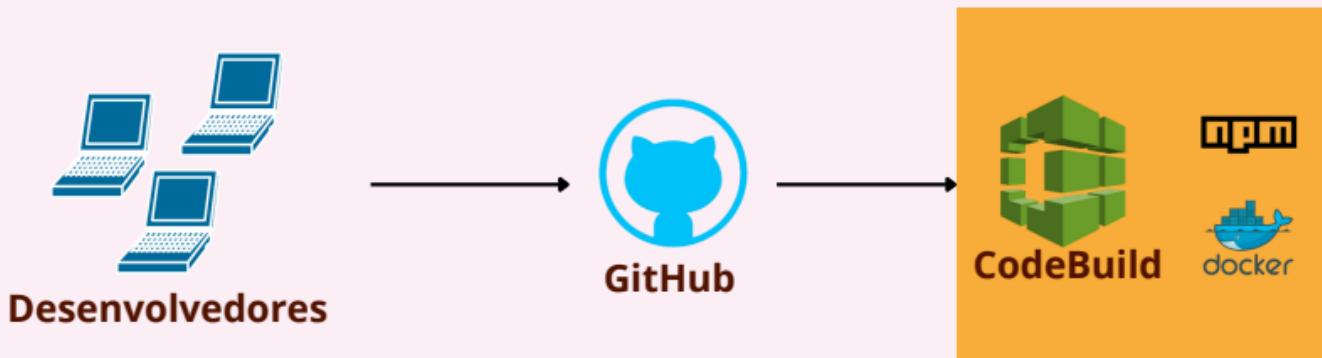
A graphic featuring a dark blue background with a white outline of a city skyline containing various buildings like a church, office towers, and a hospital. A large white "aws" logo with its signature yellow arrow is positioned above the city. A red YouTube play button icon is overlaid on a yellow hexagon icon in the center of the city. A light blue curved line arches over the top of the city.

O que é a AWS?

Watch on YouTube

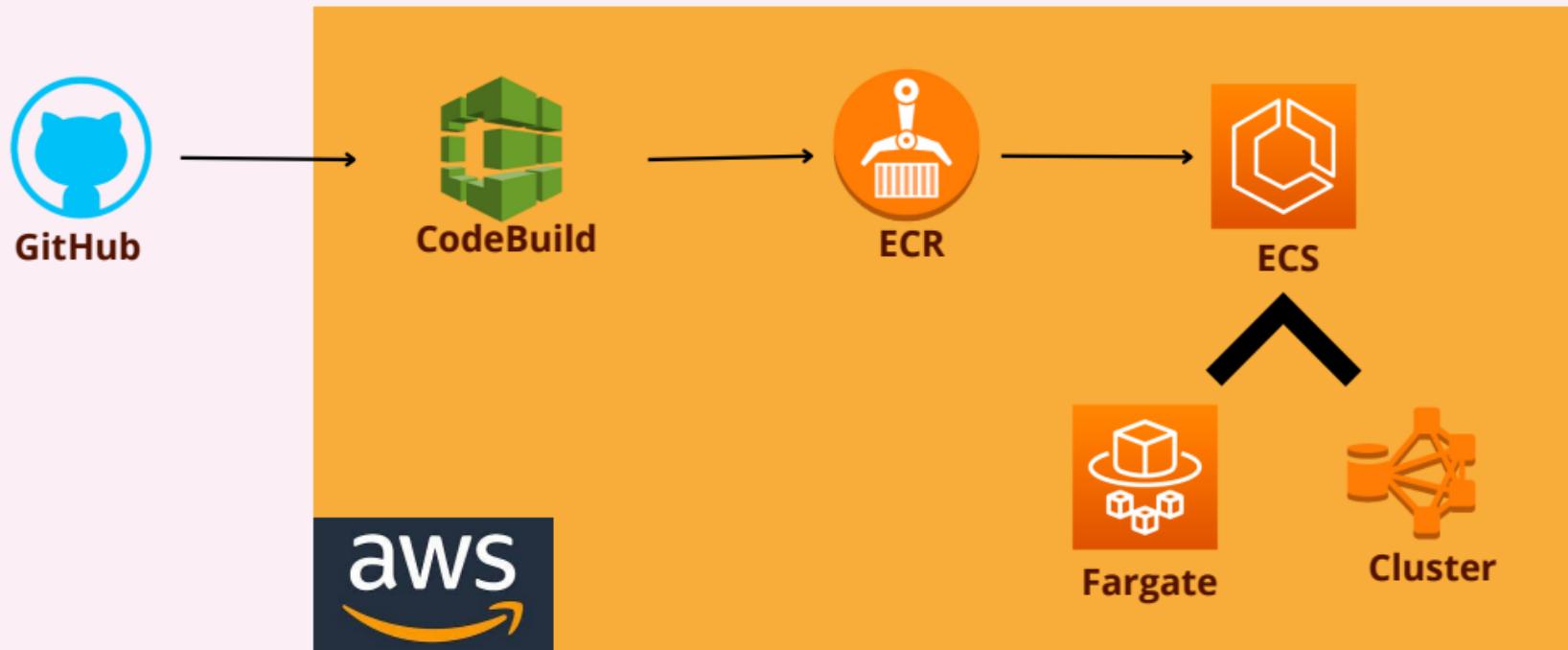


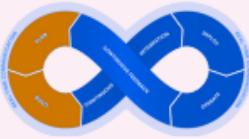
Na fase de **Build** são realizada as etapas integração, construção, versionamento, qualidade e compilação do código-fonte produzido. Muitas ferramentas, tais como Maven e Git (github, gitlab, etc..) são utilizadas para auxiliar na integração e monitoramento dos artefatos. Quando há quebra no **Build** devido a problema neste processo, todos os envolvidos deve ser informados, e os responsáveis precisam fazer as correções disponibilizar os códigos na esteira novamente.





Exemplo Build





Exemplo Build



GitHub

cassiowt/nodepipeline-devops



1 Contributor 0 Issues 0 Stars 0 Forks

cassiowt/nodepipeline-devops

Contribute to cassiowt/nodepipeline-devops development by creating an account on GitHub.

[GitHub](#)



Exemplo Build



Amazon ECR > Repositories > Create repository

Create repository

General settings

Visibility settings | [Info](#)

Choose the visibility setting for the repository.

Private

Access is managed by IAM and repository policy permissions.

Public

Publicly visible and accessible for image pulls.

Repository name

Provide a concise name. A developer should be able to identify the repository contents by the name.

489323066454.dkr.ecr.us-east-1.amazonaws.com/nodipeline-devops

19 out of 256 characters maximum (2 minimum). The name must start with a letter and can only contain lowercase letters, numbers, hyphens, underscores, periods and forward slashes.

Tag immutability | [Info](#)

Enable tag immutability to prevent image tags from being overwritten by subsequent image pushes using the same tag. Disable tag immutability to allow image tags to be overwritten.

Disabled

Once a repository is created, the visibility setting of the repository can't be changed.

Push commands for nodipeline-devops

[macOS / Linux](#) | Windows

Make sure that you have the latest version of the AWS CLI and Docker installed. For more information, see [Getting Started with Amazon ECR](#).

Use the following steps to authenticate and push an image to your repository. For additional registry authentication methods, including the Amazon ECR credential helper, see [Registry Authentication](#).

1. Retrieve an authentication token and authenticate your Docker client to your registry.

Use the AWS CLI:

```
aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin  
489323066454.dkr.ecr.us-east-1.amazonaws.com
```

Note: If you receive an error using the AWS CLI, make sure that you have the latest version of the AWS CLI and Docker installed.

2. Build your Docker image using the following command. For information on building a Docker file from scratch see the instructions [here](#). You can skip this step if your image is already built:

```
docker build -t nodipeline-devops .
```

3. After the build completes, tag your image so you can push the image to this repository:

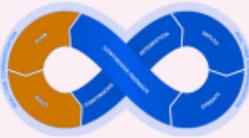
```
docker tag nodipeline-devops:latest 489323066454.dkr.ecr.us-east-1.amazonaws.com/nodipeline-devops:latest
```

4. Run the following command to push this image to your newly created AWS repository:

```
docker push 489323066454.dkr.ecr.us-east-1.amazonaws.com/nodipeline-devops:latest
```

X

Close



Exemplo Build



AWS CodeBuild

Developer Tools > CodeBuild > Build projects > Create build project

Create build project

Project configuration

Project name

A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.

Description - optional

Build badge - optional

 Enable build badge

Enable concurrent build limit - optional

Limit the number of allowed concurrent builds for this project.

 Restrict number of concurrent builds this project can start

► Additional configuration

tags

Build

Source

Add source

Source 1 - Primary

Source provider

Repository

 Public repository Repository in my GitHub account

Repository URL

https://github.com/<user-name>/<repository-name>

Connection status

You are connected to GitHub using a personal access token.

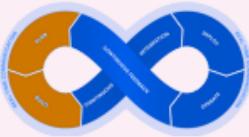
[Disconnect from GitHub](#)

Source version - optional [Info](#)

Enter a pull request, branch, commit ID, tag, or reference and a commit ID.

► Additional configuration

Git clone depth, Git submodules



Exemplo Build



Environment

Environment image

Managed image
Use an image managed by AWS CodeBuild

Custom image
Specify a Docker image

Operating system

Amazon Linux 2

(i) The programming language runtimes are now included in the standard image of Ubuntu 18.04, which recommended for new CodeBuild projects created in the console. See [Docker Images Provided by CodeBuild](#) for details.

Runtime(s)

Standard

Image

aws/codebuild/amazonlinux2-x86_64-standard:3.0

Image version

Always use the latest image for this runtime version

Privileged

Enable this flag if you want to build Docker images or want your builds to get elevated privileges

Service role

New service role
Create a service role in your account

Existing service role
Choose an existing service role from your account

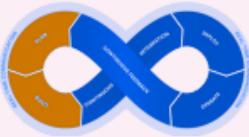
Role name

00-codebuild-nodepipeline-devops-service-role

Type your service role name

Additional configuration

Timeout, certificate, VPC, compute type, environment variables, file systems



Exemplo Build



AWS CodeBuild



AWS IAM

IAM > Roles

Roles (7) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.



Delete

Create role

Search

< 1 >

| <input type="checkbox"/> | Role name | Trusted entities |
|--------------------------|---|---|
| <input type="checkbox"/> | 00-codebuild-nodepipeline-devops-service-role | AWS Service: codebuild |
| <input type="checkbox"/> | AWSServiceRoleForApplicationAutoScaling_DynamoDBTable | AWS Service: dynamodb.application-autoscaling |
| <input type="checkbox"/> | AWSServiceRoleForAutoScaling | AWS Service: autoscaling (Service-Linked Role) |
| <input type="checkbox"/> | AWSServiceRoleForElasticLoadBalancing | AWS Service: elasticloadbalancing (Service-Linked Role) |
| <input type="checkbox"/> | AWSServiceRoleForSupport | AWS Service: support (Service-Linked Role) |

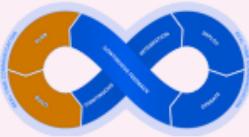
Attach policy to 00-codebuild-nodepipeline-devops-service-role

Current permissions policies (1)

Other permissions policies (824)

Filter policies by property or policy name and press enter.
2 matches

| <input type="checkbox"/> | Policy name <small>(Required)</small> | Type |
|--------------------------|---|-------------|
| <input type="checkbox"/> | AmazonEC2ContainerRegistryPowerUser | AWS managed |
| <input type="checkbox"/> | AmazonElasticContainerRegistryPublicPowerUser | AWS managed |



Exemplo Build



AWS ECS



AWS Cluster

New ECS Experience
[Tell us what you think](#)

Amazon ECS

Clusters

Task Definitions

Account Settings

Amazon EKS

Clusters

Amazon ECR

Repositories

AWS Marketplace

Discover software

Subscriptions

The new Amazon ECS console will become the default for all users starting 2023. You can opt-in to the new console today to use the new simplified workflows for deploying tasks and services, dark mode, task definition JSON editor, and new ECS features. You can continue to use the classic console for any unsupported features in the new experience.

Clusters

An Amazon ECS cluster is a regional grouping of one or more container instances on which you can run task requests. Each account receives a default cluster the first time you use the Amazon ECS service. Clusters may contain more than one Amazon EC2 instance type.

For more information, see the [ECS documentation](#).

[Create Cluster](#) [Get Started](#)

View [list](#) [card](#) [view all](#) [filter](#)

No clusters found

[Get Started](#)



Exemplo Build



AWS ECS



AWS Cluster

Select cluster template

The following cluster templates are available to simplify cluster creation. Additional configuration and integrations can be added later.

Networking only ?

Resources to be created:

Cluster

VPC (optional)

Subnets (optional)

For use with either AWS Fargate (Windows/Linux) or with External instance capacity.

EC2 Linux + Networking

Resources to be created:

Cluster

VPC

Subnets

EC2 Windows + Networking

Resources to be created:

Cluster

VPC

Subnets

Auto Scaling group with Windows AMI

*Required

Configure cluster

Cluster name*

nodepipeline



Networking

Create a new VPC for your cluster to use. A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Fargate tasks.

Create VPC Create a new VPC for this cluster

Tags

Key

Add key

Value

Add value

CloudWatch Container Insights

CloudWatch Container Insights is a monitoring and troubleshooting solution for containerized applications and microservices. It collects, aggregates, and summarizes compute utilization such as CPU, memory, disk, and network; and diagnostic information such as container restart failures to help you isolate issues with your clusters and resolve them quickly. [Learn more](#)

CloudWatch Container Insights Enable Container Insights

*Required

Cancel

Previous

Create



Exemplo Build



AWS ECS



AWS Fargate

Run Task

Select the cluster to run your task definition on and the number of copies of that task to run. To apply click Advanced Options.

Launch type FARGATE EC2 EXTERNAL ⓘ

AWS Fargate is migrating service quotas from the current Amazon ECS task count-based quotas to vCPU-based quotas. [To learn more, refer to the AWS Fargate FAQs.](#)

Platform version LATEST ⓘ

Cluster nodepipeline ⓘ

Number of tasks 1

Task Group ⓘ

VPC and security groups

VPC and security groups are configurable when your task definition uses the awsvpc network mode.

Operating system family Linux

Task Definition Family nodepipelinea-devops Edit

Revision 1 (latest)

Platform version LATEST ⓘ

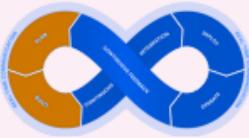
Cluster VPC* vpc-071f33b08b0885ac9 (172.31.0.0/16) ⓘ

Subnets* ⓘ

- subnet-04f3fe52b81ce7e5f (172.31.0.0/20) - us-east-1a assign ipv6 on creation: Disabled
- subnet-0ecbd8ef337782348 (172.31.16.0/20) - us-east-1c assign ipv6 on creation: Disabled

Security groups* nodepi-9608 Edit ⓘ

Auto-assign public IP ENABLED ⓘ



Exemplo Build



AWS ECS



AWS Fargate

Created tasks successfully
Task Ids : ["nodepipeline/dfd14a8e8c9f42eaa2249e3134ce5578"]

Clusters > nodepipeline

Cluster : nodepipeline

Get a detailed view of the resources on your cluster.

Cluster ARN arn:aws:ecs:us-east-1:489323066454:cluster/nodepipeline

Status ACTIVE

Registered container instances 0

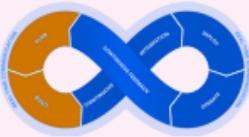
Pending tasks count 1 Fargate, 0 EC2, 0 External

Running tasks count 0 Fargate, 0 EC2, 0 External

Active service count 0 Fargate, 0 EC2, 0 External

Draining service count 0 Fargate, 0 EC2, 0 External

[Update Cluster](#) [Delete Cluster](#)



Exemplo Build



AWS ECS



AWS Fargate

Services Tasks ECS Instances Metrics Scheduled Tasks Tags Capacity Providers

Last updated on March 5, 2023 9:43:20 PM (0m ago) [Edit](#) [Details](#)

Desired task status: [Running](#) Stopped

Filter in this page Launch type ALL

| <input type="checkbox"/> | Task | Task defin... | Container... | Last statu... | Desired st... | Started at... | Started B... | Group | Launch ty... | Platform ... |
|--------------------------|--------------|-----------------|--------------|---------------|---------------|---------------|--------------|---------------|--------------|--------------|
| <input type="checkbox"/> | dfd14a8e8... | nodepipeline... | -- | RUNNING | RUNNING | 2023-03-0... | | family:nod... | FARGATE | 1.4.0 |

Clusters > nodepipeline > Task: dfd14a8e8c9f42eaa2249e3134ce5578

Task : dfd14a8e8c9f42eaa2249e3134ce5578

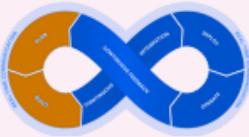
Details Tags Logs

Last updated on Mar

Filter logs

Timestamp (UTC+00:00) Message

| Timestamp (UTC+00:00) | Message |
|--|--|
| 2023-03-05 21:42:59 | listening on port:80 |
| | listening on port:80 |
| 2023-03-05 21:42:59 | > devops-nodejs-app-example@1.0.0 start /ecs-app |
| > devops-nodejs-app-example@1.0.0 start /ecs-app | |
| 2023-03-05 21:42:59 | > node server.js |
| | > node server.js |



Exemplo Build

Home About

You are looking at this page on ip-172-31-15-176.ec2.internal.

This is great, application works ! Click on About Tab to load the other page. The pages are divided in four parts head, header, body, footer. The body of every page will change according to page/data requested.

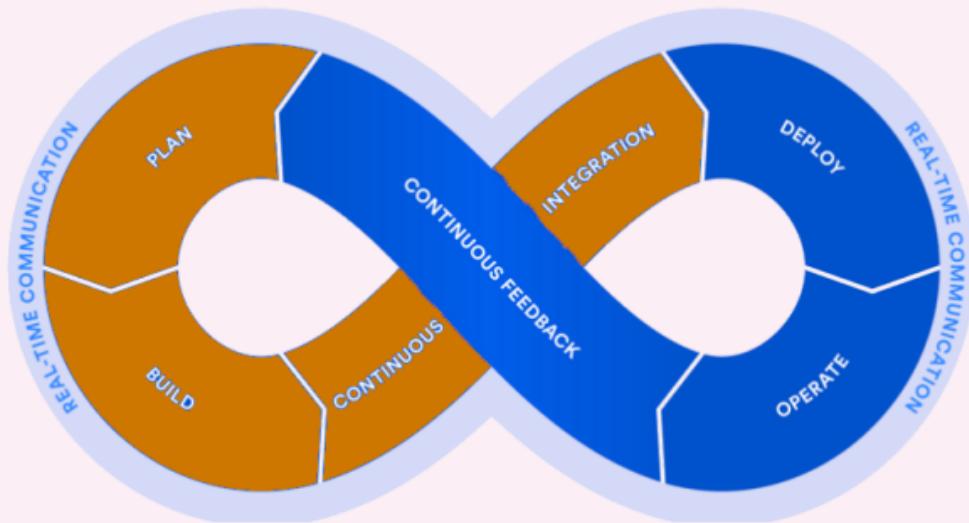
Welcome to templating using EJS.

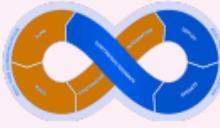
© Copyright 2023 - CAWT

02

CI/CD

Continuous Integration/ Continuous Delivery





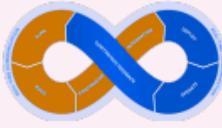
CI / CD

Projetos de software podem ser desenvolvidos de forma modular.

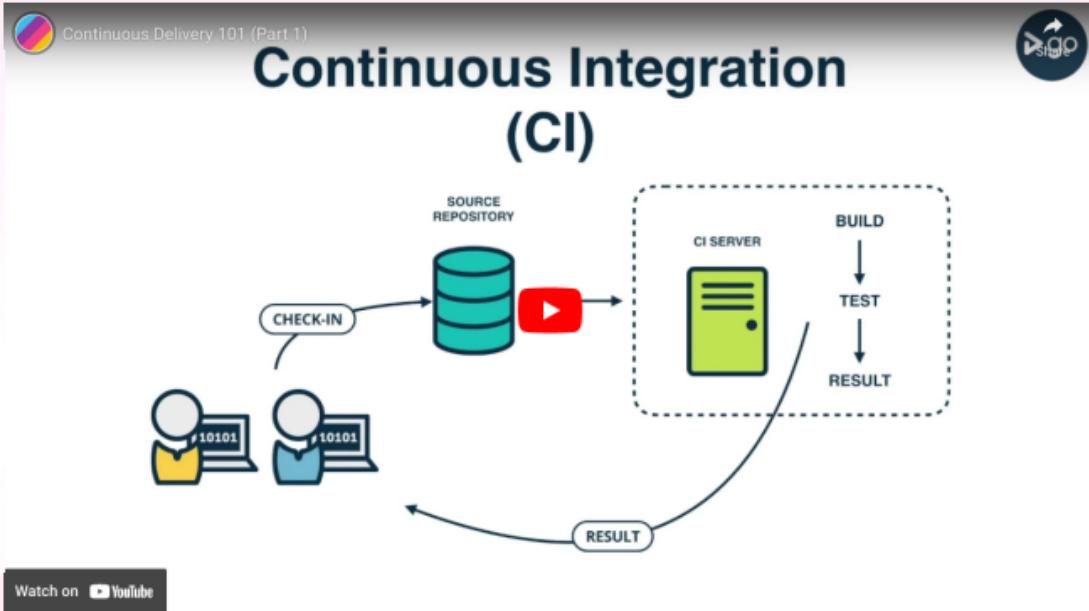
- Cada desenvolvedor implementa e testa a sua parte
- Necessidade de testes de integração no final de cada ciclo de desenvolvimento (ex: sprints, milestones)
- Problema: ciclos muito longos, erros demoram a aparecer

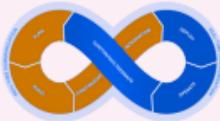
Integração contínua é a prática de automatizar o build, teste e o deploy, de todos os componentes "continuamente".

CI / CD



Ken Mugrage
Technology Evangelist at Thoughtworks





CI / CD - Combinação das seguintes técnicas

Versionamento de código

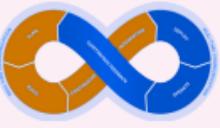
- git, svn, cvs, etc.

Automatização de build/compilação

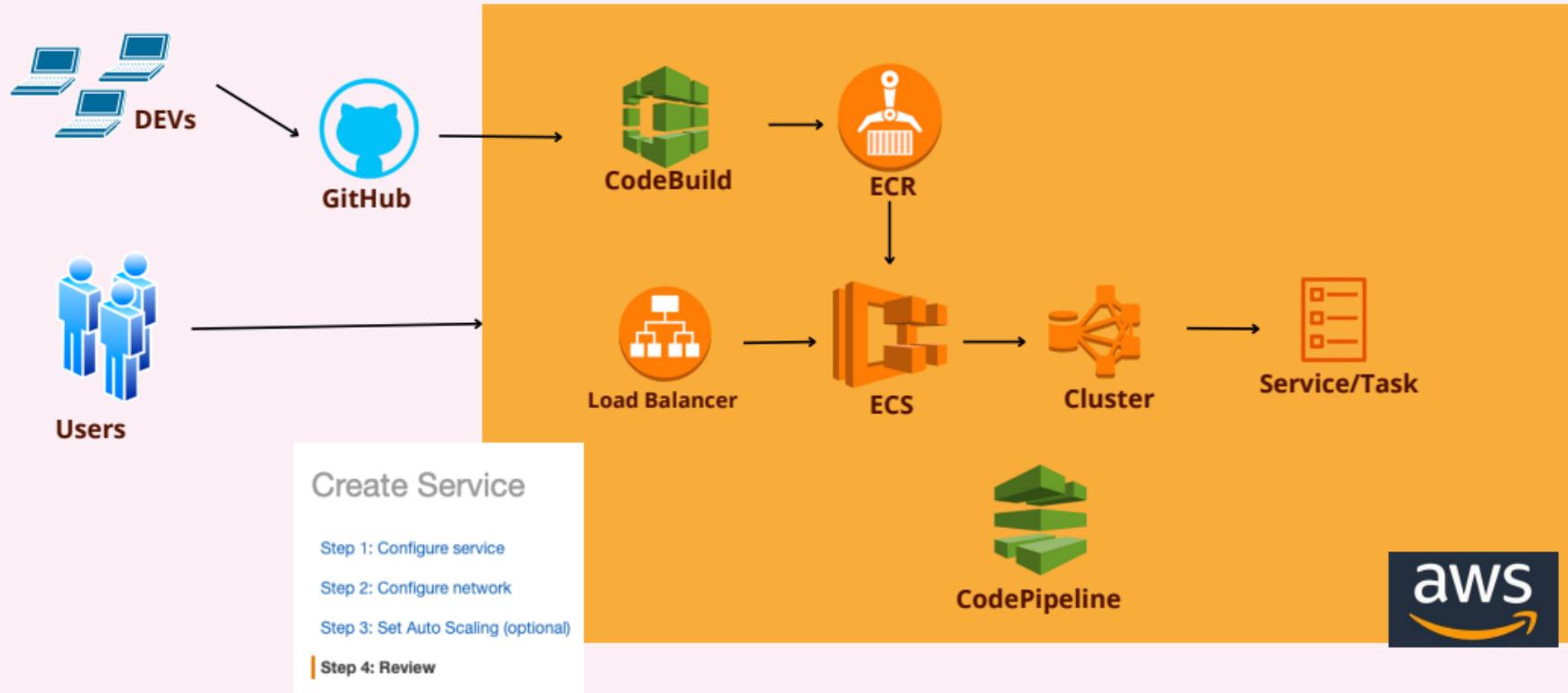
- NPM para javascript
- Autotools/make para C/C++
- Ant e Maven para Java
- Nant e MSBuild para .net

Automação de testes

- Unitários - Mocha, Chai (js), JUnit(java),
- Funcionais- Cypress (js), Selenium (java),
- Desempenho/stress - JMeter



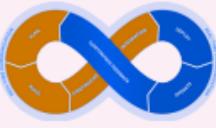
CI / CD



CI / CD

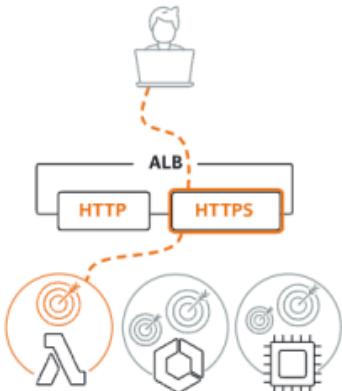


Load Balancer



Load balancer types

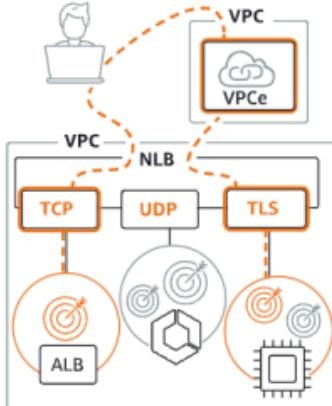
Application Load Balancer [Info](#)



Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.

[Create](#)

Network Load Balancer [Info](#)



Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.

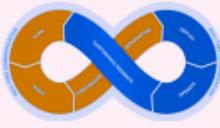
[Create](#)

Gateway Load Balancer [Info](#)



Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.

[Create](#)



CI / CD



Load Balancer

Create Application Load Balancer Info

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

▶ How Elastic Load balancing works

Basic configuration

Load balancer name

Name must be unique within your AWS account and cannot be changed after the load balancer is created.

node-app

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme Info

Scheme cannot be changed after the load balancer is created.

Internet-facing

An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)

Internal

An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type Info

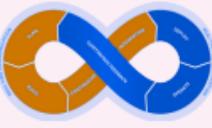
Select the type of IP addresses that your subnets use.

IPv4

Recommended for internal load balancers.

Dualstack

Includes IPv4 and IPv6 addresses.



CI / CD



Load Balancer

Listeners and routing Info

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80

[Remove](#)

Protocol

Port

Default action

[Info](#)

HTTP

:

80

Forward to

Select a target group

▼



1-65535

[Create target group](#)

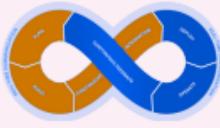
Listener tags - optional

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

[Add listener tag](#)

You can add up to 50 more tags.

[Add listener](#)



CI / CD - Target groups



Load Balancer

Target group name

tg-node-app

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol

Port

HTTP

:

80

IP address type

Only targets with the indicated IP address type can be included in this target group.

IPv4

IPv6

VPC

Select the VPC that hosts the load balancer. Only VPCs that support the IP address type selected above are available in this list. On the [Register targets](#) page, you can register IP addresses from this VPC, or from private IP addresses located outside of this load balancer's VPC (such as a peered VPC, EC2-Classic, or on-premises targets that are reachable over Direct Connect or VPN).

-

vpc-071f33b08b0885ac9

IPv4: 172.31.0.0/16

Protocol version

HTTP1

Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

HTTP2

Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.

gRPC

Send requests to targets using gRPC. Supported when the request protocol is gRPC.



Load Balancer

EC2 > Load balancers > node-app > Create Application Load Balancer

Create Application Load Balancer



Suggested next steps

- Review, customize, or enable attributes for your load balancer and listeners using the **Description** and **Listeners** tabs within [node-app](#).
- Discover other services that you can integrate with your load balancer. Visit the **Integrated services** tab within [node-app](#).

[View load balancer](#)

Load balancers (1)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.



[Actions ▾](#)

[Create load balancer](#)

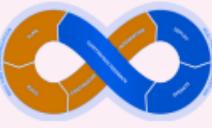


Filter by property or value

< 1 >



| <input type="checkbox"/> | Name | DNS name | State | VPC ID | Availability |
|--------------------------|--------------------------|---|--------------|-----------------------|----------------|
| <input type="checkbox"/> | node-app | node-app-229560.sa-east-... | Provisioning | vpc-0599ccdcf240a1978 | 3 Availability |



CI / CD



Clusters > nodepipeline

Cluster : nodepipeline

Get a detailed view of the resources on your cluster.

Update Cluster **Delete Cluster**

Cluster ARN arn:aws:ecs:us-east-1:489323066454:cluster/nodepipeline
Status ACTIVE

Registered container instances 0

Pending tasks count 0 Fargate, 0 EC2, 0 External

Running tasks count 0 Fargate, 0 EC2, 0 External

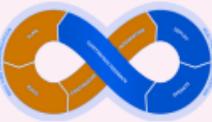
Active service count 0 Fargate, 0 EC2, 0 External

Draining service count 0 Fargate, 0 EC2, 0 External

Services **Tasks** **ECS Instances** **Metrics** **Scheduled Tasks** **Tags** **Capacity Providers**

Create **Update** **Delete** **Actions** Last updated on March 6, 2023 3:52:58 PM (0m ago) **Filter in this page** **Launch type** ALL **Service type** ALL

| <input type="checkbox"/> | Service Name | Status... | Service... | Task D... | Desire... | Runni... | Launc... | Platfor... |
|--------------------------|--------------|-----------|------------|-----------|-----------|----------|----------|------------|
| No results | | | | | | | | |



CI / CD



Configure service

A service lets you specify how many copies of your task definition to run and maintain in a cluster. You can optionally use an Elastic Load Balancing load balancer to distribute incoming traffic to containers in your service. Amazon ECS maintains that number of tasks and coordinates task scheduling with the load balancer. You can also optionally use Service Auto Scaling to adjust the number of tasks in your service.

Launch type FARGATE ?

AWS Fargate is migrating service quotas from the current Amazon ECS task count-based quotas to vCPU-based quotas. To learn more, refer to the AWS Fargate FAQs.

- EC2
- EXTERNAL

[Switch to capacity provider strategy](#)

Operating system family

Linux ?

Task Definition

Family

nodepipeline-devops ?

Revision

1 (latest) ?

?

Deployments

Choose a deployment option for the service.

Deployment type*

Rolling update ?

Blue/green deployment (powered by AWS CodeDeploy) ?

This sets AWS CodeDeploy as the deployment controller for the service. A CodeDeploy application and deployment group are created automatically with **default settings** for the service. To change to the rolling update deployment type after the service has been created, you must re-create the service and select the "rolling update" deployment type.

Task tagging configuration

Enable ECS managed tags ?

Propagate tags from

Do not propagate ?

Tags ?

Key

Add key

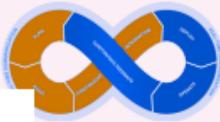
Value

Add value

*Required

Cancel

Next step



CI / CD



- Load balancer type***
- None
Your service will not use a load balancer.
 - Application Load Balancer
Allows containers to use dynamic host port mapping (multiple tasks allowed per container instance). Multiple services can use the same listener port on a single load balancer with rule-based routing and paths.
 - Network Load Balancer
A Network Load Balancer functions at the fourth layer of the Open Systems Interconnection (OSI) model. After the load balancer receives a request, it selects a target from the target group for the default rule using a flow hash routing algorithm.
 - Classic Load Balancer
Requires static host port mappings (only one task allowed per container instance); rule-based routing and paths are not supported.

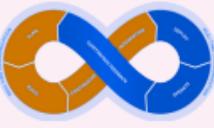
Service IAM role Task definitions that use the awsvpc network mode use the AWSServiceRoleForECS service-linked role, which is created for you automatically. [Learn more.](#)

Load balancer name

Container to load balance

Container name : port **Add to load balancer**

CI / CD



Container to load balance

nodepipeline-devops : 80

Remove

Production listener port* 80:HTTP

Production listener protocol* HTTP

Target group name tg-node-app

Target group protocol HTTP

Target type ip

Path pattern / Evaluation order default

Health check path /

Additional health check

Set Auto Scaling (optional)

Automatically adjust your service's desired count up and down within a specified range in response to CloudWatch alarms. You can modify your Service Auto Scaling configuration at any time to meet the needs of your application.

Service Auto Scaling Do not adjust the service's desired count

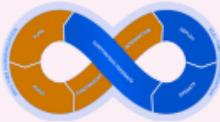
Configure Service Auto Scaling to adjust your service's desired count

*Required

Cancel

Previous

Next step



CI / CD

Acesso via loadbalance!

home page of sample node app x +

Mostrar aplicativos

Não seguro | node-app-695745312.us-east-1.elb.amazonaws.com/index

Atualizar :

Apps JS+ CAWT DV puc Bitly MCC DI Bitly MCC DI

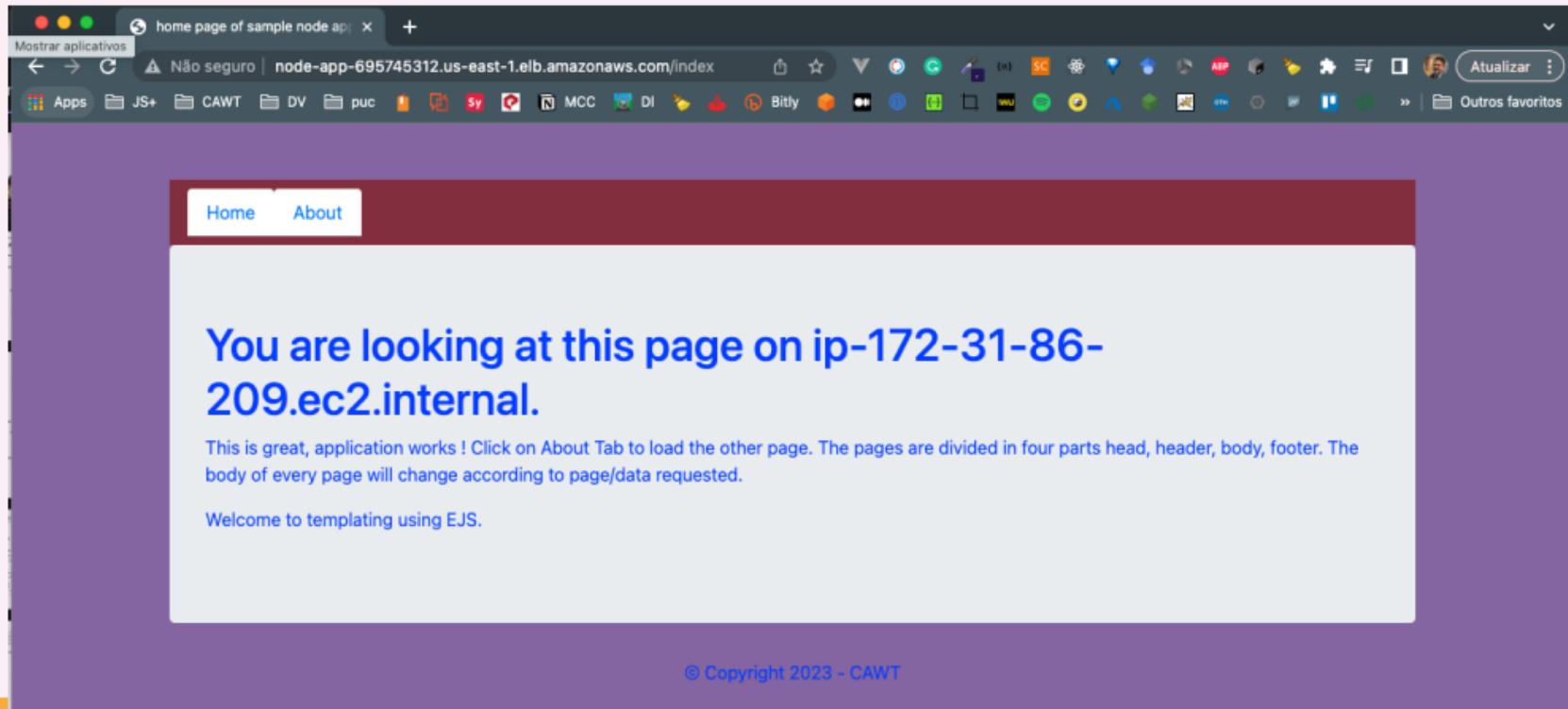
Home About

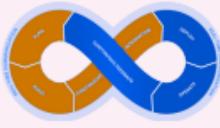
You are looking at this page on ip-172-31-86-209.ec2.internal.

This is great, application works ! Click on About Tab to load the other page. The pages are divided in four parts head, header, body, footer. The body of every page will change according to page/data requested.

Welcome to templating using EJS.

© Copyright 2023 - CAWT





CI / CD



CodePipeline

Developer Tools > CodePipeline

Step 1

Choose pipeline settings

Step 2

Add source stage

Step 3

Add build stage

Step 4

Add deploy stage

Step 5

Review

Developer Tools > CodePipeline > Pipelines

Pipelines Info



Notify ▾

View history

Release change

Delete pipeline

Create pipeline



< 1 >



Name

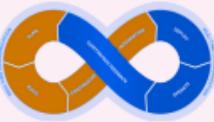
Most recent execution

Latest source revisions

Last executed

No results

There are no results to display.



CodePipeline

Choose pipeline settings Info

Pipeline settings

Pipeline name

Enter the pipeline name. You cannot edit the pipeline name after it is created.

nodepipeline

No more than 100 characters

Service role

 New service role

Create a service role in your account

 Existing service role

Choose an existing service role from your account

Role name

00-AWSCodePipelineServiceRole-us-east-1-nodepipeline

Type your service role name

- Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

► Advanced settings

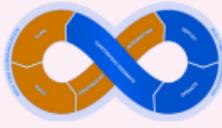
Cancel

Next

CI / CD



CodePipeline



Add source stage Info

Source

Source provider

This is where you stored your input artifacts.



AWS CodeCommit

Amazon ECR

Amazon S3

Bitbucket

GitHub (Version 1)

GitHub (Version 2)

GitHub Enterprise Server

Source provider

This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

GitHub (Version 2)



New GitHub version 2 (app-based) action

To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. [Learn more](#)

Connection

Choose an existing connection that you have already configured, or create a new one and then return to this task.

arn:aws:codestar-connections:us-east-1:489323066454:connection/117f5abi X

or

[Connect to GitHub](#)



Ready to connect

Your GitHub connection is ready for use.

Repository name

Choose a repository in your GitHub account.

cassiowrt/devops-nodepipeline X

<account>/<repository-name>

Branch name

Choose a branch of the repository.

main X

Change detection options

Start the pipeline on source code change

Automatically starts your pipeline when a change occurs in the source code. If turned off, your pipeline only runs if you start it manually or on a schedule.

Output artifact format

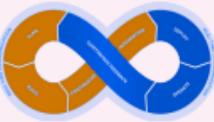
Choose the output artifact format.

CodePipeline default

AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include git metadata about the repository.

Full clone

AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full git clone. Only supported for AWS CodeBuild actions.

Add deploy stage Info

Deploy - optional

Deploy provider

Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.

Amazon ECS



Region

US East (N. Virginia)



Cluster name

Choose a cluster that you have already created in the Amazon ECS console. Or create a cluster in the Amazon ECS console and then return to this task.

nodepipeline



Service name

Choose a service that you have already created in the Amazon ECS console for your cluster. Or create a new service in the Amazon ECS console and then return to this task.

service-nodepipeline



Image definitions file - optional

Enter the JSON file that describes your service's container name and the image and tag.

imagedefinitions.json

Deployment timeout - optional

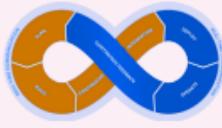
Enter the timeout in minutes for the deployment action.

Cancel

Previous

Skip deploy stage

Next

Add build stage Info**Build - optional****Build provider**

This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.

AWS CodeBuild

**Region**

US East (N. Virginia)

**Project name**

Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.

nodepipeline-devops



or

Create project

Environment variables - optional

Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. [Learn more](#)

[Add environment variable](#)**Build type** Single build

Triggers a single build.

 Batch build

Triggers multiple builds as a single execution.

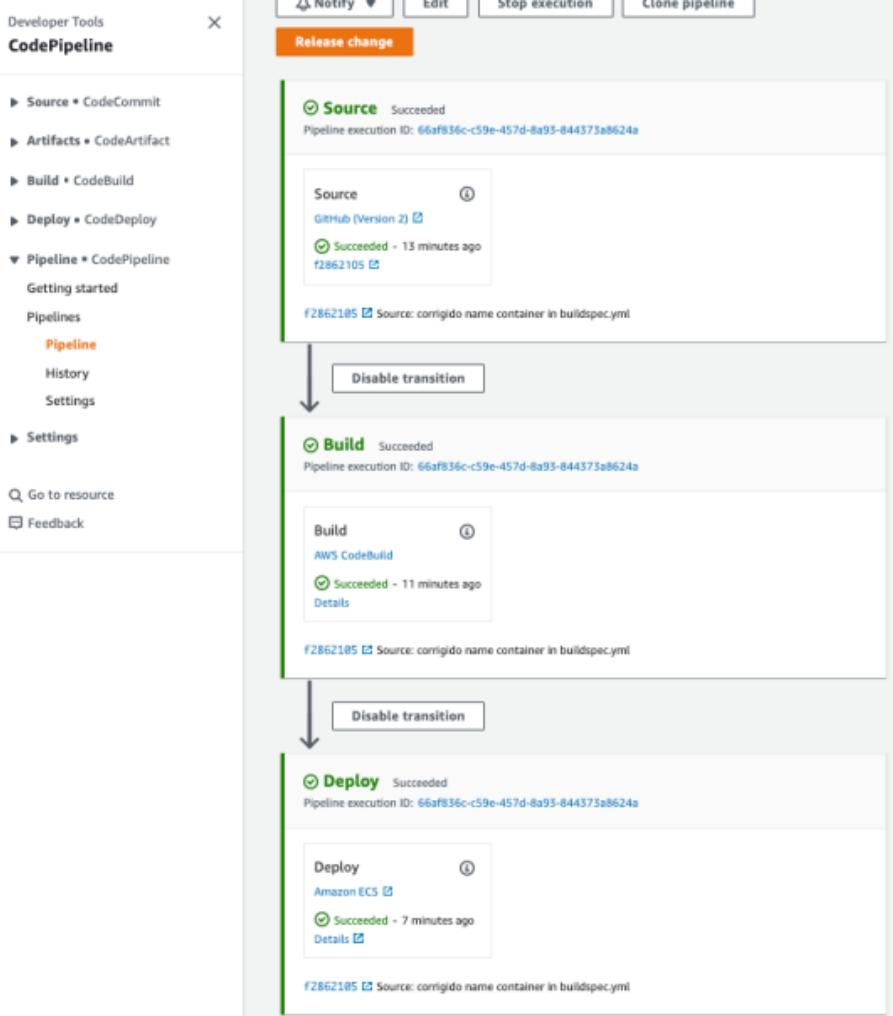
[Cancel](#)[Previous](#)[Skip build stage](#)[Next](#)

CI / CD



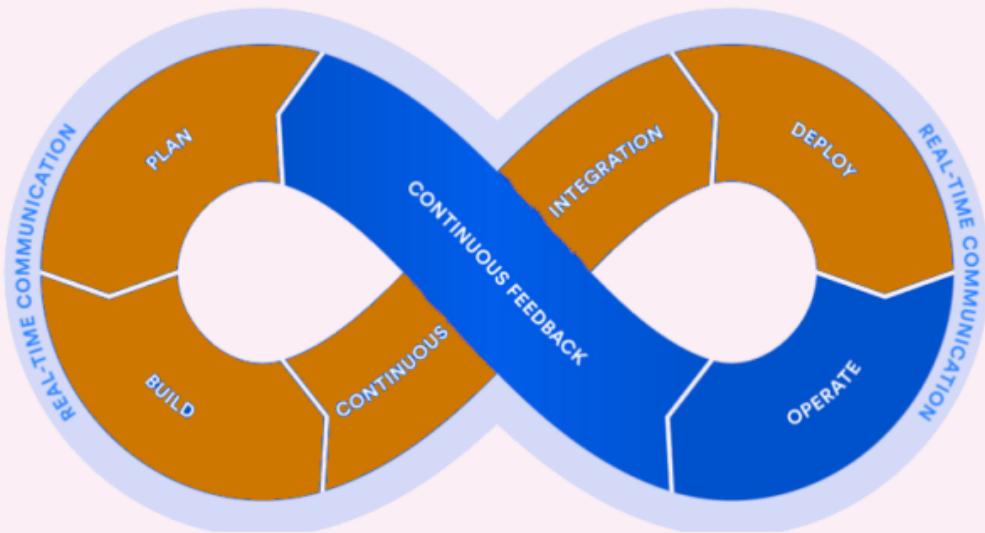
CodePipeline

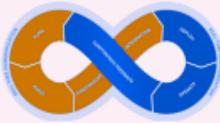
CI/CD



03

Deploy





CI / CD



CodePipeline

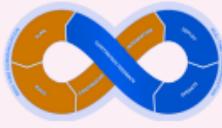


```
1 body class="container" style="
background-color: rgb(60, 113, 173);">
2
```

CI / CD



CodePipeline



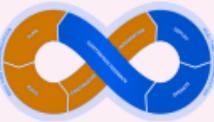
A screenshot of a web browser window displaying a sample Node.js application. The browser's title bar reads "home page of sample node app" and "Não seguro | node-app-69574531...". The address bar shows the URL. The browser toolbar includes various icons for file operations and extensions. Below the toolbar, a tab bar lists several items: Apps, JS+, CAWT, DV, puc, Sy, MCC, DI, Bitly, and others. The main content area features a dark red header bar with "Home" and "About" tabs. The body of the page contains the following text:

You are looking at this page on ip-
172-31-26-1.ec2.internal.

This is great, application works ! Click on About Tab to load the other page. The pages are divided in four parts head, header, body, footer. The body of every page will change according to page/data requested.

Welcome to templating using EJS.

CI / CD - Conceitos



CONTAINERS



ORCHESTRATION



ECS
Elastic Container
Service



EKS
Elastic Container
Kubernetes



SERVERLESS



Fargate

PUCRS online  uol edtech