

# Prova de Conceito do *Kubernetes*

## Introdução

### A Aplicação

A Aplicação, que leva o nome de “Diário de um Programador” gerencia postagens de blog, permitindo a criação e o compartilhamentos de artigos.

Após o login, o usuário tem acesso ao painel, onde é possível criar artigos com imagens na capa. As postagens são públicas e acessíveis ao demais usuários do sistema.

O sistema foi desenvolvido utilizando PHP, Laravel Framework e Bootstrap. A arquitetura utilizada na aplicação é o MVC (Model, View e Controller) e o gerenciador de banco de dados escolhido foi o MariaDB.

Funcionalidades do “Diário de um Programador”:

- Controle de acesso ao painel;
- Gerenciamento das categorias de artigos no painel;
- Gerenciamento dos artigos no painel;
- Gerenciamento das imagens de capa dos artigos;
- Exibição dos artigos nas páginas públicas.

### O Kubernetes

Kubernetes, conhecido pela forma abreviada “k8s”, é um gerenciador de contêineres feito para distribuir os contêineres de uma aplicação de forma automatizada. É um sistema robusto, que permite orquestrar aplicações na nuvem ou em um servidor.

De acordo com estudo de caso da Mercedes Benz, a migração para um projeto do Kubernetes permitiu a execução de atualizações contínuas, e diminuiu o tempo de deploy. Não somente esta, mas 78% das empresas descritas no relatório da CNCF o utilizam.

# Desenvolvimento

## Gerando imagens

```
# Aplicação
docker build . -t thunder8598/diario-de-um-programador

# Banco de dados
docker build . -t thunder8598/mariadb
```

## Enviando imagens ao Docker Hub

As imagens foram salvas no Docker Hub, repositório para imagens de contêineres.

```
# Aplicação
docker image push thunder8598/diario-de-um-programador:latest

# Banco de dados
docker image push thunder8598/mariadb:latest
```

## Configurando a aplicação no *Kubernetes*

1. Criando um *deployment* a partir de uma imagem Docker:

```
kubectl create deployment app --image=thunder8598/diario-de-um-programador:latest
```

2. Expondo o serviço através de uma porta:

```
kubectl expose deployment app --type=LoadBalancer --port=8000
```

3. Iniciando a virtualização e acessando o serviço através do comando:

```
minikube start
minikube service app
```

## Configurando o banco de dados no *Kubernetes*

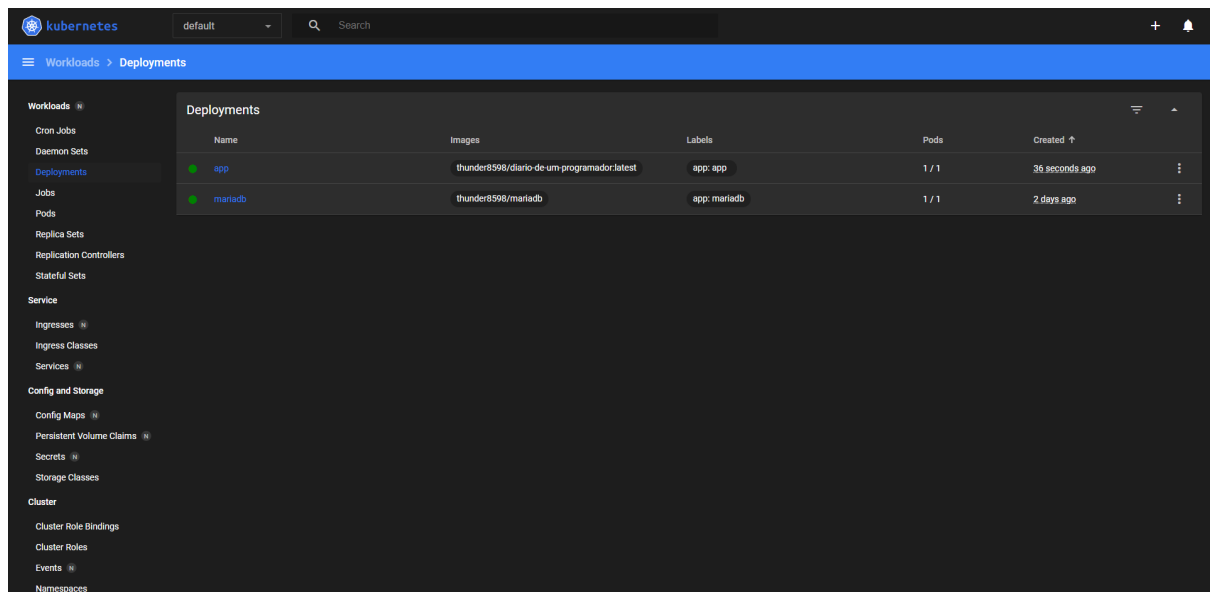
1. Criando um *deployment* a partir de uma imagem Docker:

```
kubectl create deployment mariadb --image=thunder8598/mariadb:latest
```

2. Expondo o serviço através de uma porta:

```
kubectl expose deployment mariadb --type=LoadBalancer --port=3306
```

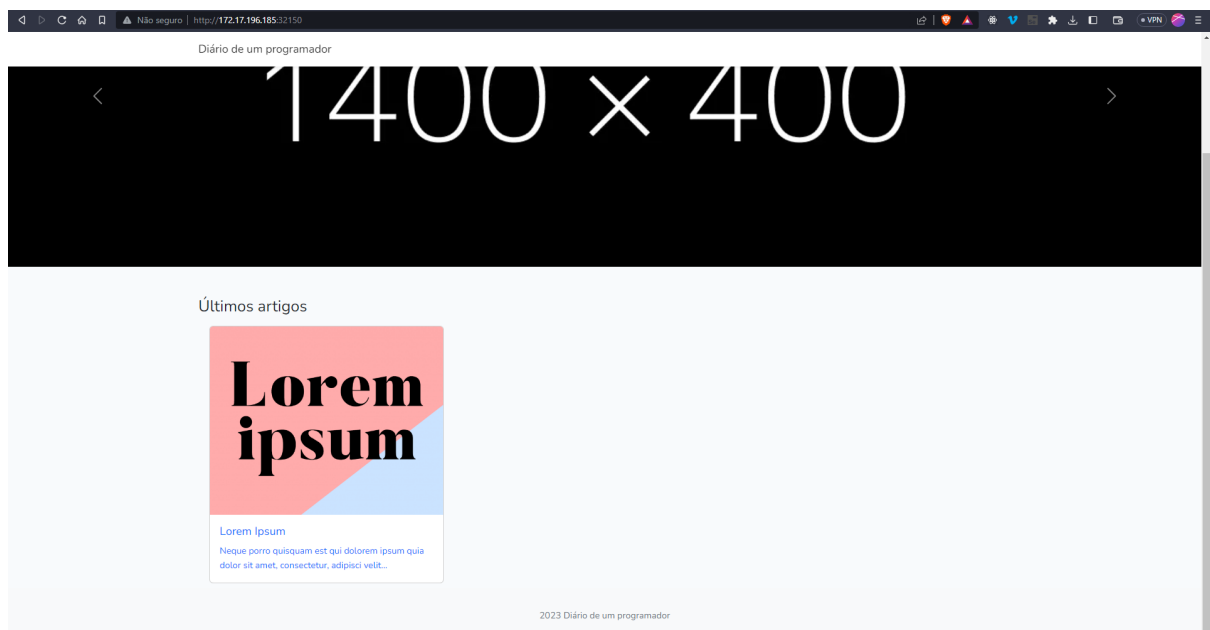
## Serviços ativos no *Kubernetes*



The screenshot shows the Kubernetes dashboard with the 'Deployments' section selected. A table lists the active deployments:

Name	Images	Labels	Pods	Created
app	thunder8598/diario-de-um-programador:latest	app: app	1 / 1	36 seconds ago
mariadb	thunder8598/mariadb	app: mariadb	1 / 1	2 days ago

## Aplicação



# Conclusão

A solução do Sk8, utilizada neste projeto pequeno, serve para aplicações de grande porte ou até para microsserviços.

Durante o deployment no K8s foi necessário enviar as imagens ao Docker Hub, para então as utilizar no sistema, pois localmente o processo apresentava erros.