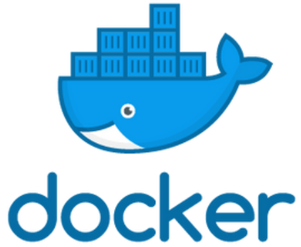
del.tech



Conteinerização com Docker

Wesley Alves

del.tech

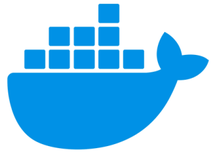
Problemática

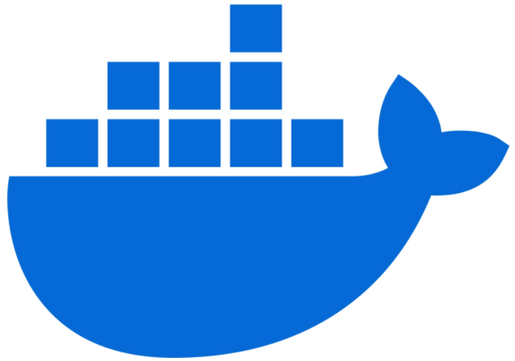
“Na minha máquina funciona!”

“Preciso de um banco de dados para testes”

Possível solução: Virtualização

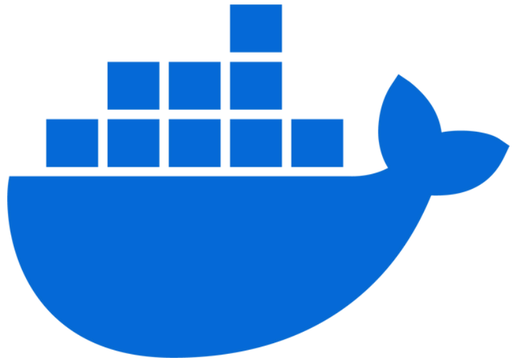
del.tech

01. O que é  ?



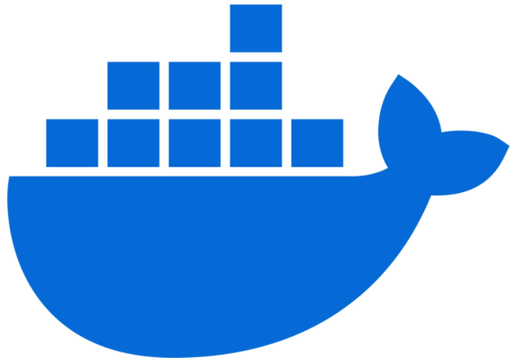
Docker

- Plataforma de código aberto que permite criar, executar e gerenciar **contêineres**.
- Um contêiner é um pacote **leve** e **portátil** que inclui tudo o que uma aplicação precisa para rodar (código, bibliotecas, configs, etc).
- Os contêineres funcionam de maneira consistente em qualquer sistema operacional que suporte Docker, garantindo a portabilidade.



Docker vs VMs

- VMs incluem um *sistema operacional completo* (e.g., Windows, Linux) e são gerenciadas por um hypervisor.
- VMs consomem mais recursos (RAM, CPU) porque cada instância inclui um sistema operacional.
- Contêineres *compartilham o kernel do sistema operacional host*, sendo mais leves e rápidos.
- Contêineres são mais eficientes, pois usam apenas o que é necessário.

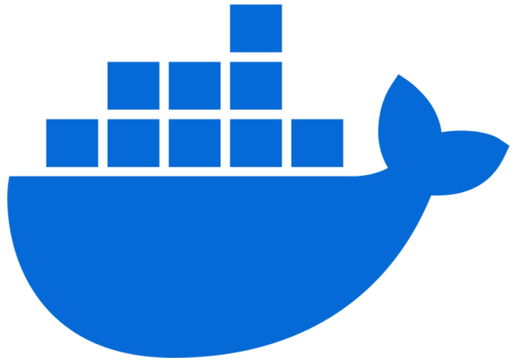


Casos de Uso

- **Facilitar testes com infraestrutura em ambiente local.**
- **Padronizar ambientes entre desenvolvedores e times.**
- **Usado em pipelines de CI/CD para testar, empacotar e implantar aplicações.**
- **Cada serviço em uma arquitetura de microsserviços pode ser empacotado em um contêiner, facilitando escalabilidade e manutenção.**

del.tech

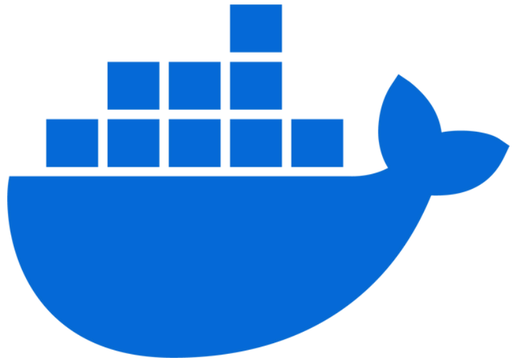
02. Conceitos Essenciais



Imagens

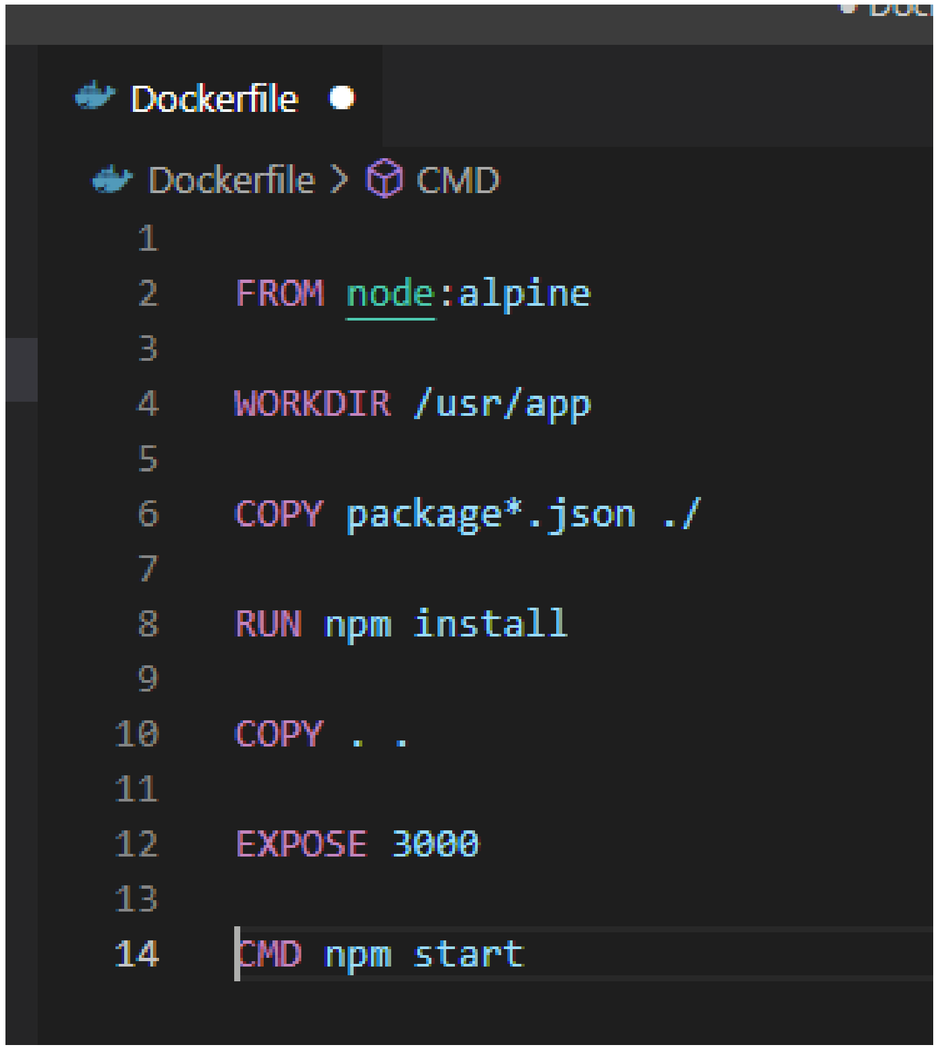
- **É como um modelo ou *blueprint* que define tudo o que um contêiner precisa para ser executado: o sistema operacional base, bibliotecas, dependências e o código da aplicação.**

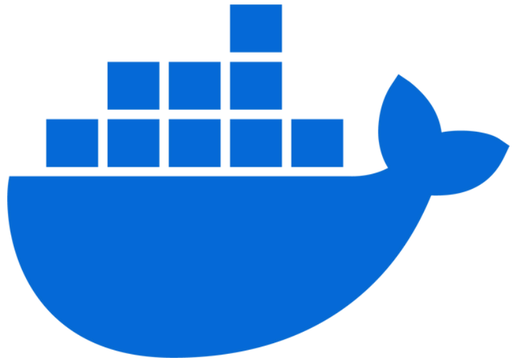
- **Imagens são separadas em camadas. Cada comando adiciona uma nova camada. Elas são *read-only* e são reutilizáveis.**



Dockerfile

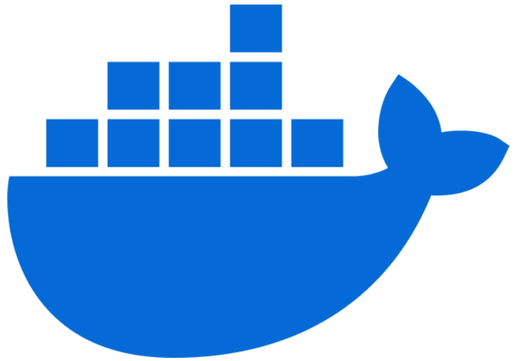
- **É um arquivo de texto usado para criar imagens. Ele contém instruções para construir a imagem, como:**
  - **Qual base usar (e.g., FROM ubuntu:20.04)**
  - **Dependências a instalar (RUN apt-get install)**
  - **Qual comando será executado (CMD ou ENTRYPOINT)**


Dockerfile
Dockerfile > CMD
1
2 FROM node:alpine
3
4 WORKDIR /usr/app
5
6 COPY package*.json ./
7
8 RUN npm install
9
10 COPY . .
11
12 EXPOSE 3000
13
14 CMD npm start



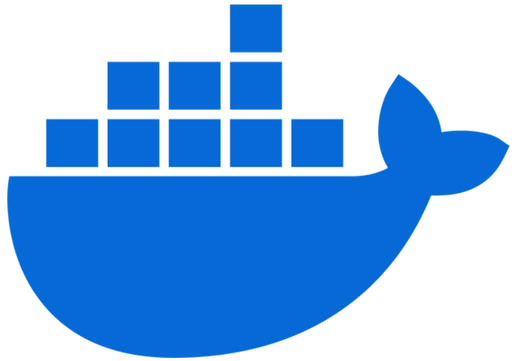
Contêineres

- **É uma instância em execução de uma imagem. Ele é isolado, mas compartilha o kernel do sistema operacional host.**
- **Leve: Contêineres utilizam poucos recursos.**
- **Efêmero: Por padrão, eles não armazenam estado; se o contêiner parar ou for removido, os dados são perdidos (salvo em volumes).**



Volumes

- Volumes são usados para persistir dados que precisam sobreviver ao ciclo de vida de um contêiner.
- Exemplo: Bancos de dados precisam armazenar informações que permanecem mesmo que o contêiner seja reiniciado.
- Tipos de volume:
  - Volumes gerenciados pelo Docker.
  - Bind mounts: Diretórios do sistema host são vinculados diretamente ao contêiner.



Redes

- Redes Docker permitem que os contêineres se comuniquem entre si e com o mundo externo.

- Tipos de redes:
  - Bridge: Rede padrão, usada para contêineres na mesma máquina se comunicarem.
  - Host: Usa diretamente a interface de rede do host, sem isolamento.
  - Overlay: Permite comunicação entre contêineres em diferentes hosts (geralmente em clusters com Docker Swarm ou Kubernetes).

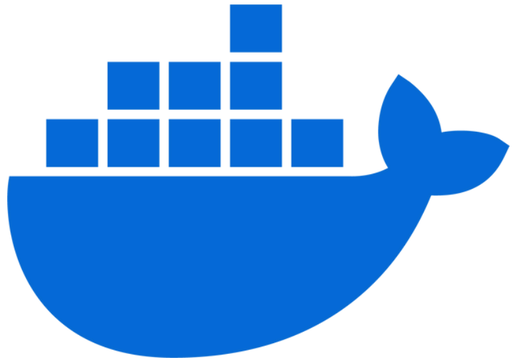
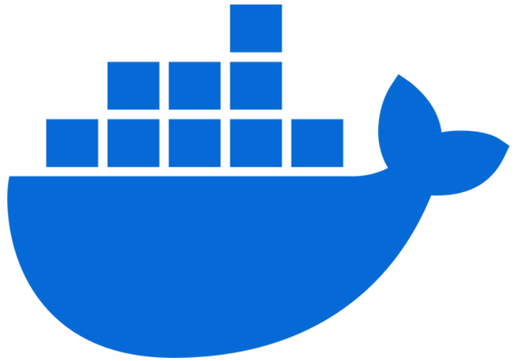


Image Registry (Registro de Imagens)

- **Docker Hub: repositório padrão para armazenar e compartilhar imagens.**

- **Você pode:**
  - **Fazer o pull de imagens públicas (e.g., nginx, postgres).**
  - **Fazer o push de suas próprias imagens para compartilhar com outros.**

- **Registros privados:**
  - **AWS ECR, Azure Container Registry, GitHub Container Registry.**

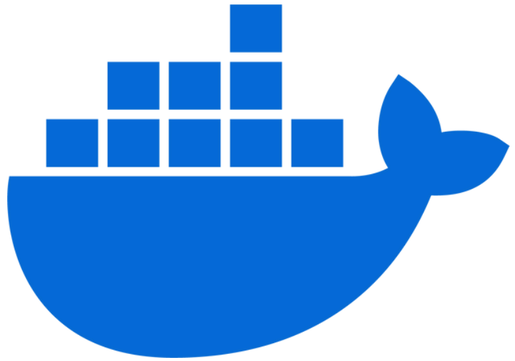


Principais Comandos

- docker pull: Baixa uma imagem do Docker Hub.
- docker build: Cria uma imagem a partir de um Dockerfile.
- docker ps: Lista contêineres em execução.
- docker logs: Mostra os logs de um contêiner.
- docker exec: Executa um comando dentro de um contêiner em execução.
- docker stop, docker start, docker rm: Gerenciamento de ciclo de vida do contêiner.

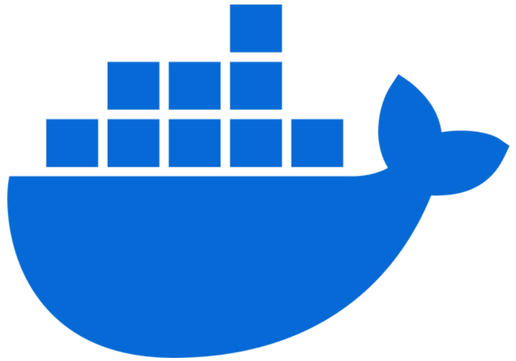
del.tech

03. Docker Compose



Docker Compsoe

- É uma ferramenta que permite definir e gerenciar vários serviços Docker de forma simplificada.
- Ao invés de vários comandos, basta único arquivo YAML (docker-compose.yml).
- Com ele, você pode:
  - Configurar múltiplos contêineres, redes e volumes.
  - Especificar dependências entre serviços.
  - Executar todos os contêineres com um único comando (docker-compose up).



Principais Comandos

- `docker-compose up --build`
- `docker-compose down`
- `docker-compose ps`
- `docker-compose logs -f`
- `docker-compose up <service-name>`

del.tech

Obrigado.

Obrigado.

Obrigado.

wesley.santana@delbank.com