

# Sistemas Distribuídos

Docker

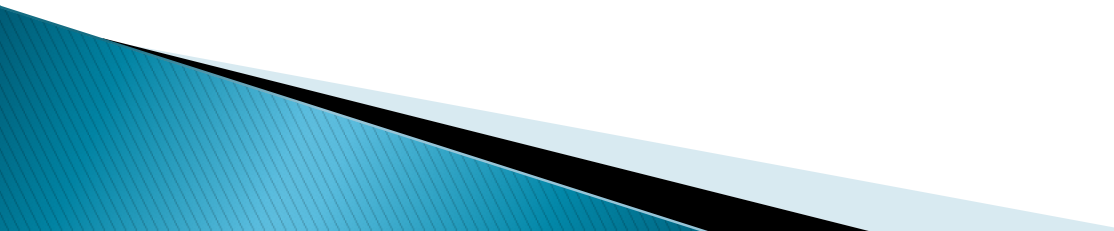
Prof. Tales Viegas

<https://fb.com/ProfessorTalesViegas>


# Docker – O que é

- ▶ Plataforma open-source com a finalidade de criar ambientes isolados para aplicações e serviços. Com este isolamento, garante que cada container tenha tudo o que um serviço precise para ser executado

# Docker – Vantagens

- ▶ Facilidade de configuração do ambiente
  - ▶ Poder trabalhar em outros projetos sem comprometer os recursos da máquina e sem a necessidade de rodar uma VM inteira para isto
  - ▶ Versionamento de código
- 

# Imagens

- ▶ Para que se possa utilizar o Docker, é necessário utilizar uma imagem
  - ▶ Pode ser utilizada uma imagem disponível no Dockerhub (<https://hub.docker.com>) ou em outro registro de imagens
  - ▶ Imagens podem ser criadas a partir de outras imagens
  - ▶ Para baixar: `docker pull <nome da imagem>`
- 

# Containers

- ▶ Um container simplesmente é a execução real de uma imagem do Docker
- ▶ Para inicializar o container, basta executar o comando:
  - `docker run <nome da imagem>`

# Pre-Requisitos

- ▶ Baixar a versão do Docker-CE compatível com o seu sistema operacional
- ▶ <https://www.docker.com/community-edition#/download>
- ▶ Testar se está funcionando através do comando:
  - `docker run hello-world`

# Criando uma imagem

- ▶ Para criar uma imagem, utilizamos um arquivo Dockerfile, contendo tudo o que é necessário para a execução da imagem
- ▶ Após, deve ser criada a imagem com o comando `docker build`
  - `docker build -t <tag>`
  - Tag: tag da imagem, por exemplo: `selatotal/flightdelay`
- ▶ Para ver as imagens disponíveis, usar o comando `docker images`

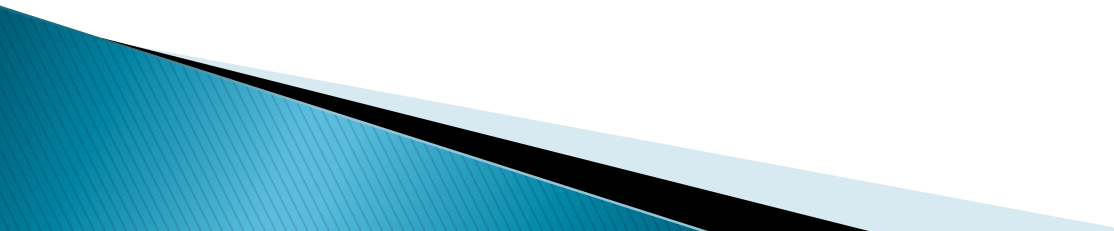
# Exemplos de Comandos Dockerfile

- ▶ FROM <image>
  - Define a imagem base do container
- ▶ ENV <variavel> <valor>
  - Define uma variável de ambiente
- ▶ COPY <origemLocal> <destinoContainer>
  - Copia um arquivo da máquina local para o container
- ▶ WORKDIR <path>
  - Altera a pasta do container para o path informado
- ▶ RUN <comando>
  - Executa um comando no container



# Exemplo Dockerfile Java

```
FROM openjdk:8-alpine
COPY . /usr/src/myapp
WORKDIR /usr/src/myapp
RUN mvn build
WORKDIR /usr/src/myapp/target/classes
CMD ["java", "br.edu.ulbra.flightdelay.Main"]
```



# Comandos úteis

- ▶ `docker ps`
  - Lista os containers que estão rodando
- ▶ `docker exec <container> comando`
  - Executa um comando dentro de um container que está executando
- ▶ `docker stop <container>`
  - Pára a execução de um container
- ▶ `docker rm <container>`
  - Remove um container
- ▶ `docker rmi <imagem>`
  - Remove uma imagem