

Sistemas Operacionais e Arquiteturas Cloud Native

Requisitos para o PI

SlowFu

Aline Germano Costa

Bruna Rodrigues da Silva Pavechi

Gabriela Ventura Oliveira

Maria Fernanda Lima dos Santos

Pedro Rodante Vicente

São Paulo
2023

O que é o Docker?

Docker é uma plataforma de código aberto que permite empacotar e executar aplicativos em contêineres isolados. Os contêineres fornecem um ambiente consistente e portátil, permitindo que os aplicativos sejam executados de maneira eficiente e confiável, independentemente do sistema operacional. O Docker simplifica o processo de desenvolvimento, implantação e orquestração de aplicativos, tornando-o uma ferramenta popular na área de tecnologia.

Por que iremos usar no nosso projeto?

Usá-lo pode oferecer diversas vantagens e benefícios, algumas das vantagens são essas:

Portabilidade: Ele permite empacotar o site e todas as suas dependências, fazendo com que o site possa ser implantado em qualquer ambiente de maneira consistente, independente do sistema operacional, provedor em nuvem ou configuração do servidor.

Consistência: Quando utilizamos o Docker, conseguimos garantir que ele seja sempre consistente em todos os estágios do desenvolvimento, teste e produção. Podemos definir, por exemplo, as dependências do nosso site em um arquivo Dockerfile, fazendo com que a configuração e replicação do ambiente sejam extremamente fáceis.

Escalabilidade: O Docker permite dimensionar facilmente o site para atender às demandas de tráfego crescente. Com a ajuda de ferramentas de orquestração, como o Docker Swarm ou o Kubernetes, você pode criar clusters de contêineres e distribuir a carga de trabalho de maneira eficiente. Isso possibilita que o site seja dimensionado horizontalmente, adicionando ou removendo contêineres conforme necessário, sem interrupções ou impacto na disponibilidade.

Isolamento: Os contêineres do Docker fornecem um ambiente isolado para o site. Isso significa que o site é executado em seu próprio ambiente, com suas próprias bibliotecas, dependências e configurações. Isso oferece maior segurança e evita que um site afete negativamente outros sites ou aplicativos em execução no mesmo servidor.


Colaboração e padronização: O Docker facilita a colaboração entre equipes de desenvolvimento, pois todos os membros podem usar o mesmo ambiente de desenvolvimento, independentemente do sistema operacional que estão utilizando. Além disso, o Docker promove a padronização dos processos de implantação e configuração, tornando mais fácil para as equipes trabalharem juntas e manterem o site de forma consistente.

Em resumo, usar o Docker em um site oferece portabilidade, consistência, escalabilidade, isolamento e facilita a colaboração entre equipes. Essas vantagens tornam o Docker uma escolha sólida para o desenvolvimento, implantação e gerenciamento de sites, permitindo uma experiência mais eficiente e confiável.

Por isso, **utilizamos da seguinte maneira:**

Conteúdo dentro da imagem: Após a construção da imagem Docker com base no Dockerfile, a imagem resultante contém todos os arquivos e diretórios necessários para executar o aplicativo de forma isolada. Durante o processo de criação da imagem, os arquivos do projeto são copiados para dentro da imagem, seguindo as instruções definidas no Dockerfile.

Segue a imagem:



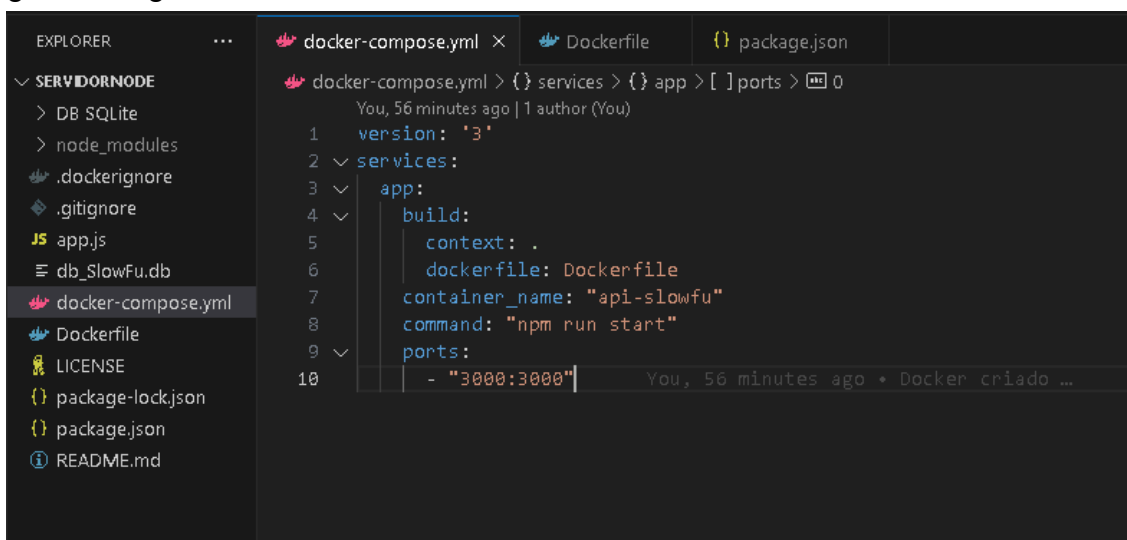
The screenshot shows the VS Code interface with the Explorer on the left and the Dockerfile editor in the center. The Explorer shows a project structure for 'SERVIDORNODE' with files like DB SQLite, node_modules, .dockerignore, .gitignore, app.js, db_SlowFu.db, docker-compose.yml, Dockerfile, LICENSE, package-lock.json, package.json, and README.md. The Dockerfile editor shows the following content:

```
Dockerfile > ...
You, 54 minutes ago | 1 author (You)
1 FROM node:20-alpine
2
3 LABEL version="1.0"
4
5 WORKDIR /app
6
7 COPY package*.json ./
8
9 RUN npm install
10
11 COPY . .
12
13 EXPOSE 3000
14
15 CMD ["npm", "start"]
```

The status bar at the bottom right indicates 'You, 55 minutes ago • Docker criado'.

Com o **Compose**, você pode definir facilmente a estrutura e a configuração do seu aplicativo usando uma sintaxe simples e declarativa. Isso elimina a necessidade de executar comandos separados para criar e iniciar cada serviço individualmente.

Segue a imagem:



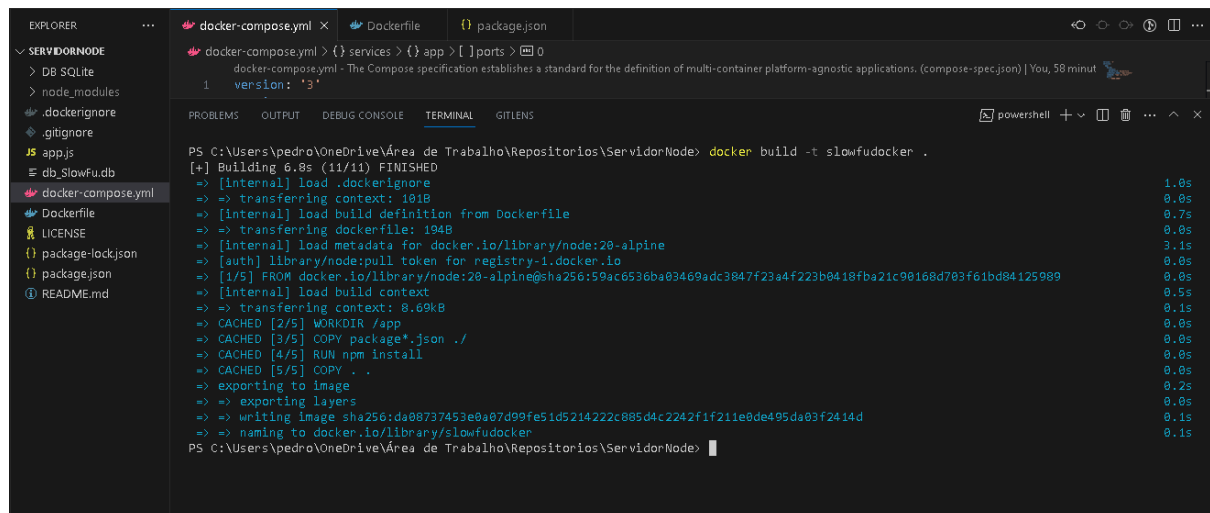
The screenshot shows the VS Code interface with the Explorer on the left and the docker-compose.yml editor in the center. The Explorer shows the same project structure as the previous image. The docker-compose.yml editor shows the following content:

```
docker-compose.yml > {} services > {} app > [ ] ports > [ ] 0
You, 56 minutes ago | 1 author (You)
1 version: '3'
2 services:
3   app:
4     build:
5       context: .
6       dockerfile: Dockerfile
7     container_name: "api-slowfu"
8     command: "npm run start"
9     ports:
10      - "3000:3000"
```

The status bar at the bottom right indicates 'You, 56 minutes ago • Docker criado ...'.

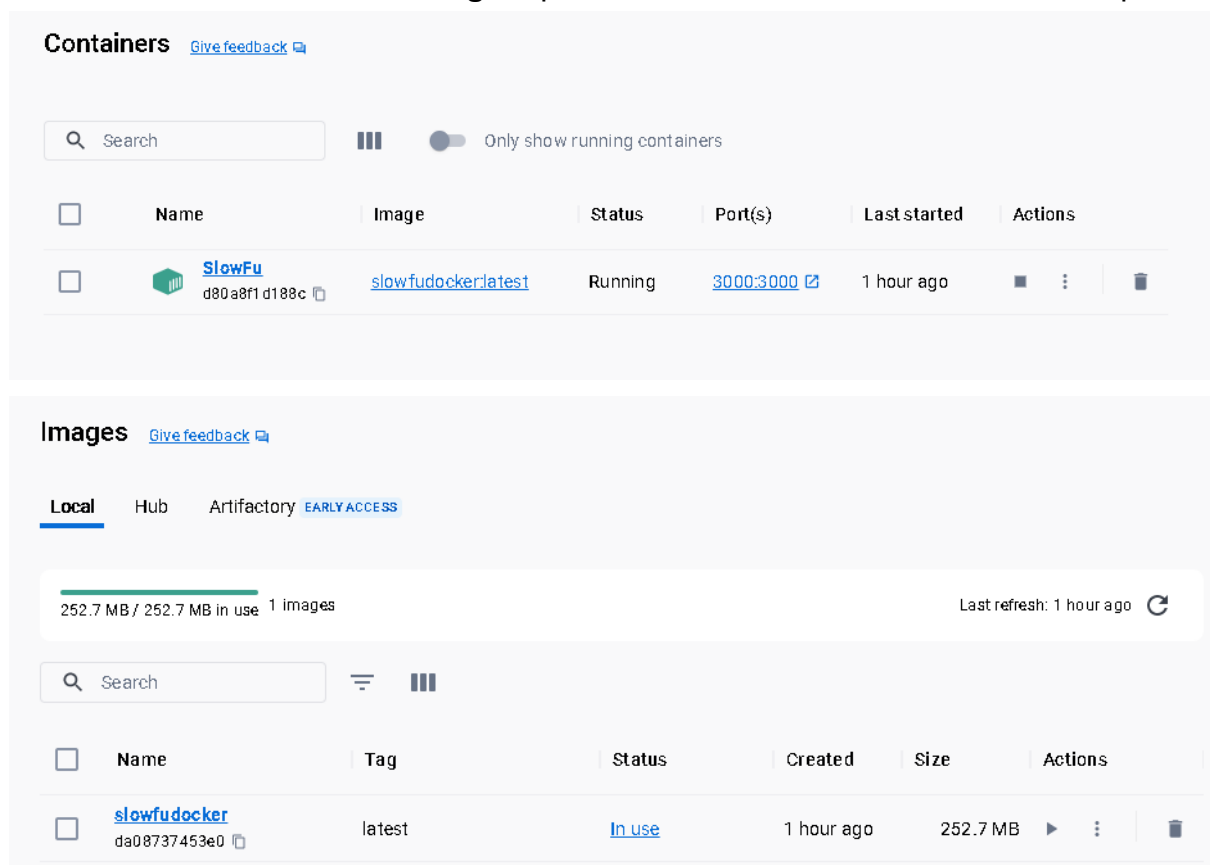
O comando **docker build -t slowfudocker .** combina a definição da tag com a especificação do diretório atual contendo o Dockerfile. Isso instruiu o Docker a construir a imagem usando as instruções do Dockerfile presente no diretório atual e atribuir a tag "slowfudocker" a essa imagem resultante.

Segue a imagem:




```
PS C:\Users\pedro\OneDrive\Área de Trabalho\Repositorios\ServidorNode> docker build -t slowfudocker .
[+] Building 6.8s (11/11) FINISHED
=> [internal] load .dockerignore
=> => transferring context: 101B
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 194B
=> [internal] load metadata for docker.io/library/node:20-alpine
=> [auth] library/node:pull token for registry-1.docker.io
=> [1/5] FROM docker.io/library/node:20-alpine@sha256:59ac6536ba03469adc3847f23a4f223b0418fba21c90168d703f61bd84125989
=> [internal] load build context
=> => transferring context: 8.69kB
=> CACHED [2/5] WORKDIR /app
=> CACHED [3/5] COPY package*.json ./
=> CACHED [4/5] RUN npm install
=> CACHED [5/5] COPY . .
=> exporting to image
=> => exporting layers
=> => writing image sha256:da08737453e0a07d99fe51d5214222c885d4c2242f1f211e0de495da03f2414d
=> => naming to docker.io/library/slowfudocker
PS C:\Users\pedro\OneDrive\Área de Trabalho\Repositorios\ServidorNode>
```

Além disso, o **contêiner** e a **imagem** prontos e visualizados no Docker Desktop:




SlowFu

<




[slowfudocker:latest](#)

d80a8f1d188c





[3000:3000](#)





STATUS

Running (1 hour ago)









Logs

Inspect

Terminal

Files

Stats

2023-06-08 05:03:41 Atualizei os Posts

2023-06-08 05:03:41 Acesse-os em: /todos_posts

2023-06-08 05:03:48 Fui acionado

2023-06-08 05:03:48 Saindo!

2023-06-08 05:03:48 Deu tudo certo!

2023-06-08 05:05:05 {}

2023-06-08 05:05:05 Realizando login

2023-06-08 05:05:05 Usuário encontrado.

2023-06-08 05:05:05 Senha correta.

2023-06-08 05:05:05 Atualizei os Usuários

2023-06-08 05:05:05 { email: 'pedrorodantevicente@gmail.com' }

2023-06-08 05:05:05 pedrorodantevicente@gmail.com

2023-06-08 05:05:05 Acesse-os em: /dados_usuario

2023-06-08 05:05:05 [

2023-06-08 05:05:05 {

2023-06-08 05:05:05 ID: 31,

2023-06-08 05:05:05 nome: 'Pedro Rodante',

2023-06-08 05:05:05 senha: '123123',

2023-06-08 05:05:05 email: 'pedrorodantevicente@gmail.com',


2023-06-08 05:05:05 telefone: 11975744022


2023-06-08 05:05:05 }


2023-06-08 05:05:05]


2023-06-08 05:05:06 Atualizei os Posts


2023-06-08 05:05:06 Acesse-os em: /todos_posts











Conclusão

Em resumo, o Docker revoluciona a forma como desenvolvemos, implantamos e gerenciamos aplicativos. Sua abordagem baseada em contêineres oferece benefícios significativos, como portabilidade, consistência, escalabilidade e isolamento. Ao adotar o Docker, as empresas podem acelerar o ciclo de desenvolvimento, melhorar a eficiência operacional e fornecer experiências de usuário mais confiáveis. Com um futuro promissor, o Docker continua a desempenhar um papel vital na evolução da tecnologia moderna.