

## Formas de uso de Containers (Docker) no serviço Sites@UFSC

17/05/2024 09:06:55

[Imprimir artigo da FAQ](#)

<b>Categoria:</b>	Hospedagem no serviço Sites@UFSC::Informações adicionais	<b>Votos:</b>	0
<b>Estado:</b>	público (todos)	<b>Resultado:</b>	0.00 %
		<b>Última atualização:</b>	Seg 23 Out 15:49:34 2023

### Problema (público)

Quais são as formas de uso, limitações e restrições da hospedagem de Containers (Dockers) no serviço Sites@UFSC?

### Solução (público)

O plataforma de containers dentro do serviço Sites@UFSC tem como objetivo atender as demandas de hospedagem que não se encaixam no cluster padrão (Apache+PHP) do serviço, assim como não se encaixam nas demandas de Processamento de Dados de Pesquisa (abrir chamado para melhor análise.) Esta alternativa exige que o usuário-técnico tenha conhecimentos de containers (Docker+Compose) e GIT para efetuar a instalação de sua aplicação. Denominamos aqui:

- aplicação: qualquer software, site, sistema que necessite ser hospedado.
- service/workload: uma ou mais instancias de um mesmo container
- stack/namespace: conjunto de services que constituem uma aplicação (Ex. Web+DB+Cache)
- project/team: grupo de pessoas que poderão ter acesso ao stack/namespace.

1 - O usuário deve prover (desenvolver ou buscar na internet) um docker-compose.yml que execute o seu ambiente de containers. Ao executar o comando `docker-compose up`, na base do projeto (pasta principal), a aplicação deve ser executada por completo.

1.1 - Se este comando não funcionar no seu computador, não adianta passar pro próximo passo.

1.2 - Se você é o desenvolvedor da aplicação, certifique-se de que as senhas seja externalizadas, ou seja, seja configuradas como variáveis de ambiente no docker-compose.

1.3 - As bases de dados dentro do docker-compose serão consideradas apenas para teste, a SeTIC proverá novas credenciais durante a configuração dos serviços (stack).

2 - Os códigos (aplicação, docker-compose e Dockerfiles) devem ser armazenados no serviço codigos.ufsc.br, para que o processo de implantação em produção seja automatizado (CI/CD) pela SeTIC.

2.1 - Convenção: O arquivo Dockerfile deve estar na base do projeto, pois será executado o "docker build" desta pasta.

3 - Sobre as responsabilidades:

- 3.1 - Caso o requisitante não seja servidor da UFSC, deve-se ter um responsável que responda pelo sistema. Coloque-o em cópia dos e-mails/chamados.
- 3.2 - Caso o requisitante seja da UFSC, deve ser especificado o vínculo da necessidade (Projeto, DoD, etc).

Passos a serem seguidos:

- 1 - Criar o seu projeto localmente (caso esteja desenvolvendo).
- 2 - Criar e testar o Dockerfile (ou copia-lo da internet)
- 3 - Criar e testar o docker-compose.yml, lembre-se de discriminar as variáveis de ambiente neste arquivo.
- 4 - Subir o projeto no codigos.ufsc.br, nos branches master ou main.
- 5 - Criar o arquivos .gitlab-ci.yml com base no template do codigos.ufsc.br (Ver video tutorial)
  - 5.1 - Caso a aplicação necessite de variáveis de ambiente durante o build (build-args), as variáveis devem ser registradas nas configurações do CI/CD e o "docker build" deve ser personalizado no .gitlab-ci.yml, como explicado no video.
- 6 - Informar a SeTIC (que habilitará a integração (CI/CD)) informando a URL do Repositorio e as devidas variáveis de ambiente de produção (caso necessário).

Obs: Após criada a hospedagem (passo 5) toda alteração de código no codigos.ufsc.br (commit) será "automaticamente" colocada em produção.

Videos Tutoriais:

\* [1]Configurar o CI/CD no codigos.ufsc.br

[1] <https://youtu.be/MyQr0HBIVpg>