

ESTUDO DE VIABILIDADE DE IMPLANTAÇÃO DE UM AMBIENTE WEB EM CONTAINERS UTILIZANDO DOCKER

FEASIBILITY STUDY OF IMPLEMENTING A WEB ENVIRONMENT IN CONTAINERS USING DOCKER

Emerson Bervian Klunk¹, Aléssio Inácio Cagliari²

RESUMO

Este artigo visa apresentar uma pesquisa de viabilidade de implantação de um ambiente de *containers* utilizando o *Docker* como ferramenta de gerenciamento. Para tanto, busca-se realizar um estudo sobre vantagens da utilização da ferramenta, conhecer experiências profissionais da utilização do *Docker*, aprofundar conhecimentos sobre funcionalidades dos *containers* e analisar o custo benefício de implantação do ambiente. A pesquisa se dá de forma aplicada, classifica-se como exploratória, outrossim, sua abordagem é qualitativa e seu procedimento classifica-se como sendo estudo de caso. Por fim, notou-se a viabilidade da implantação do ambiente de *containers* pelo fato do seu baixo custo uma vez que os *softwares* são gratuitos e também pelos inúmeros benefícios que o acompanham.

Palavras-chave: *Container; Docker; Virtualização; Computação em Nuvem; Servidor.*

ABSTRACT

This paper aims to present a feasibility research for the implementation of a container environment using Docker as a management tool. To this purpose, we seek to conduct a study about the advantages of using the tool, learn about professional experiences of using Docker, a deeper knowledge about container functionality and analyze the cost-benefit of deploying this environment. The research is applied, is classified as exploratory, its approach is qualitative and its procedure is classified as a case study. Finally, it was noted the feasibility of deploying the container environment due to its low cost since the software is free and also the numerous benefits that accompany it.

Keywords: *Container; Docker; Virtualization; Cloud Computing; Server; Linux*

1 INTRODUÇÃO

Nos dias atuais, o uso da tecnologia para os mais diversos fins, desde comunicação até o armazenamento de dados empresariais, tem se tornado evidente. Estas e outras tecnologias possuem algo em comum: a necessidade de uma máquina (seja ela física ou virtual), onde os

¹ Acadêmico de Gestão da Tecnologia da Informação, Suporte Técnico em TI, UCEFF Itapiranga, emersonklunk2014@gmail.com.

² Mestre em Ensino Científico e Tecnológico pela Universidade Regional Integrada de Santo Ângelo – RS, URI. Especialista pela Escola Superior Aberta do Brasil - ESAB. Professor do Curso de Gestão da Tecnologia da Informação. Sócio proprietário, APC Inova Tecnologia, alessio@uceff.edu.br.

Centro Universitário FAI • |49| 3678.8700
Rua Carlos Kummer, 100
Bairro Universitário
Itapiranga - SC • 89896-000

Centro Politécnico • |49| 3319.3800
Av. Irineu Bornhausen, 2045 E
Bairro Quedas do Palmital
Chapecó - SC • 89814-650

Unidade Central • |49| 3319.3838
Rua Lauro Müller - 767 E
Bairro Santa Maria
Chapecó - SC • 89812-214

dados, configurações e bibliotecas necessárias para o correto funcionamento do serviço fiquem armazenados. Esta máquina é conhecida como sendo o servidor da ferramenta e, para ela ser acessível por todos os usuários que desejarem usufruir do recursos, deve estar diretamente conectada à *internet*. Quando um servidor está conectado à *internet*, ou seja, para usufruir de suas funcionalidades, basta ter acesso físico a qualquer computador com acesso à grande rede. Este servidor passa a atender os conceitos de computação em nuvem.

Neste contexto, o tema de estudo é a viabilidade de implantação de um ambiente *Web* com *containers*, utilizando *Docker*. O problema que se busca responder é: É viável instalar um sistema de *containers* para executar *softwares Web* em uma empresa do setor de educação? Assim, o presente artigo visa apresentar a estruturação de um servidor *Web* em *container*, utilizando *Docker*. Silva (2016), explica o *Docker* como sendo uma forma de construir aplicações independente da plataforma (como por exemplo a *Web*) e linguagem (como por exemplo PHP³) sob *containers* as quais podem ser executadas em qualquer ambiente.

A ideia de utilizar esta tecnologia serve principalmente para quebrar aquele cenário típico de um Centro de Processamento de Dados (CPD) que, segundo Silva (2016), é compreendido pelo ato de comprar ou alugar máquinas *desktop* específicas para serem utilizadas como servidores para hospedar as aplicações, uma vez que os *containers* podem armazenar e gerenciar diversos serviços em uma mesma máquina, tirando a obrigatoriedade de existir um computador para cada serviço.

Para tanto, o objetivo geral deste artigo é desenvolver um estudo de viabilidade de implantação de um sistema de gestão de *containers* em uma empresa da área da educação. Neste sentido, tem-se como objetivos específicos: realizar um estudo teórico sobre os motivos de utilizar *containers*; conhecer as vantagens em utilizar *containers* no lugar de máquinas virtuais; aprofundar conhecimentos sobre configurações e funcionalidades dos *containers*; analisar o custo-benefício de utilizar *containers* em uma organização.

O presente artigo justifica-se pela ideia de trazer conhecimentos sobre uma ferramenta relativamente nova que é o *Docker* que, segundo Silva (2016), foi anunciado publicamente pela primeira vez por Solomon Hykes, *CEO* da *dotCloud*, em 15 de março de 2013, durante uma conferência de *Python* na Califórnia.

³ De acordo com o site oficial (2019), o PHP pode ser compreendido como sendo uma linguagem *script* de código aberto, de uso geral e que atua no lado do servidor, é muito utilizada no desenvolvimento para *Web* e pode ser embutida no *HTML* (Linguagem utilizada para estruturação de páginas *Web*).

Além disso, justifica-se pelos benefícios que essa tecnologia traz, entre eles, a facilidade no momento de restaurar os serviços que estejam rodando em *containers*, os quais tenham sido interrompidos por conta de alguma anomalia, podendo ser relacionada a rede elétrica ou a própria máquina *host* do serviço, entre outros

Vale ressaltar que este é serviço que permite a utilização customizada dos recursos e bibliotecas do sistema operacional, tornando assim os *containers* muito mais leves do que as conhecidas máquinas virtuais.

2 FUNDAMENTAÇÃO TEÓRICA

Atualmente, a tecnologia está inserida em grande parte das atividades que realizamos, e como exemplo pode-se citar os diversos meios de comunicação, entre eles, o *WhatsApp* e o correio eletrônico. Para que estes e outros serviços estejam disponíveis na *Internet* e os usuários tenham acesso, segundo Kurose e Ross (2003), é necessário toda uma infraestrutura de rede que servirá de provedor para a ferramenta. Um computador específico incluso nesta infraestrutura, é responsável por concentrar todas as codificações necessárias para o correto funcionamento da ferramenta.

De acordo com Veras (2016), Este computador é conhecido como sendo o *host* ou o servidor que, por sua vez, pode ser físico (rodando diretamente no *hardware* do computador) ou virtual (rodando em uma máquina virtual, do inglês *Virtual Machine* (VM)).

O ato de hospedar um serviço específico em uma máquina física totalmente dedicada à execução deste, segundo Silva (2016), faz parte de um paradigma que se estendeu por anos.

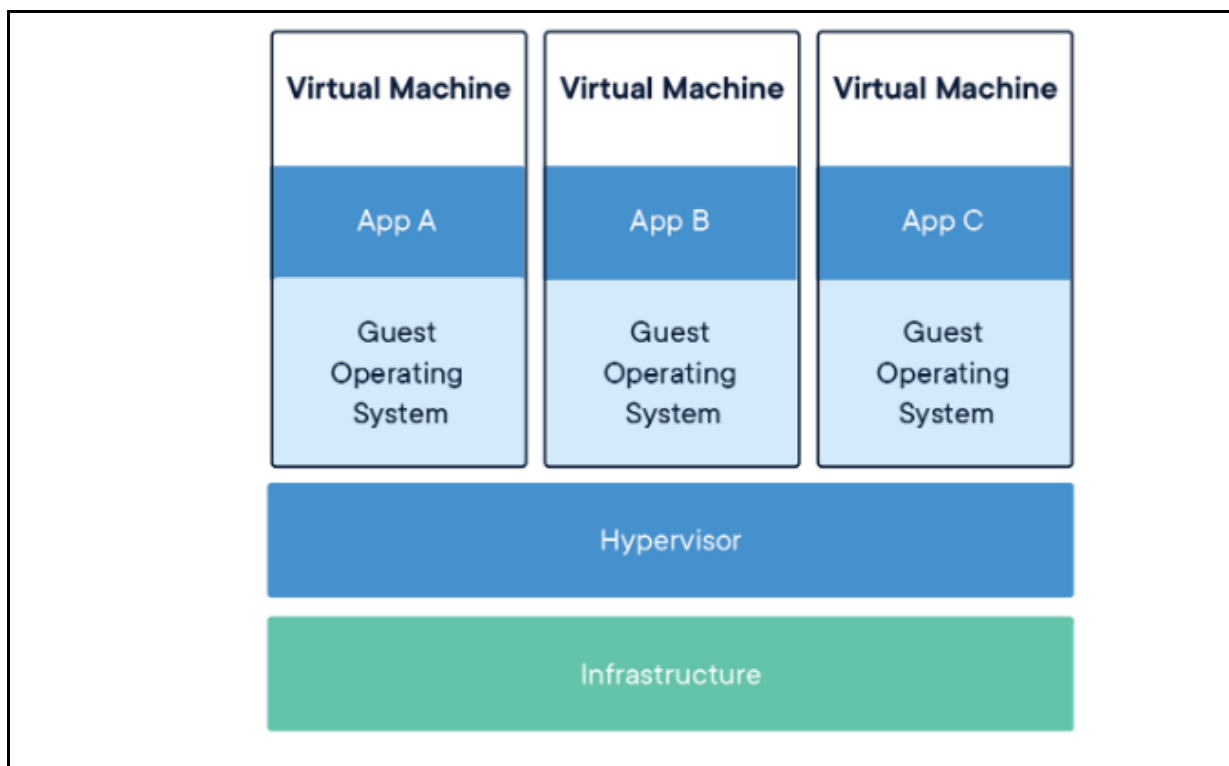
O paradigma mencionado se quebrou com a chegada da tecnologia, a qual possibilitou virtualizar um servidor. O mesmo autor ainda complementa que esta é a forma mais utilizada no momento de colocar um serviço no ar, a qual vem crescendo cada vez mais com a popularização da Computação em Nuvem (SILVA, 2016), que segundo Witting e Witting (2016), é compreendido como forma de expressão que remete ao consumo de serviços de Tecnologia da Informação (TI). Estes não podem ser vistos ou tocados diretamente pelo usuário, ou seja, existem diversas camadas interligadas separando o usuário e o *host* do serviço acessado.

As ferramentas disponibilizadas em Nuvem, segundo Silva (2016), não se encontram mais em computadores físicos, mas sim encapsuladas em blocos de processamento e memória. Na técnica da virtualização há um sistema operacional inteiro conhecido por *guest* com todas

as bibliotecas, binários e seu *kernel*⁴ rodando em um segundo sistema operacional, conhecido por *host*, que por sua vez, utiliza um *hypervisor*, como por exemplo o *VMware* ou o *Xen*, para gerenciar a máquina virtual e mapear seu sistema operacional *guest*, conforme apresentado na figura 1.

Em outras palavras, conforme afirma Veras (2012), a virtualização vem para simplificar o gerenciamento, de forma a permitir a flexibilização e ampliação da capacidade de processamento da máquina *host*. As funcionalidades disponibilizadas pelos *softwares* que gerenciam máquinas virtuais permitem melhorar a disponibilidade dos serviços e a recuperação de desastres de uma forma mais agilizada e mais barata.

Figura 1 - Organização lógica das máquinas virtuais



Fonte: Docker, 2019

Hypervisor, de acordo com afirmações de Chee e Franklin Jr. (2013), nada mais é do que um *software* que possui como finalidade virtualizar o *hardware* e o *software*, permitindo assim, a execução de múltiplos sistemas operacionais em um mesmo computador de forma simultânea.

⁴ Segundo Deitel, Deitel e Choffnes (2005), *kernel* é o núcleo do sistema operacional, onde se encontram os componentes centrais do sistema.

Silva (2016) contribui dizendo que é o *hypervisor* que utiliza os recursos de hardware do *host* e simula para as VMs.

2.1 DIFERENÇAS ENTRE VIRTUALIZAÇÃO E COMPUTAÇÃO EM NUVEM

Existem diferenças entre a virtualização e a computação em nuvem. Conforme já ressaltado anteriormente, a computação em nuvem se resume na utilização de recursos de TI por parte dos usuários. Segundo Chee e Franklin Jr. (2013), indiferente se a ideia é criar uma nuvem de menor escala em um ambiente de transição de dados já existente ou colocar diversos serviços em uma nuvem completa, nos dois casos o ambiente em que estarão rodando será virtual.

O mesmo autor ressalta também que uma coisa é criar ambientes virtuais em um único *hardware* com o objetivo de maximizar o uso da máquina, outra coisa é utilizar uma imagem virtual única por um determinado número de recursos de *hardware* para otimizar o desempenho da máquina e a disponibilidade do serviço disponibilizado. Isso reflete a escolha do que virtualizar, e esta deve partir de uma análise de quais os recursos já estão sendo utilizados em vários servidores para então centralizá-los com o intuito de otimizar o desempenho, diminuir o tempo de ociosidade, diminuir gastos com energia e manutenção de *hardware* e *software*. (CHEE; FRANKLIN JR., 2013).

A maior vantagem de se realizar a unificação destes serviços em uma máquina é a possibilidade de gerar uma imagem das máquinas virtuais em qualquer que seja seu estado e armazenar em *backup* externo, o que virá a eliminar, em caso de desastre, o retrabalho de configurar do zero todas as VMs, uma vez havendo a possibilidade de restauração das imagens salvas. (CHEE; FRANKLIN JR., 2013).

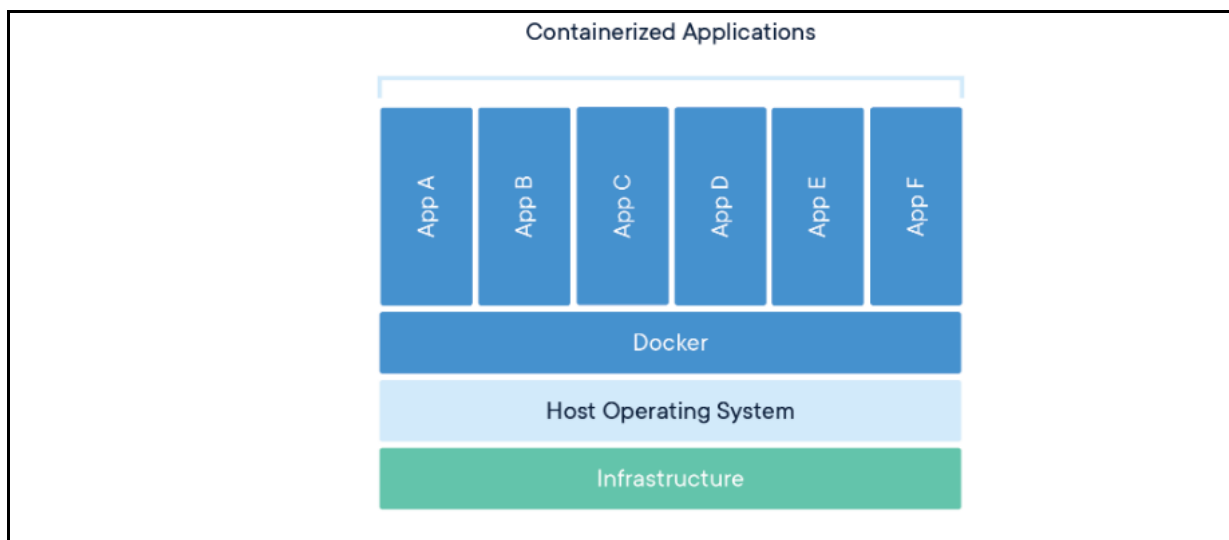
Resumindo, todos os serviços acessíveis através da *Internet* ou não, desde que estejam disponíveis de forma virtual de modo a desvincular os usuários da necessidade de gerenciamento de armazenamento e demais recursos, podem ser considerados como estando armazenados em nuvem. A computação em nuvem é o resultado de um ambiente totalmente virtualizado.

2.2 CONTAINERS E O DOCKER

O *Docker* é uma ferramenta para gerenciamento de *containers*, os quais, segundo Castro e Vitalino (2018), se compreendem pelo ato de agrupar aplicações juntamente com todas as dependências necessárias para seu correto funcionamento em uma espécie de bloco virtual,

compartilhando assim o mesmo *kernel* do sistema operacional, o qual atua como *host* do *container*, e que pode ser virtual ou físico. É possível observar na figura 2 a organização lógica de um ambiente de *containers*.

Figura 2: Organização lógica de aplicações em *containers*



Fonte: *Docker*, 2019

Segundo o *site* oficial do *Docker* (2019), as imagens dos *containers* se tornam *containers* em tempo de execução e, no caso do *Docker*, quando são executadas pelo *Docker Engine*⁵. Os *containers* isolam os *softwares* de seu ambiente de execução, garantindo assim um funcionamento uniforme.

A ferramenta de gestão de *containers* está disponível para *Windows* e *Linux* e é sempre executada da mesma forma, independente da infraestrutura em que está inserida. (DOCKER, 2019).

Os *containers* que são gerenciados pelo *Docker*, conforme seu próprio *site* oficial (2019), possuem três principais características:

- Padronização: O *Docker* estabeleceu um padrão de execução de *containers*, o que possibilita a portabilidade para qualquer ambiente.
- Leveza: Como os *containers* compartilham do mesmo *kernel* do sistema operacional *host*, não necessitam de todo um sistema operacional próprio para

⁵ Segundo o *site* do *Docker* (2019), *Docker Engine* é o tempo de execução de *container* de fato. ele permite que aplicativos rodem em diversos ambientes, independentes de infraestrutura e sistema operacional, desta forma, dando uma solução bastante eficiente para os problemas relacionados às dependências e bibliotecas no momento de importar alguma aplicação.

sua execução, desta forma, aumentando em sua eficiência em larga escala, além de reduzir custos com licenciamento.

- Segurança: Os aplicativos executados em *containers* são mais seguros, e o *Docker* fornece excelentes recursos de isolamento para os blocos virtuais.

2.3 DIFERENÇAS ENTRE *CONTAINERS* E MÁQUINAS VIRTUAIS

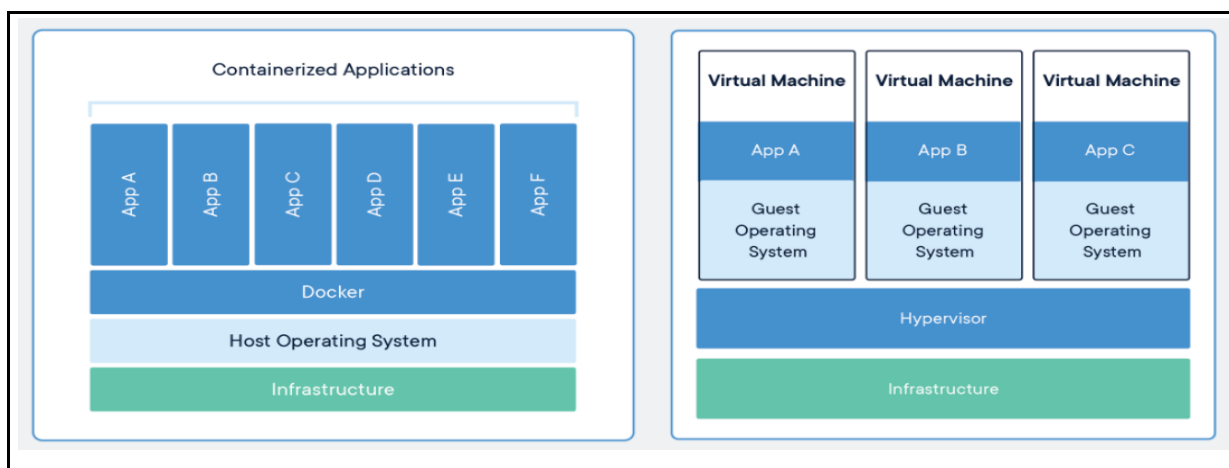
Conforme já colocado, as duas tecnologias compartilham a ideia de aprimorar a utilização dos recursos de *hardware* do *host* em tempo de execução, reduzir o espaço físico utilizados pelos servidores no CPD, além de facilitar a portabilidade das aplicações e possibilitar o isolamento do *software*, esteja ela em uma VM ou em um *container*.

Segundo o *site* oficial do *Docker* (2019), embora existam benefícios semelhantes entre *containers* e VMs, as duas tecnologias funcionam de maneira diferente, uma vez que, ao contrário das máquinas virtuais, que virtualizam o *hardware* do *host*, os *containers* virtualizam o sistema operacional, além de serem mais portáteis e eficientes.

Castro e Vitalino (2018), complementam que esta tecnologia é mais leve e possui maior integração com o sistema operacional, o que traz maior desempenho, uma vez que o gerenciamento de recursos é centralizado.

Os autores ainda afirmam que, ao contrário das VMs, que carregam consigo um sistema operacional inteiro, em sua grande maioria, as imagens de *containers* são bastante enxutas, uma vez compreendido que armazenam somente as bibliotecas necessárias para o funcionamento da aplicação, conforme pode-se observar na figura 3. Isso ocorre porque existe o compartilhamento do *kernel* do sistema operacional *host* com o *container*. (CASTRO; VITALINO. 2018).

Figura 3: Comparação entre um ambiente de *containers* e um ambiente de máquinas virtuais



Fonte: Docker, 2019

Quando executamos uma VM, segundo Castro e Vitalino (2018), é necessária a instalação de um sistema operacional completo e o *hardware* do *host* é totalmente virtualizado, desta forma, utilizando mais recursos para a execução de máquinas virtuais. Os *containers* atuam de uma maneira mais eficiente, porque, conforme já dito, não necessitam de todo um sistema operacional completo e, além de compartilhar os recursos do *host*, possibilitam também a execução de um maior número de *containers* em um mesmo servidor, se comparado a quantidade de VMs que seriam possíveis rodar em um servidor com o mesmo *hardware*.

Os autores completam dizendo que outro ponto interessante da utilização de *containers* é a portabilidade. Desvinculado do ambiente em que ele foi criado, é possível rodar o mesmo bloco virtual em qualquer outro *host*, independente de estar rodando em *Linux*, *MacOS* ou *Windows*. Isso porque todas as dependências necessárias estão contidas no *container*. (CASTRO; VITALINO, 2018).

3 METODOLOGIA

O presente artigo é resultado de uma pesquisa aplicada em sua finalidade, pois, de acordo com Mascarenhas (2012), objetiva estudar um problema no contexto em que está inserido de forma a encontrar a melhor solução ou melhoria. Assim, a pesquisa objetiva melhorar a utilização de um serviço já em funcionamento.

Ainda, classifica-se como sendo exploratória, pois segundo Severino (2007, p. 123), “busca apenas levantar informações sobre um determinado objeto, delimitando assim um campo de trabalho, mapeando as condições de manifestação desse objeto”. Isso posto, nesta

pesquisa busca-se levantar informações acerca dos *containers* e sua utilização na execução de um serviço *Web*.

Em relação à abordagem, a pesquisa é qualitativa, pois conforme Mascarenhas (2012), esta abordagem é utilizada quando a intenção é aprofundar os conhecimentos sobre o assunto principal. Neste sentido, este artigo busca aprofundar os conhecimentos sobre o gerenciamento de *containers* utilizando o *Docker* como ferramenta.

Quanto aos procedimentos, esta pesquisa classifica-se como Estudo de Caso, pois segundo Severino (2007), concentra estudos de um assunto específico que utiliza conhecimentos já existentes. Nesta acepção, a presente pesquisa concentra-se no estudo dos *containers* e seu gerenciamento com base em publicação dos autores citados no decorrer do artigo.

O estudo de viabilidade a ser desenvolvido nesta pesquisa inclui o levantamento sobre custos, tanto de equipamentos, bem como da capacitação aos profissionais para que possam manipular a ferramenta da forma correta, sendo que estes dados serão coletados através de pesquisas com fornecedores e *sites de e-commerce*.

4 APRESENTAÇÃO E DISCUSSÃO DOS RESULTADOS

O estudo aqui apresentado foi realizado em uma instituição de ensino superior privada localizada na cidade de Itapiranga, no Oeste de Santa Catarina. Além da unidade de Itapiranga, existe uma unidade localizada na cidade de Chapecó.

A empresa é compreendida como sendo de grande porte com aproximadamente 180 funcionários, divididos entre colaboradores, professores e reitoria. Em se tratando de pessoas que frequentam a instituição, o número se aproxima de 1700 indivíduos, podendo variar de acordo com o dia.

Além dos serviços de Graduação, a instituição ainda provê cursos de Pós-Graduação e Extensão em variadas áreas e uma grande quantidade de eventos abertos à comunidade acadêmica e externa.

Por conta da proximidade com o Rio Grande do Sul, a instituição conta com um grande número de acadêmicos vindo da região noroeste deste estado, e ainda alguns vindos de regiões mais distantes. Porém, a maioria dos alunos vêm do mesmo estado no qual a empresa se localiza, abrangendo cidades mais próximas como São João do Oeste, Iporã do Oeste e Tunápolis, até cidades mais distantes como Mondaí, São Miguel do Oeste, Riqueza, Palmitos e Santa Helena.

A empresa organiza-se de uma forma na qual o contato entre colaboradores, professores e direção seja o mais próximo possível, em outras palavras, existe grande facilidade na troca de informações. Os colaboradores são os responsáveis por prestar auxílio nas necessidades específicas dos acadêmicos e funcionários. Neste grupo inclui-se o setor de desenvolvimento responsável por, além do desenvolvimento e garantia de funcionamento dos sistemas internos da instituição, prestar suporte aos funcionários e acadêmicos em eventuais dúvidas.

Os sistemas citados são hospedados localmente em sua maioria na unidade de Chapecó, os sistemas hospedados no CPD da unidade de Itapiranga são distribuídos em algumas máquinas por motivos de compatibilidade, uma vez que os diferentes sistemas hospedados utilizam diferentes versões do PHP. Isso faz com que o *hardware* das máquinas utilizadas não sejam aproveitados de forma eficiente.

Para maximizar o aproveitamento dos recursos da máquina, estuda-se a viabilidade de implantação de um sistema de *containers*, que possibilitará a melhor utilização dos recursos de *hardware*, uma vez que esta tecnologia permite a execução de vários *containers* simultaneamente e independentes entre si. Desta forma, possibilita que mais de um sistema, sendo eles compatíveis com diferentes versões do PHP, sejam rodados em uma mesma máquina *host* que irá hospedar os *containers* que, por sua vez, estarão configurados com as respectivas versões do PHP necessárias para rodar aplicação *Web*.

4.1 VIABILIDADE TÉCNICA

Para que a implantação de um sistema de *containers* seja possível, consoante ao *site* do *Docker* (2019), é necessário atender alguns requisitos, dentre eles estão, um computador conectado à Internet com no mínimo 4GB de memória RAM e um disco rígido com no mínimo 3GB disponíveis. Neste dispositivo deverá estar instalado alguma distribuição GNU/Linux cuja versão do *kernel* seja igual ou superior a 3.10. Além disso, é preferível que este computador tenha seu IP⁶ fixado.

Considerando que seja necessário a aquisição de uma máquina que atenda os requisitos acima listados, compreende-se que tais recursos sejam de fácil acesso, uma vez que existe uma grande quantidade de *sites* de *e-commerce* que fazem a comercialização de dispositivos computacionais.

⁶ De acordo com Brito (2013), o endereço IP pode ser definido como sendo um número que corresponde ao endereço virtual de cada máquina em uma rede. É utilizado para a identificação das máquinas de um determinada rede e deve ser exclusivo para cada dispositivo conectado.

Por fim, é necessário que o indivíduo que fará a instalação do *Docker* tenha conhecimentos específicos sobre a tecnologia para que a instalação e configuração seja feita da maneira correta. Para tanto, recomenda-se a realização do curso ‘*Docker* do zero - introdução à administração de *containers*’, disponível no site *Udemy* (2019).

4.1.1 Processo de Instalação

Iniciando-se o processo de instalação, já com o sistema operacional devidamente instalado e configurado com um endereço de IP fixo, conforme requisitos, realiza-se a instalação do *Docker*, conforme descrito a seguir. Como exemplo será utilizado o sistema operacional Ubuntu 16.04.

```
$ sudo apt-get update  
$ sudo apt-get install docker.io
```

Com o gerenciador de *container* instalado, deve-se buscar a imagem de *container* que atenda aos requisitos de sistema da aplicação *Web* que se pretende executar. Para isso, deve-se acessar o site *Docker Hub* que é um repositório em nuvem com uma grande variedade de imagens projetadas para atender os mais variados requisitos. Como exemplo utiliza-se uma imagem contendo o PHP em sua versão 5.6 e o serviço do Apache. Com a imagem já definida, executa-se o comando a seguir para fazer o *download* da imagem.

```
$ docker pull php:5.6-apache
```

Onde, ‘*pull*’ é o parâmetro do *Docker* utilizado para fazer o *download* da imagem, ‘*php:5.6-apache*’, é a indicação da imagem localizada no *Docker Hub* que será baixada na sua respectiva versão.

Uma vez realizado o *download*, inicia-se a parte de execução do *container*. Para que esta ação seja realizada, é necessário que os arquivos do código-fonte da aplicação estejam armazenados em um diretório do computador *host*. Feito isso, executa-se o comando abaixo.

```
$ docker run -it -d -p 80:80 -v /root/aplicacao/:/var/www/html/ php:5.6-apache
```

Onde, ‘*run*’ é a ação do *Docker* utilizada para indicar que pretende-se iniciar a execução de um *container*, ‘*-it*’ são os parâmetros que definem que o *container* será interativo e terá um

terminal válido e acessível, respectivamente, `'-p 80:80'` é o parâmetro que indica que a porta 80 do servidor estará reservada para este *container*, e todo acesso que o *host* recebe nesta porta será direcionado para a porta 80 do *container*. `'-v /root/aplicacao/:/var/www/html/'` é o parâmetro que indica um volume do *host* para o *container*. Em outras palavras, é um diretório em comum entre as duas partes, onde `'/root/aplicacao/'` é o diretório do *host* e `'/var/www/html/'` é o diretório do *container* e `'php:5.6-apache'` é a imagem que será executada no *container*.

Agora, deve-se utilizar um outro computador-cliente que esteja na mesma rede do *host* e digitar o endereço IP fixado no servidor em um navegador do computador-cliente para testar o funcionamento do *container*.

Considera-se que, pela facilidade da implantação do ambiente de *containers*, é necessário um total de 10 dias de trabalho para deixar os serviços funcionando, levando em conta 7 dias variáveis para a entrega do computador, seguido da realização do curso disponibilizado no *site Udemy*, o que levaria aproximadamente 1 dia para ser concluído. Então inicia-se a parte da instalação e devidas configurações do sistema operacional para, posteriormente, realizar a instalação do *Docker* e configuração dos *containers*, o que levaria aproximadamente mais 2 dias de trabalho.

4.2 VIABILIDADE FINANCEIRA

O levantamento de viabilidade financeira presente no Quadro 1 foi realizado utilizando como base a média de valores de computadores com configurações semelhantes disponíveis para aquisição nos *sites* das empresas *Dell* e *Lenovo* (2019). O valor do curso foi retirado da plataforma de cursos *on-line* da *Udemy* (2019). As horas de trabalho foram calculadas com base na média de salário de um Técnico em Informática, obtido no *site Glassdoor* (2019).

Quadro 1 - Levantamento de custos de implantação de um ambiente de <i>containers</i>			
Produto	Quantidade	Valor unitário	Valor total
Computador	1	R\$ 2148,50	R\$ 2148,50
Curso na <i>Udemy</i>	1	R\$ 144,99	R\$ 144,99
Horas de trabalho	80	R\$ 10,43	R\$ 834,54
<i>GNU/Linux</i>	1	-	-
<i>Docker</i>	1	-	-
Dados obtidos em 23 Out. 2019		Valor total	R\$ 3128,03

Conforme apresentado no quadro 1, a implantação do sistema de *containers* não é um processo caro, ainda mais se considerar que a instituição de ensino já possui um computador que atenda aos requisitos necessários, pelo fato de possibilitar a unificação de todos os sistemas *Web* rodando em máquinas separadas em um mesmo servidor, o que descartaria a necessidade de aquisição de um computador específico para rodar o serviço de *containers*. Outro ponto que vale ressaltar é a existência de cursos sobre *Docker* disponibilizados de forma gratuita no *Youtube*, por exemplo.

5 CONCLUSÕES E CONSIDERAÇÕES FINAIS

Percebe-se a partir dos resultados apresentados que a implantação de um ambiente de *containers* é um processo simples e rápido que trará inúmeros benefícios para a instituição. Exemplos disso são a redução de espaço físico e de consumo de energia por parte das máquinas do CPD, bem como a facilitação do processo de restauração de serviço em caso de falha ou desastre que possa vir a causar sua parada repentina.

Neste sentido, este estudo aponta que a implantação de *containers* para hospedagem de serviços *Web* é viável na empresa estudada, pois possui uma boa relação custo-benefício, uma vez que, conforme já reiterado, as vantagens de sua utilização são inúmeras e o investimento para implantação é baixo, ainda mais se levar em consideração que a instituição teria uma máquina que atendesse aos requisitos necessários para a execução do serviço.

Desta forma, responde-se o problema de pesquisa “É viável instalar um sistema de *containers* para executar *softwares Web* em uma empresa do setor de educação?”, tendo como resultado a viabilidade da implantação da tecnologia proposta.

O objetivo geral de desenvolver um estudo de viabilidade de implantação de um sistema de gestão de *containers* em uma empresa da área da educação foi atingido por meio de pesquisas de viabilidade técnica e financeira e através de fatos importantes que evidenciam as vantagens da utilização desta tecnologia destacados por autores da área. Já o objetivo específico de se realizar um estudo teórico sobre os motivos para se utilizar *containers* foi atingido por meio de análises e estudos de publicações de autores que afirmam que um dos grandes motivos para a utilização desta tecnologia é a redução de espaço físico que os servidores utilizam no CPD, uma vez que é possível executar diversos serviços em um único *host*.

O objetivo específico de conhecer as vantagens em utilizar *containers* no lugar de máquinas virtuais foi atingido com base em estudos que comprovam inúmeros benefícios da

utilização desta tecnologia, como por exemplo, a facilidade na restauração de serviço em caso de desastre que venha a ocasionar sua parada.

O objetivo específico de aprofundar conhecimentos sobre configurações e funcionalidades dos *containers* foi atingido por meio de levantamentos de autores da área que explicam comandos e passos para a execução de um serviço sob a tecnologia estudada, conforme registrado no item 4.1.1 deste artigo.

O objetivo específico de analisar o custo-benefício em se utilizar *containers* em uma organização foi atingido através de pesquisas que apontam os requisitos para a execução da tecnologia assim como o custo de aquisição de equipamentos que atendam à demanda.

Por fim, a utilização de *containers* para a execução de serviços *Web* é uma tecnologia que surgiu para facilitar a vida dos profissionais de TI no que diz respeito à administração e implantação de servidores que objetivam a hospedagem de algum serviço. Conforme já comentado, tal afirmação é comprovada ao analisar as vantagens da implantação deste recurso que incluem sua grande portabilidade, espaço de armazenamento necessário reduzido, e facilidade na instalação.

A empresa alvo deste estudo, assim como várias outras empresas que utilizam serviços em execução através da *internet*, só têm a ganhar com a utilização de *containers*, uma vez que ocorrerá redução no consumo de energia, assim como o espaço físico utilizado pelas máquinas *host*. Sobretudo, a realização de um estudo sobre a presente tecnologia traz ao acadêmico uma visão mais ampla do que abordado comumente em sala de aula, uma vez que tal tecnologia é relativamente nova e possui grande tendência a ser fortemente utilizada. Esta probabilidade fortalece os conhecimentos do futuro profissional, ainda mais se o objetivo é atuar na área de infraestrutura, servidores e computação em nuvem.

6 REFERÊNCIAS

BRITO, Edivaldo. **O que é o IP? Descubra para que serve e qual é seu número.** 2013. Disponível em: <<https://www.techtudo.com.br/artigos/noticia/2013/05/o-que-e-o-ip-descubra-para-o-que-serve-e-qual-e-seu-numero.html>>. Acesso em: 31 out. 2019.

CASTRO, Marcus André Nunes; VITALINO, Jeferson Fernando Noronha. **Descomplicando o Docker.** Rio de Janeiro: Brasport Livros e Multimídia Ltda., 2018.

CHEE, Brian J. S.; FRANKLIN JUNIOR, Curtis. **Computação em Nuvem: Tecnologias e Estratégias.** São Paulo: M.books do Brasil Editora Ltda., 2013.

DEITEL, H. M.; DEITEL, P. J.; CHOFFNES, D. R.; **Sistemas Operacionais.** São Paulo: Pearson Prentice Hall, 2005.

uceff.edu.br

Centro Universitário FAI • |49| 3678.8700
Rua Carlos Kummer, 100
Bairro Universitário
Itapiranga - SC • 89896-000

Centro Politécnico • |49| 3319.3800
Av. Irineu Bornhausen, 2045 E
Bairro Quedas do Palmital
Chapecó - SC • 89814-650

Unidade Central • |49| 3319.3838
Rua Lauro Müller - 767 E
Bairro Santa Maria
Chapecó - SC • 89812-214

DELL. **Computadores Vostro**. 2019. Disponível em: <<https://www.dell.com/pt-br/work/shop/computadores-all-in-ones-e-workstations/sc/desktops-n-workstations/vostro-desktops>>. Acesso em: 23 out. 2019.

DOCKER. **What is a Container?:** A standardized unit of software. 2019. Disponível em: <<https://www.docker.com>>. Acesso em: 29 set. 2019.

GLASSDOOR. **Salários de Técnico De Informática**. 2019. Disponível em: <https://www.glassdoor.com.br/Salários/técnico-de-informática-salário-SRCH_KO0,22.htm>. Acesso em: 12 out. 2019.

KUROSE, James F.; ROSS, Keith W.. **Rede de Computadores e Internet: Uma nova abordagem**. São Paulo: Addison Wesley, 2003.

LENOVO. **ThinkCentre M720q Tiny**. 2019. Disponível em: <<https://www.lenovo.com/br/pt/desktops-y-all-in-one/thinkcentre/serie-m-tiny/ThinkCentre-M720q/p/10T8CTO1WWPTBR1>>. Acesso em: 23 out. 2019.

MASCARENHAS, Sidnei Augusto. **Metodologia Científica**. São Paulo: Pearson Education do Brasil, 2012.

PHP. **O que é o PHP?**. 2019. Disponível em: <https://www.php.net/manual/pt_BR/intro-what-is.php>. Acesso em: 31 out. 2019.

SEVERINO, Antônio Joaquim. **Metodologia do Trabalho Científico**. 23. ed. São Paulo: Cortez Editora, 2007.

SILVA, Wellington Figueira da. **Aprendendo Docker**. São Paulo: Novatec Editora Ltda., 2016.

UDEMY. **Docker do zero - introdução a administração de containers**. 2019. Disponível em: <<https://www.udemy.com/course/docker-introducao-a-administracao-de-containers/>>. Acesso em: 23 out. 2019.

VERAS, Manoel. Cloud Computing: **Nova Arquitetura da TI**. Rio de Janeiro: Brasport, 2012.

VERAS, Manoel. **Virtualização: Tecnologia Central do Datacenter**. 2. ed. São Paulo: Brasport, 2016.

WITTING, Andreas; WITTING, Michael. **Amazon Web Services em Ação**. São Paulo: Novatec Editora Ltda., 2016.