



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS QUIXADÁ
TECNÓLOGO EM REDES DE COMPUTADORES

ANTONIO CARLOS SILVA

**UMA FERRAMENTA PARA CRIAÇÃO DE *CLUSTERS* VIRTUAIS PARA
EXECUÇÃO DE APLICAÇÕES PARALELAS E DISTRIBUÍDAS EM NUVENS
PÚBLICAS**

QUIXADÁ

2021

ANTONIO CARLOS SILVA

UMA FERRAMENTA PARA CRIAÇÃO DE *CLUSTERS* VIRTUAIS PARA EXECUÇÃO DE
APLICAÇÕES PARALELAS E DISTRIBUÍDAS EM NUVENS PÚBLICAS

Monografia apresentada ao curso de Redes de Computadores da Universidade Federal do Ceará, como requisito parcial à obtenção do título de bacharel em Redes de Computadores. Área de concentração: Computação.

Orientador: Prof. Dr. João Marcelo Uchôa de Alencar

QUIXADÁ

2021

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- S578f Silva, Antonio Carlos.
Uma ferramenta para criação de clusters virtuais para execução de aplicações paralelas e distribuídas em nuvens públicas / Antonio Carlos Silva. – 2021.
41 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá, Curso de Redes de Computadores, Quixadá, 2021.
Orientação: Prof. Dr. João Marcelo Uchôa de Alencar.
1. Computação em nuvem. 2. Infraestrutura como serviço. 3. Cluster (Sistema de computador). I. Título.
CDD 004.6
-

ANTONIO CARLOS SILVA

UMA FERRAMENTA PARA CRIAÇÃO DE *CLUSTERS* VIRTUAIS PARA EXECUÇÃO DE
APLICAÇÕES PARALELAS E DISTRIBUÍDAS EM NUVENS PÚBLICAS

Monografia apresentada ao curso de Redes de Computadores da Universidade Federal do Ceará, como requisito parcial à obtenção do título de bacharel em Redes de Computadores. Área de concentração: Computação.

Aprovada em: ____/____/____

BANCA EXAMINADORA

Prof. Dr. João Marcelo Uchôa de Alencar (Orientador)
Universidade Federal do Ceará – UFC

Prof. Dr. Jefferson de Carvalho Silva
Universidade Federal do Ceará - UFC

Prof. Dr. Wagner Guimarães Al-Alam
Universidade Federal do Ceará - UFC

À minha família, por acreditarem no meu sonho.
À minha irmã Valdira Carlos da Silva, pelo s
cuidado, dedicação e pela esperança que me deu
pr alcançar essa conquista, que não é só minha,
mas dela também.

AGRADECIMENTOS

À Deus, por ter me dado força para continuar e poder concluir este trabalho.

À minha família, pelo apoio nessa jornada. Em especial a minha irmã Valdira, que sempre esteve do meu lado e me presenteou com um notebook para me ajudar nas atividades diárias durante a graduação.

Aos meus professores, em especial o meu orientador, Prof. Dr. João Marcelo Uchôa de Alencar, pelo seu excelente trabalho, por todos os conselhos, pela paciência e motivação que me deu durante esse período.

Aos meus amigos de faculdade, Jardel Gonçalves, Jefte Souza, Esdras Emanuel e Matheus Cavalcante, pelo companheirismo em vários trabalhos e momentos que contribuíram na minha formação. Pessoas que com certeza vão continuar presentes na minha vida.

Aos meus amigos de infância, em especial Alisson Lima, Alan Braúna, Alan Nascimento, Leandro Cardoso, Iris Leandro, Natalia Fernandes e Monalisa Duarte, que estiveram comigo durante todo esse tempo, nos momentos bons e ruins.

“A persistência é o caminho do êxito.”

(Charles Chaplin)

RESUMO

A computação em nuvem é uma tecnologia que permite a distribuição dos seus serviços de computação, fornecendo recursos através dos modelos de implantação. Os serviços oferecidos pelas nuvens são parte da infraestrutura da Internet, tornando-se um diferencial competitivo para o uso em aplicações distribuídas. Pensando nisso, o presente trabalho tem o objetivo de desenvolver uma ferramenta que permita selecionar entre duas ou mais opções de nuvens públicas para a criação de um *cluster* virtual voltado para a programação de aplicações paralelas e distribuídas, atuando no nível IaaS de uma nuvem pública. Para validar as funcionalidades da ferramenta, foi realizada um exercício de programação no *cluster* criado, após a execução do exercício foi aplicado o questionário para os alunos e egressos, com o objetivo de realizar uma avaliação qualitativa da ferramenta proposta com o intuito de obter resultados a respeito da usabilidade da ferramenta. Com os resultados consolidados realizamos o tratamento dos dados obtidos na avaliação. Concluimos com este trabalho, que a ferramenta desenvolvida é capaz de criar *clusters* virtuais nas plataformas de nuvens públicas abordada através do modelo de serviço IaaS, no qual o usuário faz uso de infraestruturas virtuais para práticas de programação paralelas e distribuídas.

Palavras-chave: Computação em nuvem. Infraestrutura como serviço. Cluster (Sistema de computador).

ABSTRACT

Cloud computing is a technology that allows the distribution of its computing services, providing computational resources, through the deployment models for a computing cloud, the services offered by clouds are part of the base infrastructure of the Internet, becoming a competitive differential for use in distributed applications. With this in mind, the present work has the objective of developing a tool that allows selecting between two or more options of public clouds for the creation of a virtual cluster focused on the programming of parallel and distributed applications, acting at the IaaS level of a public cloud. To validate the functionalities of the tool, a programming exercise was carried out in the created cluster, after the exercise was carried out, the questionnaire was applied to students and alumni, in order to carry out a qualitative assessment of the proposed tool in order to obtain results from regarding the usability of the tool. With the consolidated results, we carried out the treatment of the data obtained in the evaluation. We concluded with this work, that the developed tool is capable of creating virtual clusters on public cloud platforms approached through the IaaS service model, in which the user makes use of virtual infrastructures for parallel and distributed programming practices.

Keywords: Cloud computing. Infrastructure as a service. Cluster (computer system).

SUMÁRIO

1	INTRODUÇÃO	10
1.1	Objetivos	11
1.1.1	<i>Objetivo Geral</i>	11
1.1.2	<i>Objetivos Específicos</i>	11
1.1.3	<i>Visão da Arquitetura</i>	11
2	TRABALHOS RELACIONADOS	13
3	FUNDAMENTAÇÃO TEÓRICA	15
3.1	Cluster	15
3.2	Computação em Nuvem	16
3.2.1	<i>Modelos de Implantação</i>	16
3.2.2	<i>Nuvem Pública</i>	16
3.2.3	<i>Nuvem Privada</i>	17
3.2.4	<i>Nuvem Híbrida</i>	18
3.2.5	<i>Modelos de Serviços</i>	18
3.2.6	<i>SaaS - (Software as a Service)</i>	19
3.2.7	<i>PaaS - (Platform as a Service)</i>	20
3.2.8	<i>IaaS - (Infrastructure as a Service)</i>	20
3.2.9	<i>AWS Cloud Formation</i>	21
3.2.10	<i>Azure Resource Manager</i>	22
4	PROCEDIMENTOS METODOLÓGICOS	25
4.1	Levantamento de nuvens públicas com créditos acadêmicos.	25
4.2	Configuração das ferramentas de linha de comando das nuvens selecionadas.	25
4.3	Definir os parâmetros para criação de um <i>cluster</i> virtual.	25
4.4	Realizar testes de utilização da ferramenta.	26
5	CONFIGURAÇÃO DA FERRAMENTA	27
5.1	Preparando o Ambiente para Execução	27
5.1.1	<i>Linha de Comando da AWS</i>	27
5.1.2	<i>Linha de Comando da Azure</i>	28
5.2	Opções de Execução do <i>Script</i>	28
5.2.1	<i>Scripts</i>	28

5.2.2	<i>Funções dos Scripts</i>	29
6	AVALIAÇÃO DA FERRAMENTA	30
6.1	Descrição da avaliação	30
6.2	Metodologia de avaliação	30
6.3	Descrição do Exercício	31
6.4	Resultados e Discussões	32
7	CONCLUSÃO	39
	REFERÊNCIAS	40

1 INTRODUÇÃO

A computação em nuvem é um modelo de serviço que possibilita o acesso a diversos recursos computacionais, como servidores, redes, aplicações e serviços por meio da Internet. A computação em nuvem vem crescendo cada vez mais nos últimos anos e os serviços oferecidos pelas nuvens são parte da infraestrutura base da Internet, tornando-se um diferencial competitivo para o uso em aplicações distribuídas. Nas últimas décadas, diferentes infraestruturas de computação foram usadas para fornecer poder de computação para a comunidade científica (CABALLER et al., 2014).

Dois usos consolidados das nuvens computacionais são o fornecimento de armazenamento sob demanda e poder de processamento. Este ambiente permite a locação de recursos para aprimorar a capacidade computacional localmente disponível, atribuindo novos recursos computacionais quando necessário. Em uma nuvem, o usuário acessa recursos de computação como utilitários gerais que podem ser concedidos e liberados (CHOPRA; SINGH, 2013).

A computação em nuvem é um modelo para permitir acesso à rede onipresente, conveniente e sob demanda a um conjunto compartilhado de recursos de computação configuráveis (por exemplo, redes, servidores, armazenamento, aplicativos e serviços) que podem ser rapidamente provisionados e liberados com esforço mínimo de gerenciamento ou interação entre prestadores de serviços. A computação em nuvem fornece basicamente três tipos de serviços: Software como Serviço (SaaS), Plataforma como Serviço (PaaS) e Infraestrutura como Serviço (IaaS) (SADASHIV; KUMAR, 2011).

Para realizar o desenvolvimento de aplicações distribuídas em um cenário realista, é necessário que os pesquisadores consigam interligar estações de trabalhos disponíveis nos laboratórios de forma que venham a montar um *cluster* improvisado. Realizar tais atividades ocasionam em custo de manutenção de hardware além das configurações de software. A confiabilidade e disponibilidade dessas atividades fora do ambiente das nuvens ocasionam baixa produtividade Caballer et al. (2014). Tendo como uma das principais desvantagens dessas plataformas de computação é o investimento inicial relativamente grande junto com o custo de manutenção. Para grupos ou organizações de pesquisa de pequeno e médio porte, a compra de tal equipamento pode representar um custo importante (CABALLER et al., 2013).

Diante dessa problemática, este trabalho tem como propósito desenvolver uma ferramenta para criação de um *cluster virtual*, no qual o usuário fará uso de infraestruturas

virtuais para a prática de programação paralela e distribuída com foco na integração de duas ou mais nuvens públicas. A ferramenta desenvolvida vai trabalhar em cima de dois serviços principais para criação do cluster virtual, na AWS é utilizado o *AWS CloudFormation* e na Azure o *Azure Resource Manager*, esses serviços oferecem a possibilidade de implantar a infraestrutura como código (IaC) para um melhor gerenciamento da infraestrutura. Isso abre caminho para que os usuários acessem facilmente vastos recursos de computação sob demanda com investimento mínimo na portabilidade de aplicativos.

1.1 Objetivos

A seguir, o objetivo geral e os produtos intermediários ou colaterais esperados.

1.1.1 Objetivo Geral

Desenvolver uma ferramenta que permita selecionar entre duas ou mais opções de nuvens públicas para a criação de um *cluster virtual* voltado para a programação de aplicações paralelas e distribuídas.

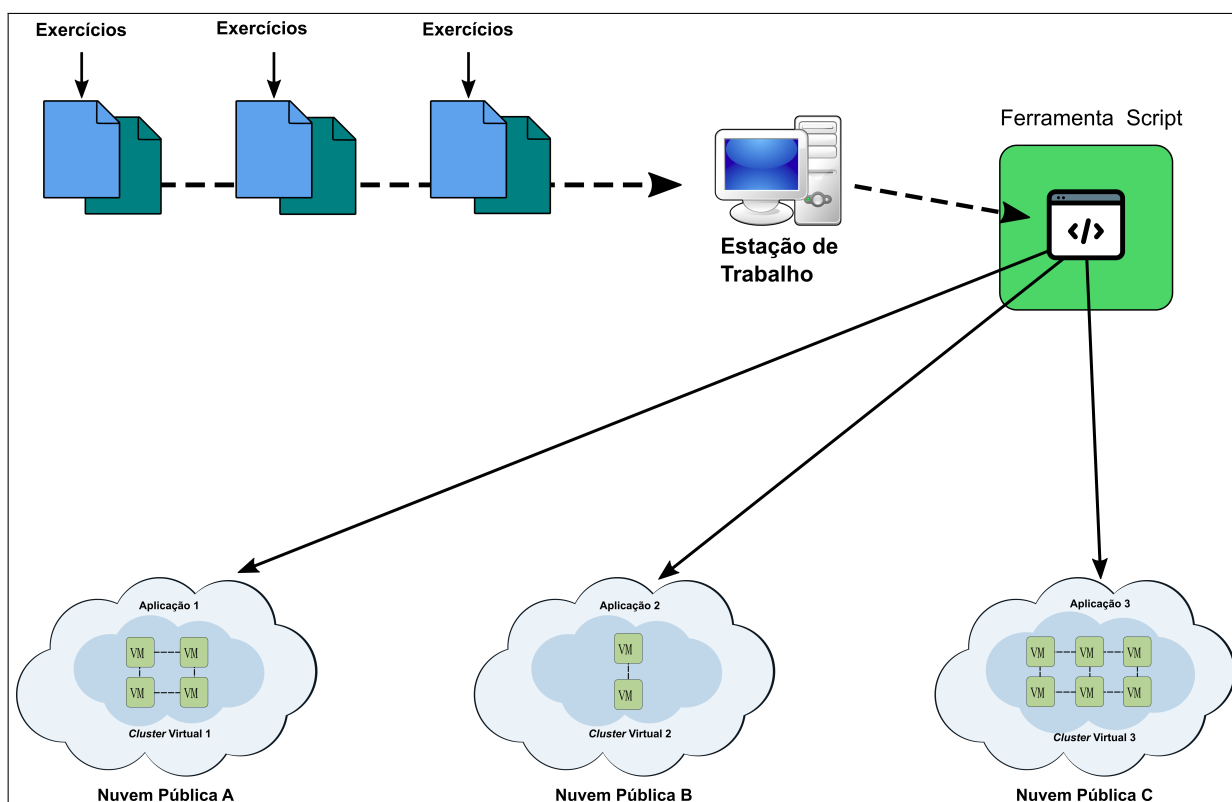
1.1.2 Objetivos Específicos

- Levantar provedores de nuvens públicas indicados para uso acadêmico, que apresentem algum nível de gratuidade.
- Implementação de uma ferramenta no formato de *shell script* para instanciação do *cluster* de acordo com a plataformas disponíveis.
- Garantir que o aluno tenha possibilidade de dimensionar os recursos do *cluster* de acordo com as necessidades de sua atividade ou exercício.
- Realizar testes de utilização da ferramenta.

1.1.3 Visão da Arquitetura

A Figura 1 é uma representação do ambiente proposto neste trabalho, mostrando como será feita a interação da ferramenta com as nuvens públicas. Dessa forma, na arquitetura proposta, podemos observar como o aluno irá criar o ambiente a partir da sua estação de trabalho, para que em seguida o *script* aloque o *cluster* virtual nas nuvens públicas.

Figura 1 – Arquitetura Proposta



Fonte – Elaborada pelo autor

2 TRABALHOS RELACIONADOS

A computação em nuvem fornece recursos sob demanda para requisitos de computação e armazenamento. A nuvem híbrida é um ambiente de computação que combina nuvens públicas e privadas, permitindo que os dados e aplicativos sejam compartilhados entre elas. Nesse sentido, nesta seção, será abordados alguns trabalhos que utilizam dos recursos de nuvens públicas e híbridas para eliminar custos de capitais e reduzir custos contínuos.

No trabalho de (CHOPRA; SINGH, 2013), os autores fazem uso da nuvem híbrida com o objetivo de executar diversas tarefas utilizando uma abordagem em que caso os recursos da nuvem privada não sejam suficientes, os recursos da nuvem pública sejam utilizados. Além disso, os autores desenvolveram um algoritmo de escalonamento baseado no nível que são executadas as tarefas e conceitos como prazo que visam descobrir os melhores recursos da nuvem pública buscando reduzir os custos e finalizar a execução do fluxo dentro dos prazos.

Semelhante ao trabalho mencionado anteriormente, em nosso trabalho realiza a execução de atividades paralelas e distribuídas na nuvem, porém se difere no fato que é utilizado nuvens públicas, com o propósito de tirar proveitos dos recursos fornecidos pelas nuvens públicas, ou seja, criar um *cluster* virtual em uma das várias opções de nuvens públicas.

Referente ao trabalho de (CABALLER et al., 2014), os autores apresentam uma plataforma que permite a execução de aplicações em nuvem híbrida fornecendo uma linguagem para declarar os requisitos da aplicação. Na arquitetura proposta pelos autores são utilizado contêineres para encapsular toda a infraestrutura necessária para executar na nuvem. Cada aplicativo lançado tem seu próprio contêiner que gerencia todo o ciclo de vida da execução da nuvem (evitando problemas de multi-hospedagem, como segurança ou contabilidade).

Neste trabalho é feito uso de nuvens públicas, para tirar proveito de todos os recursos fornecidos pelas mesmas para implementar uma ferramenta ou *framework* que facilite a criação de infraestruturas virtuais para a prática de programação paralela e distribuída com foco na integração de duas ou mais nuvens públicas auxiliando na formação do corpo discente.

Relacionado ao trabalho de (RIGHI et al., 2015), os autores apresenta um modelo de elasticidade que se concentra em aplicativos de alto desempenho. Sua abordagem diferencial consiste em fornecer elasticidade para aplicativos de alto desempenho sem intervenção do usuário. Dessa forma, sua abordagem se aplica de duas maneiras, controle eficiente do lançamento e consolidação de VMs(máquinas virtuais) totalmente transparente para o usuário e um mecanismo para executar programas HPC(computação de alto desempenho) na nuvem.

Semelhante ao trabalho mencionado anteriormente, este trabalho foi realizado a execução de atividades paralelas e distribuídas na nuvem, porém se difere no fato que utilizado modelos de serviços diferente. No caso do trabalho de (RIGHI et al., 2015) utiliza um modelo de elasticidade baseado em *PaaS* para executar programas HPC na nuvem, porém se difere no fato que será utilizado o modelo de serviço *IaaS* para criação de um *cluster* virtual em uma das várias opções de nuvens públicas.

A tabela 1 apresenta um comparativo dos trabalhos citados acima.

Tabela 1 – Comparativo entre trabalhos

Trabalho	IaaS	PaaS	SaaS	Nuvem Publica	Nuvem Privada	Nuvem Hibrida
Caballer et al. (2014)	SIM	SIM	NÃO	NÃO	NÃO	SIM
Chopra e Singh (2013)	SIM	NÃO	NÃO	NÃO	NÃO	SIM
Righi et al. (2015)	NÃO	SIM	NÃO	NÃO	SIM	NÃO
Este Trabalho	SIM	NÃO	NÃO	SIM	NÃO	NÃO

Fonte – Elaborada pelo autor

3 FUNDAMENTAÇÃO TEÓRICA

Nesta Seção será apresentada uma visão geral sobre os conceitos abordados neste trabalho. Na Seção 3.1 é apresentada conceitos sobre a tecnologia utilizada para preparar o ambiente, abordando funcionalidades do trabalho proposto. Na Seção 3.2 é apresentado conceitos sobre computação em nuvem, desde seus benefícios, até informações sobre os principais recursos que será utilizados neste trabalho.

3.1 Cluster

Cluster é um termo em inglês que significa “aglomerar” ou “aglomeração” e pode ser aplicado em vários contextos. No caso da computação, o termo define uma arquitetura de sistema capaz de combinar vários computadores para trabalharem em conjunto ou pode denominar o grupo em si de computadores combinados. Os *clusters* de computadores processam as tarefas paralelas de forma transparente, ou seja, aparentando ser um único sistema para o usuário. Neste projeto tem como finalidade montar um *cluster* em umas das opções de nuvem públicas fornecidas (BACELLAR, 2009).

Os *clusters* são direcionados principalmente para aplicações bastante exigentes no que diz respeito ao processamento, para fins de computação de alto desempenho. Dessa forma, para aproveitar os recursos fornecidos pelas nuvens públicas será criado um *cluster virtual* em uma das nuvens públicas com o propósito de facilitar a execução de programações paralelas e distribuídas (SADASHIV; KUMAR, 2011).

Quando vários computadores são vinculados em um *cluster*, eles compartilham a carga de trabalho computacional como um único computador virtual. Do ponto de vista do usuário, são várias máquinas, mas funcionam como uma única máquina virtual. Tornando possível a distribuição de processo e tarefas entre várias máquinas, fazendo com que a execução de aplicações tenham um maior desempenho na execução. A solicitação do usuário é recebida e distribuída entre todos os computadores independentes para formar um *cluster*. Isso resulta em trabalho computacional equilibrado entre diferentes máquinas, melhorando o desempenho dos sistemas de *cluster* (SADASHIV; KUMAR, 2011).

3.2 Computação em Nuvem

A computação em nuvem é uma forma de se referir ao uso de recursos de computação compartilhada fornecendo serviços de computação. A computação em nuvem agrupa um grande número de servidores de computação e outros recursos e normalmente oferece sua capacidade combinada sob demanda e pagamento por ciclo. Desta maneira, tornando possível utilizar recursos da nuvens públicas através do *pay per use* oferece a possibilidade de contratação de um serviço que é ajustável às suas necessidades. Ou seja, em vez de pagar um valor fixo para utilizá-lo durante todo o mês, o contratante arca apenas com os gastos relativos aos recursos fornecidos pelas nuvens (ARMBRUST et al., 2010).

O modelo de computação em nuvem tem como finalidade fornecer serviços de fácil acesso, baixo custo e com atribuições de disponibilidade e escalabilidade. Esse modelo tende a fornecer, benefícios computacionais sob demanda. Assim, diminuir o custo na aquisição e composição de toda infraestrutura requisitada para atender as necessidades do usuário, podendo essa infraestrutura ser mesclada sob demanda e com recursos diversificados e de menor custo e proporcionar uma abstração e facilidade de acesso aos usuários deste serviços(SOUSA; MOREIRA; MACHADO, 2009).

3.2.1 Modelos de Implantação

Na definição de uma infraestrutura de nuvem é preciso definir quem é responsável pela gerência dos recursos, quais usuários tem acesso e como está definida a interface de interação entre esses atores. Para entender esse cenário, apresentamos os modelos de implantação para uma nuvem computacional.

3.2.2 Nuvem Pública

No modelo de implantação de nuvem pública, a infraestrutura de nuvens é disponibilizada para o público em geral, sendo acessado por qualquer usuário que conheça a localização do serviço. Os recursos são oferecidos como um serviço, geralmente por meio de uma conexão com a Internet, por uma taxa de pagamento por uso. Os usuários podem dimensionar seu uso sob demanda e não precisam comprar *hardware* para usar o serviço (SOUSA; MOREIRA; MACHADO, 2009), eliminando custos de manutenção e eletricidade, além de exigir menos pessoal especializado.

Os provedores de nuvem pública gerenciam a infraestrutura e agrupam recursos na capacidade exigida por seus usuários. As infraestruturas públicas estão disponíveis para o público em geral ou para grandes organizações e são de propriedade de uma organização de terceiros que oferece o serviço de nuvem. Sua hospedagem na Internet é projetada para ser usada por qualquer usuário com uma conexão à Internet para fornecer uma gama semelhante de recursos e serviços (GOYAL, 2014).

Com o crescente número de provedores de nuvem, em especial a AWS¹ e Azure² na qual fornecem uma variedade de serviços, surge uma nova possibilidade para ambiente de aprendizado para *frameworks* de aplicações distribuídas. Utilizando-se apenas de ferramentas de software, o usuário pode criar infraestruturas virtuais complexas para testar suas aplicações, sendo que tais infraestruturas não diferem em nada das soluções que são usadas na indústria e academia para execução de aplicações.

Apesar da facilidade que trazem, as nuvens públicas apresentam um custo financeiro. Tal custo não costuma ultrapassar alguns centavos de dólar por hora de utilização de uma máquina virtual, porém cenários mais complexos precisam utilizar várias instâncias. Porém, os principais provedores de nuvens públicas fornecem créditos gratuitos para estudantes e professores universitários. Por exemplo, a AWS, além do nível gratuito, fornece \$100 anuais para estudantes através do programa AWS Educate³.

3.2.3 *Nuvem Privada*

As nuvens privadas estão dentro do *datacenter* interno de uma organização. Desta forma, o gerenciamento da segurança é sua principal vantagem, além de fornecer mais controle de implantação e o uso. Em comparação com nuvem pública, na qual todos os recursos e aplicativos eram gerenciados pelo provedor de serviços, esses serviços são agrupados e disponibilizados para os usuários no nível organizacional. Dessa forma, os recursos e aplicativos são gerenciados pela própria organizações (JADEJA; MODI, 2012).

A motivação para configurar uma nuvem privada dentro de uma organização tem vários aspectos. Para maximizar e otimizar a utilização de recursos internos existentes, além das preocupações de segurança, incluindo privacidade de dados e confiança, também tornam a nuvem privada uma opção para muitas empresas (DILLON; WU; CHANG, 2010).

¹ <https://aws.amazon.com/pt/>

² <https://azure.microsoft.com/pt-br/>

³ <https://aws.amazon.com/pt/education/awseducate/>

3.2.4 Nuvem Híbrida

A infraestrutura em nuvem híbrida é uma combinação de duas nuvens que permanecem como entidades únicas, As nuvens híbridas são mais complexas do que os outros modelos de implantação, pois envolvem uma composição de duas ou mais nuvens (privada e pública). As organizações usam o modelo de nuvem híbrida para otimizar seus recursos, a fim de aumentar suas competências essenciais, isolando as funções de negócios periféricos na nuvem e, ao mesmo tempo, controlando as principais atividades no local por meio da nuvem privada (DILLON; WU; CHANG, 2010).

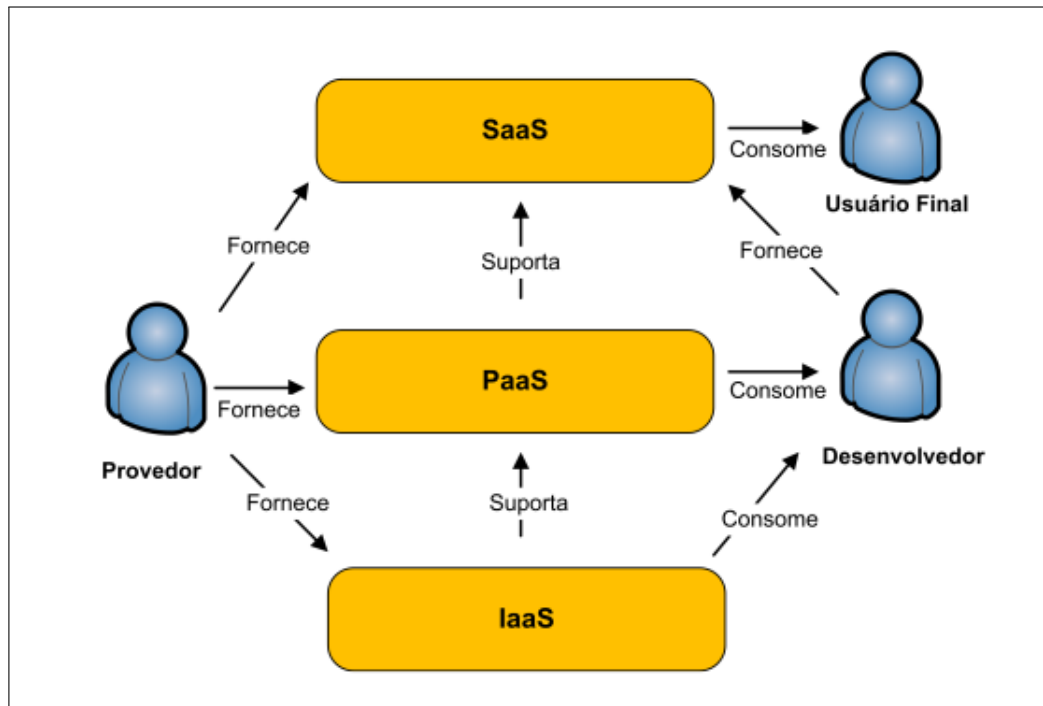
As nuvens híbridas oferecem maior flexibilidade às empresas, oferecendo opções em termos de manter o controle e a segurança. Geralmente são implementadas por organizações dispostas a enviar parte de suas cargas de trabalho para nuvens públicas, seja para fins de estouro de nuvem ou para projetos que exigem implementação mais rápida. Como os ambientes híbridos envolvem provedores de nuvem pública e Privada, algumas considerações adicionais de segurança de infraestrutura entram em cena, que normalmente estão associadas a nuvens públicas, já que no ambiente de uma nuvem privada há um controle maior sobre a implantação e segurança (GOYAL, 2014).

3.2.5 Modelos de Serviços

Uma vez que apresentamos os modelos de implantação, vamos agora discutir como o usuário interage com nuvens, em qual nível de abstração ele atua e como são estruturadas as interfaces dos serviços oferecidos.

Os papéis são essenciais para definir atribuições, acesso e perfil para os diferentes usuários que fazem parte e estão envolvidos. Para entender melhor a computação em nuvem, pode-se classificar os atores dos modelos de acordo com os papéis desempenhados (SOUSA; MOREIRA; MACHADO, 2009). O provedor de nuvem pública disponibiliza o gerenciamento e monitoramento de toda a estrutura da computação em nuvem, onde tal responsabilidade não cabe ao desenvolvedor e usuário final, fornecendo os três modelos de serviços. Os desenvolvedores utilizam os recursos fornecidos e disponibilizam serviços para os usuários finais. A Figura 2 destaca estes papéis.

Figura 2 – Papéis na Computação em Nuvem



Fonte – (SOUSA; MOREIRA; MACHADO, 2009)

3.2.6 SaaS - (Software as a Service)

O modelo de SaaS possibilita sistema de *software* com objetivos específicos que estão á disposição dos usuários através da internet. Os sistemas de *software* estão a disposição do usuário através de dispositivos por meio de computadores, a qualquer hora em qualquer lugar, por um navegador *Web*. No modelo SaaS, o usuário não tem controle administrativo sobre a infraestrutura subjacente, incluindo rede, servidores, sistemas operacionais, armazenamento ou mesmo as características individuais da aplicação, exceto configurações específicas. Dessa maneira, os desenvolvedores têm como objetivo a inovação e não a infraestrutura, proporcionando um desenvolvimento rápido de sistemas de *software*(SOUSA; MOREIRA; MACHADO, 2009).

Um provedor de SaaS geralmente hospeda e gerencia um determinado aplicativo em seu próprio data center e o disponibiliza para vários locatários e usuários pela *Web*. Alguns provedores de SaaS são executados em ofertas de serviço PaaS ou IaaS de outro provedor de nuvem. Desta modo, não é necessário instalar, configurar nem dar suporte aos aplicativos.

O provedor se responsabiliza por todas essas tarefas, inclusive pelas atualizações e correções (BHARDWAJ; JAIN; JAIN, 2010).

3.2.7 PaaS - (Platform as a Service)

A PaaS disponibiliza de uma infraestrutura de alto nível de agregação para implementar e realizar testes de aplicações na nuvem. O usuário não tem controle sobre a infraestrutura subjacente, incluindo rede, servidores, sistemas operacionais ou armazenamento, mas tem autoridade sobre as aplicações implantadas e as configurações das aplicações instaladas na infraestrutura. A PaaS oferece um sistema operacional, linguagens de programação e ambientes de desenvolvimento para as aplicações, contribuindo para implementação de sistemas de *software*, já que contém ferramentas de desenvolvimento e colaboração entre desenvolvedores(SOUSA; MOREIRA; MACHADO, 2009).

Essa plataforma consiste em software de infraestrutura e geralmente inclui ferramentas de banco de dados, *middleware* e desenvolvimento. Uma arquitetura de computação em grade virtualizada e agrupada geralmente é a base para esse software de infraestrutura. Algumas ofertas de PaaS têm uma linguagem de programação ou API(*Application Programming Interface*) específica. Dessa maneira, o usuário terá acesso a uma plataforma para desenvolver suas aplicações sem se preocupar com servidores, *framework*, SO, ou infraestrutura facilitando e tornando mais ágil o desenvolvimento das aplicações (BHARDWAJ; JAIN; JAIN, 2010).

3.2.8 IaaS - (Infrastructure as a Service)

Infraestrutura como serviço é uma forma de hospedagem, incluindo acesso à rede, serviços de roteamento e armazenamento. O provedor de IaaS geralmente fornece os serviços administrativos e de *hardware* necessários para armazenar aplicativos e uma plataforma para execução de aplicativos. A principal vantagem é poder usufruir de um serviço superior por uma fração do custo da implantação da mesma funcionalidade através de servidores próprios, já que serviços como manutenção e gerenciamento desses servidores são feitos diretamente no serviço oferecido pelo seu provedor de IaaS (BHARDWAJ; JAIN; JAIN, 2010).

A capacidade prevista para o consumidor é a prestação de transformação, armazenamento, redes e outros recursos computacionais fundamental que o consumidor seja capaz de implantar e executar programas dispensáveis, que podem incluir sistemas operacionais e aplicativos. Dessa forma, IaaS fornece toda a infraestrutura para para montar um *cluster*

virtual para executar as aplicações (RUSCHEL; ZANOTTO; MOTA, 2010).

O IaaS traz os serviços oferecidos na camada de infraestrutura, dentre os quais podemos incluir a virtualização servidores, roteadores, sistemas de armazenamento e outros recursos de computação. O IaaS traz algumas características, como uma interface única para administração da infraestrutura, a aplicação API para interação com *hosts*, *switches*, roteadores e o suporte para a adicionar novos equipamentos de forma simples e transparente. Oferecendo um praticidade na hora de preparar o ambiente, já que o usuário não terá que se preocupar com a configuração dos nós dos *cluster* (RUSCHEL; ZANOTTO; MOTA, 2010).

3.2.9 AWS Cloud Formation

O AWS CloudFormation é um serviço que ajuda você a modelar e configurar seus recursos da Amazon Web Services para despende menos tempo gerenciando esses recursos e mais tempo se concentrando em seus aplicativos executados AWS. O CloudFormation usa modelos de implantação que são arquivos em formato JSON e YAML, usando esses modelos de arquivos como base para criar os recursos na AWS.

Ao usar o AWS CloudFormation, você gerencia recursos relacionados como uma unidade única chamada de pilha. Todos os recursos baseados no modelo de implantação são criados nessa pilha, podendo assim atualizar e excluir esses recursos excluindo a pilha.

O modelo a seguir cria uma pilha contendo um servidor EC2.

Algoritmo 1 - Modelo AWS CloudFormation

```

1 {
2   "AWSTemplateFormatVersion" : "2010-09-09",
3   "Description" : "A sample template",
4   "Resources" : {
5     "MyEC2Instance" : {
6       "Type" : "AWS::EC2::Instance",
7       "Properties" : {
8         "ImageId" : "ami-0ff8a91507f77f867",
9         "InstanceType" : "t2.micro",
10        "KeyName" : "testkey",
11        "BlockDeviceMappings" : [

```

```
12     {
13         "DeviceName" : "/dev/sd1",
14         "Ebs" : {
15             "VolumeType" : "gp2",
16             "Iops" : "200",
17             "DeleteOnTermination" : "false",
18             "VolumeSize" : "20"
19         }
20     }
21 ]
22 }
23 }
24 }
25 }
```

Fonte – (AMAZON, 2021)

3.2.10 Azure Resource Manager

O Azure Resource Manager é o serviço de implantação e gerenciamento do Azure. Ele fornece uma camada de gerenciamento que lhe permite criar, atualizar e excluir recursos em sua conta do Azure. Para implementar a infraestrutura como código para suas soluções do Azure, use modelos de Azure Resource Manager (modelos ARM). O modelo é um arquivo (JSON) que define a infraestrutura e a configuração de sua aplicação.

Assim como na AWS o modelo cria uma pilha, no ARM ele vai criar um resource group (Grupo de Recursos) onde vão está todos os recursos que foram definidos no modelo de implementação ARM. Todos os recursos do grupo de recursos devem compartilhar o ciclo de vida. Você os implanta, atualiza e exclui juntos.

O modelo ARM a seguir cria um servidor em um grupo de recursos.

Algoritmo 2 - Modelo Azure Resource Manager

```
1 {
```



```
2  "$schema": "https://schema.management.azure.com/schemas
    /2019-04-01/deploymentTemplate.json#",
3  "contentVersion": "1.0.0.0",
4  "parameters": {
5    "storageAccountType": {
6      "type": "string",
7      "defaultValue": "Standard_LRS",
8      "allowedValues": [
9        "Standard_LRS",
10       "Standard_GRS"
11     ],
12     "metadata": {
13       "description": "Storage Account type"
14     }
15   },
16   "location": {
17     "type": "string",
18     "defaultValue": "[resourceGroup().location]",
19     "metadata": {
20       "description": "Location for all resources."
21     }
22   },
23   "namePrefix": {
24     "type": "string",
25     "maxLength": 11,
26     "defaultValue": "store",
27     "metadata": {
28       "description": "Prefix for storage account name"
29     }
30   }
31 },
32 "variables": {
```

```
33     "storageAccountName": "[concat(parameters ( namePrefix ) ,  
        uniquestring(resourceGroup().id))]"  
34 },  
35 "resources": [  
36     {  
37         "type": "Microsoft.Storage/storageAccounts",  
38         "apiVersion": "2019-06-01",  
39         "name": "[variables ( storageAccountName ) ]",  
40         "location": "[parameters ( location ) ]",  
41         "sku": {  
42             "name": "[parameters ( storageAccountType ) ]"  
43         },  
44         "kind": "StorageV2",  
45         "properties": {}  
46     }  
47 ],  
48 "outputs": {  
49     "storageAccountName": {  
50         "type": "string",  
51         "value": "[variables ( storageAccountName ) ]"  
52     }  
53 }  
54 }
```

4 PROCEDIMENTOS METODOLÓGICOS

Apresentamos nesta seção as etapas cumpridas para cada objetivo específico, em conjunto elas resultam na solução que atende o objetivo geral do nosso trabalho.

4.1 Levantamento de nuvens públicas com créditos acadêmicos.

Após um levantamento, considerando as nuvens públicas mais utilizadas, concluímos que a maioria oferece um período de uso gratuito que pode se estender até um ano. Porém, apenas a AWS, através do programa *AWS Educate*¹, e a Azure, através do e-mail institucional da UFC, oferecem a possibilidade de créditos recorrentes ao corpo discente. Portanto, foram as alternativas utilizadas neste trabalho.

4.2 Configuração das ferramentas de linha de comando das nuvens selecionadas.

Dada a opção de desenvolver a solução através da linguagem *shell script*, visto que a mesma está presente na maioria dos sistemas Linux instalados nos laboratórios do Campus, passamos então a estudar as ferramentas de linha de comando disponível para cada nuvem. Apesar das particularidades de cada uma, percebemos um ponto em comum: a configuração e cadastro de credenciais na estação de trabalho que irá executar os comandos. Portanto, foi realizado um levantamento do passo a passo para configuração do ambiente tanto para a AWS quanto para a Azure. Este é um ponto ainda específico de cada nuvem que nossa solução não conseguiu generalizar. Entretanto, como veremos adiante ao descrever a utilização da nossa ferramenta, uma vez configurado o acesso a cada nuvem, o *script* se comporta de maneira similar na criação de *clusters*, seja qual for a plataforma.

4.3 Definir os parâmetros para criação de um *cluster* virtual.

Uma vez que o *script* busca atender a necessidade de disciplinas distintas como Computação Paralela, Redes de Computadores e Sistemas Distribuídos, se faz necessário controlar o tamanho do *cluster*. Por exemplo, em Computação Paralela, a criação de *clusters* com mais de 2 nós é necessária para estudar a escalabilidade de algoritmos paralelos. Entretanto, em Redes de Computadores, o estudo de programação em *sockets* necessita de

¹ <https://aws.amazon.com/pt/education/awseducate/>

apenas duas máquinas. Portanto, no *script* apresenta parâmetros para escolher o **provedor** de nuvem e o **tamanho** do *cluster* a ser criado. Um terceiro parâmetro é utilizado para definir o **tipo** da configuração do nó do *cluster*.

4.4 Realizar testes de utilização da ferramenta.

Nesta etapa, definimos um exercício de programação em *sockets* na linguagem *Python* e submetes para um grupo de alunos e egressos do Campus da UFC Quixadá para resolução. Os resultados e uma discussão estão no Capítulo 6.

5 CONFIGURAÇÃO DA FERRAMENTA

Neste Capítulo, apresentamos o passo a passo que cada usuário necessita realizar para poder executar a ferramenta.

5.1 Preparando o Ambiente para Execução

Precisamos instalar e configurar os pacotes e serviços necessários para que a ferramenta execute corretamente. Vamos instalar e configurar a *AWS Command Line Interface* (AWS CLI) e a *Azure command-line interface* (Azure CLI). O *script* tem como objetivo criar um *cluster* na plataforma de nuvem pública AWS ou Azure, para essa finalidade vamos precisar instalar e configurar a AWS CLI e Azure CLI.

A *AWS Command Line Interface* (AWS CLI) é uma ferramenta de código aberto que permite interagir com os serviços da AWS usando comandos no *shell* da linha de comando. Toda a administração, gerenciamento e acesso do IaaS no AWS estão disponíveis na API e CLI do Console de gerenciamento da AWS. Assim, o *script* tem como objetivo criar um *template* para criar uma *stack* no *Cloudformation* que é um serviço que ajuda você a modelar e configurar seus recursos da AWS para gastar menos tempo gerenciando esses recursos, dessa forma podemos definir a infraestrutura como código na nuvem AWS, através de um *template* de *Cloudformation* pode ser definido em um arquivo *yaml* ou *json*, após os dados serem definidos na execução do *script* será criado um *template* para implantar o *cluster* através do AWS CLI.

A *Azure command-line interface* (Azure CLI) é um conjunto de comandos usado para criar e gerenciar recursos do Azure. A Azure CLI está disponível nos serviços do Azure e foi projetada para ajudá-lo a trabalhar rapidamente com o Azure, com ênfase na automação. Dessa forma o *script* irá criar um modelo de implantação para ser implantado através do *Resource Manager* que é o serviço de implantação e gerenciamento do Azure. Ele fornece uma camada de gerenciamento que lhe permite criar, atualizar e excluir recursos em sua conta do Azure. O modelo é um arquivo *json* que define a infraestrutura e a configuração do *cluster* a ser implantado através da Azure CLI.

5.1.1 Linha de Comando da AWS

A *AWS Command Line Interface* (AWS CLI) é uma ferramenta unificada para o gerenciamento de seus serviços da AWS. Na instalação da AWS CLI é necessário ter *Python 2*,

versão 2.7 ou posterior, ou *Python* 3, versão 3.4 ou posterior instalado, é possível instalar a *AWS Command Line Interface* (AWS CLI)¹ versão 1 e suas dependências na maioria das distribuições do Linux usando o gerenciador de pacotes *pip* ou o instalador empacotado. Algumas distribuições *linux* tem o pacote *awscli* disponível em repositórios para outros gerenciadores de pacotes, como *apt* e *yum*.

5.1.2 Linha de Comando da Azure

A *Azure command-line interface* é um conjunto de comandos usado para criar e gerenciar recursos do Azure. A Azure CLI está disponível nos serviços do Azure e foi projetada para ajudá-lo a trabalhar rapidamente com o Azure, com ênfase na automação. Existem duas opções para instalar a Azure CLI² em seu sistema. Primeiro, você pode executar um único comando que baixará um *script* de instalação e executará os comandos de instalação para você. Ou se preferir, você mesmo pode executar os comandos de instalação em um processo passo a passo.

5.2 Opções de Execução do Script

O *script* tem como objetivo construir um *cluster* em uma das plataformas de nuvens públicas AWS ou Azure. Permitindo aos desenvolvedores implantar aplicações sem gerenciar máquinas virtuais, armazenamento ou rede. Facilitando o gerenciamento de infraestrutura do *cluster*, descreve exatamente quais recursos são provisionados e suas configurações. A ferramenta conta com três *scripts* para realizar a criação do *cluster*.

5.2.1 Scripts

O *script cluster.sh* é responsável por determinar em qual plataforma o usuário irá realizar a implantação do *cluster* e fornecer as informações necessárias para para construir a infraestrutura do *cluster* determinando a quantidade de nós e o tipo de máquina virtual que será utilizado.

O *script deployAWS.sh* é responsável por criar um modelo de arquivo do tipo json para realizar a implantação do *cluster* no *CloudFormation*, esse arquivo contém todas as informações necessárias para criação dos recursos na AWS.

¹ <https://github.com/AnttoniC/cloud-deployment-tool/tree/main/Cluster/Aws-CLI>

² <https://github.com/AnttoniC/cloud-deployment-tool/tree/main/Cluster/Azure-CLI>

O *script* `deployAzure.sh` é responsável por criar um modelo de arquivo do tipo *json* para realizar a implantação do *cluster* no *Azure Resource Manager*, esse arquivo contém todas as informações necessárias para criação dos recursos na Azure.

5.2.2 Funções dos Scripts

Ao executar o comando `./cluster.sh -h` irá retornar todas as funcionalidades da ferramenta.

```
~$ ./cluster.sh -h
```

Executando o comando `./cluster.sh` sem opções ele retornará um usage.

```
~$ ./cluster.sh
```

Para realizar a criação de um *cluster* através do *script* é necessário especificar três parâmetros importantes. O primeiro parâmetro é `-c`, sendo os parâmetros aceitos `_aws` para AWS e `_azure` para Azure. O segundo parâmetro é `-n`, para escolher a quantidade de nós que desejamos executar no cluster. O terceiro parâmetro é `-i`, Para escolher o tipo de máquina virtual ou Instância do seu *cluster*, dependendo de qual nuvem você escolher as opções mudam, sendo as opções aceitas para AWS (`t2.micro`, `t2.small` e `t2.medium`) e para Azure (`Standard_B1s` e `Standard_B1ms`).

Como exemplo da utilização do *script*, considere a criação de um *cluster* com 2 nós. Para implantar um *cluster* de 2 nós na AWS o comando é o seguinte:

```
~$ ./cluster.sh -c _aws -n 2 -i t2.micro
```

Para implantar um *cluster* de 2 nós na Azure execute o seguinte comando:

```
~$ ./cluster.sh -c _azure -n 2 -i Standard_B1s
```

6 AVALIAÇÃO DA FERRAMENTA

6.1 Descrição da avaliação

Dado o objetivo principal deste trabalho, é importante fazer a avaliação para obter melhor resultado na validação. Para avaliar a usabilidade da ferramenta desenvolvida começamos propondo um questionário estruturado com 13 perguntas objetivas e subjetivas, no qual foram abordadas questões voltadas para as funções e características da utilização da ferramenta. O segundo passo foi selecionar uma amostra de indivíduos para usar a ferramenta desenvolvida e responder ao questionário. Foram escolhidos alunos e egressos da Universidade Federal do Ceará - Campus Quixadá dos cursos de Redes de Computadores e Ciência da Computação, totalizando (10 pessoas) no período 4 dias obtendo um retorno de 10 questionários. Estes alunos e egressos foram escolhidos devido a sua grade curricular, por terem visto disciplinas voltadas para execução de aplicações paralelas e distribuídas.

6.2 Metodologia de avaliação

Para avaliação da ferramenta foi elaborado um questionário abordando as funcionalidades da mesma, baseado em trabalhos anteriores do curso de Rede de Computadores seguindo a premissa de (SILVA, 2017). Esse questionário foi respondido pelos participantes após a realização de um exercício de programação no qual o ambiente de execução teve que ser criado com o auxílio do *script*. A ferramenta foi apresentada aos alunos dos cursos de Redes de computadores e Ciência da computação, mostrando todas as funcionalidades e funções disponíveis nela. Após a criação de um *cluster* virtual através, um exercício de programação foi executado. Com a conclusão do exercício, foi aplicado o questionário para os alunos e egressos, com o objetivo de realizar uma avaliação qualitativa da ferramenta proposta com o intuito de obter resultados a respeito da usabilidade da ferramenta.

As perguntas aplicadas aos alunos e egressos podem ser observadas na Tabela 2. As questões de 1 a 12 do questionário foram perguntados o nível de concordância (concordo plenamente, concordo, discordo e discordo plenamente) e a questão 13 como subjetiva a fim de obter opiniões sobre uso da ferramenta. Um detalhamento maior do exercício está na Seção 6.3.

Tabela 2 – Perguntas Aplicadas

Número	Questão
1	Na sua opinião, é preciso ter experiência em computação em nuvem (AWS e Azure) para executar a ferramenta ?
2	A preparação do ambiente para execução da ferramenta foi fácil ?
3	A instalação e configuração do AWS-CLI e AZURE-CLI foi fácil ?
4	As informações disponíveis para execução da ferramenta são aceitáveis para execução ?
5	Conseguiu realizar a execução do <i>cluster</i> através da ferramenta ?
6	Conseguiu ter acesso à infraestrutura do <i>cluster</i> ?
7	Você conseguirá realizar uma implantação de um <i>cluster</i> sem os <i>scripts</i> ?
8	O uso da ferramenta é viável para implantação de serviços e aplicações ?
9	O tempo que a ferramenta leva para criar a infraestrutura do <i>cluster</i> é aceitável ?
10	A ferramenta é relevante para práticas e construções de ambientes de desenvolvimento em Nuvens ?
11	Foi fácil obter as informações de rede para realizar o exercício ?
12	Foi fácil realizar a configuração e execução dos nós (<i>Computes</i>) do <i>cluster</i> ?
13	O que você achou da ferramenta ?

Fonte – Elaborada pelo autor

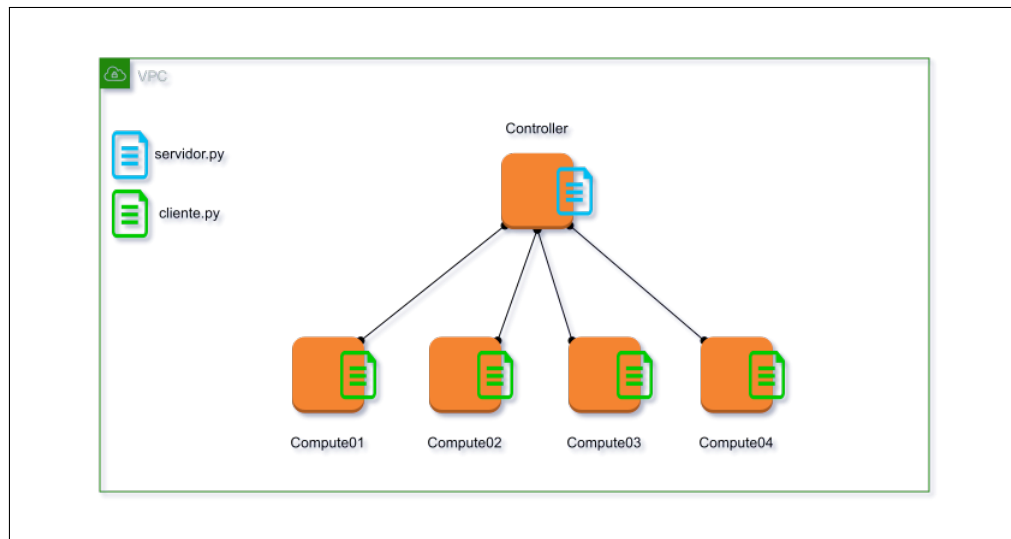
6.3 Descrição do Exercício

A Figura 3 apresenta o esquema geral da arquitetura do exercício. Um soquete de rede é um ponto final de um fluxo de comunicação entre processos através de uma rede de computadores. Neste exercício¹ definimos uma estrutura cliente/servidor para ser implantada no *cluster* criado, no qual o servidor *Controller* fica à espera de conexões dos nós *Computes*. O exercício foi aplicado em dois grupos de cinco participantes que executaram a ferramenta na plataforma Azure e cinco na AWS.

O *cluster* criado nesse exercício vai contém cinco máquinas virtuais sendo uma *Controller* e quatro *Compute*, que são os nós de processamento. O *Controller* executa o programa *servidor.py* e os nós executam o arquivo *cliente.py*. Como pode ser observado pelas extensões dos arquivos, os programas foram desenvolvidos na linguagem *Python*. Objetivo desse exercício é mostrar que o *cluster* está configurado e a infraestrutura está pronta para práticas distribuídas.

¹ <https://github.com/AnttoniC/cloud-deployment-tool/tree/main/Cluster/Socket>

Figura 3 – Infraestrutura do Cluster



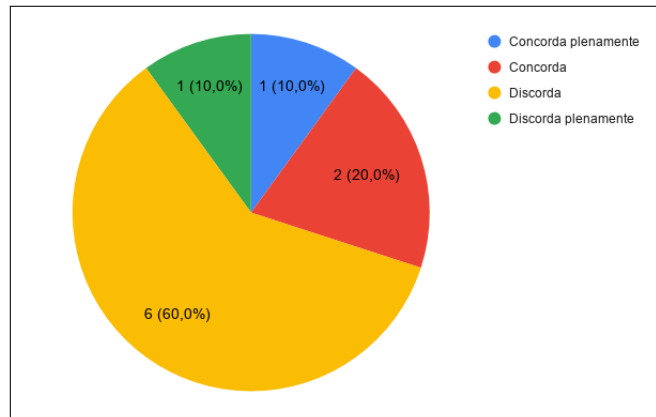
Fonte – Elaborada pelo autor

6.4 Resultados e Discussões

Com os resultados consolidados realizamos o tratamento dos dados obtidos na avaliação, feita por meio do questionário aplicado aos alunos e egressos. A maioria dos participantes foram do sexo masculino e são estudantes do curso de Redes de Computadores, sendo 60% dos envolvidos. Tal fato se deve ao curso conter na sua grade curricular disciplinas voltadas para o desenvolvimento sistemas distribuídos se adequando com o uso da ferramenta proposta. A utilização da ferramenta pelos estudantes foi visível em maior parte pelos alunos do oitavo semestre e sexto semestre, sendo estes, responsáveis por 40%. Mesmo alunos dos primeiros semestres do curso conseguiram executar o experimento sem muita dificuldade.

Seguindo esse contexto, questionamos se seria preciso ter experiência para executar a ferramenta de computação em nuvem (AWS e Azure). Como mostra o gráfico na Figura 4, a maioria dos participantes 60% discorda, 20% concorda e apenas 10% concorda plenamente e discorda plenamente. Isso se dá, pela facilidade de manuseio da ferramenta desenvolvida. No trabalho Silva (2017), estudo similar desenvolveu uma ferramenta para criação de cenários para práticas em laboratórios.

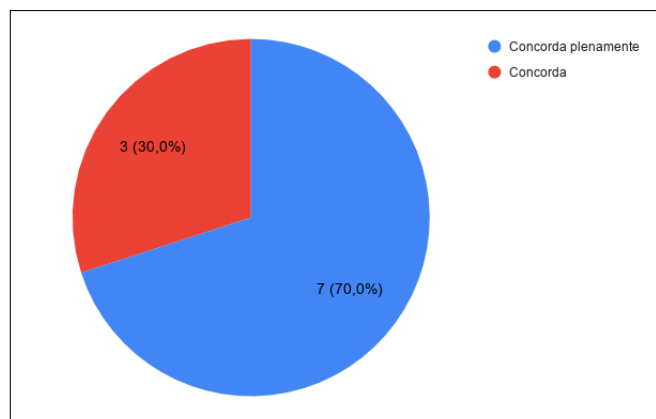
Figura 4 – Questão 1: Experiência para usar a ferramenta



Fonte – Elaborada pelo autor

Considerando a usabilidade indagamos em relação a preparação do ambiente para execução da ferramenta, se a mesma possui uso fácil. Observamos o resultado no gráfico na figura 5.

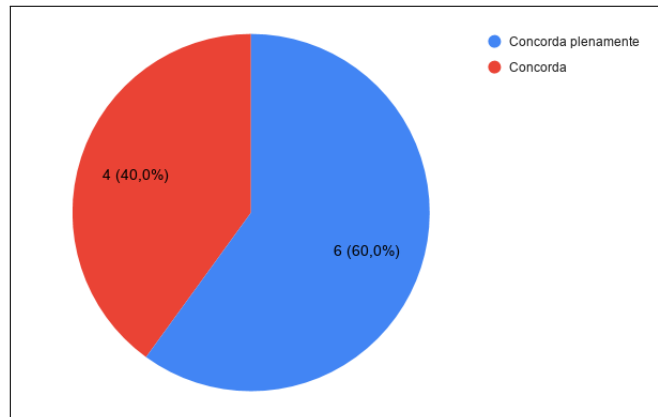
Figura 5 – Questão 2: Facilidade para preparação e execução da ferramenta



Fonte – Elaborada pelo autor

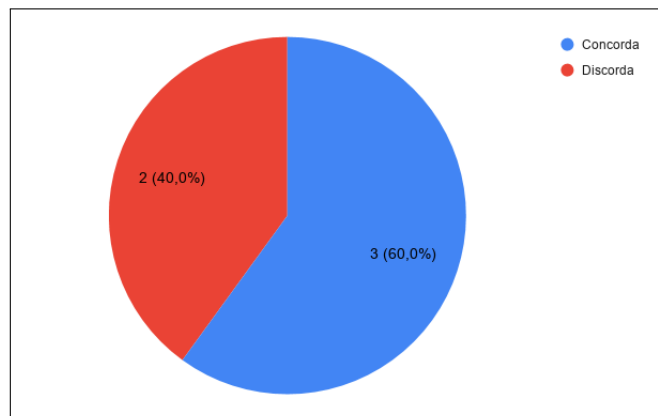
É notável que os alunos avaliaram de modo o acesso às máquinas considerado fácil com 70% concordando plenamente, porém os 30% restantes, considerou que a preparação do ambiente não é tão fácil. Quando indagados sobre as instalações e configurações do AWS-CLI e AZURE-CLI, bem como as informações disponíveis para execução considerando a facilidade, obtivemos uma boa aceitação com a maioria dos participantes concordando plenamente em ambos questionamentos. Veja o gráfico na figura 6 e 7, nos quais podemos observar que mais da metade dos participantes consideram a configuração viável.

Figura 6 – Questão 3: Instalações e configurações do AWS-CLI e AZURE-CLI



Fonte – Elaborada pelo autor

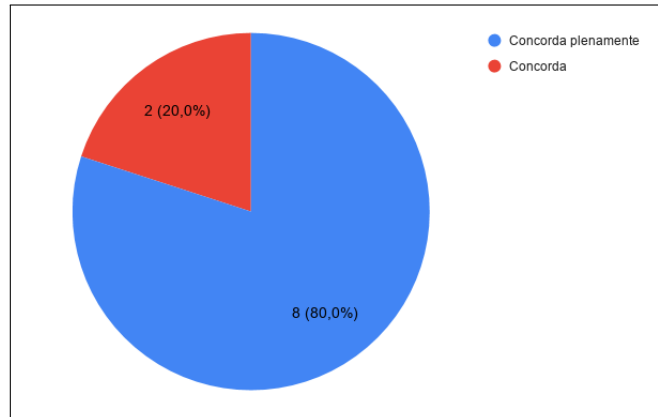
Figura 7 – Questão 4: Informações disponíveis para execução



Fonte – Elaborada pelo autor

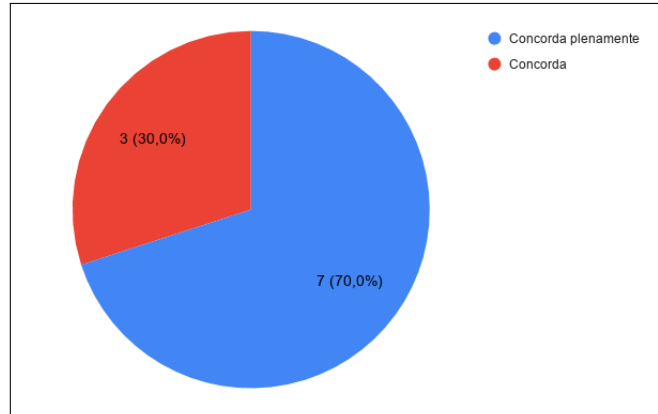
A pergunta 5 teve como foco a realização da a execução do *cluster* através da ferramenta, assim 80% concordou plenamente, 20% concorda e nenhum dos participantes discorda. Isso mostra que a execução é simples e não é necessário muito conhecimento na área para criar um *cluster*. Na pergunta 6 questionamos se os mesmo tiveram acesso a infraestrutura do *cluster*, assim 70% concordou plenamente, 30% concorda e nenhum dos participantes discorda. Mostrando que a infraestrutura do *cluster* é de fácil acesso para o usuário. Veja os gráficos na figura 8 e 9.

Figura 8 – Questão 5: Execução do *cluster* através da ferramenta



Fonte – Elaborada pelo autor

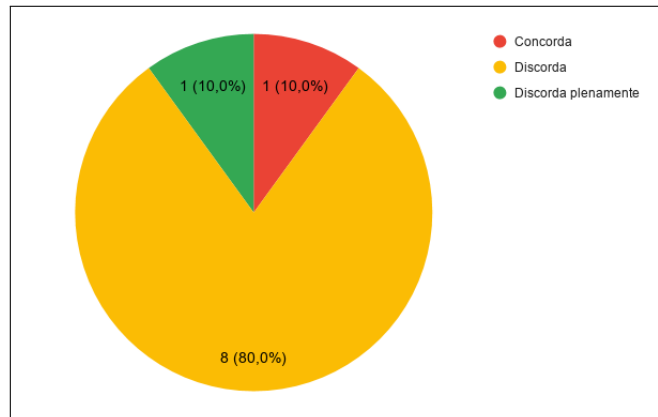
Figura 9 – Questão 6: Acesso à infraestrutura do *cluster*



Fonte – Elaborada pelo autor

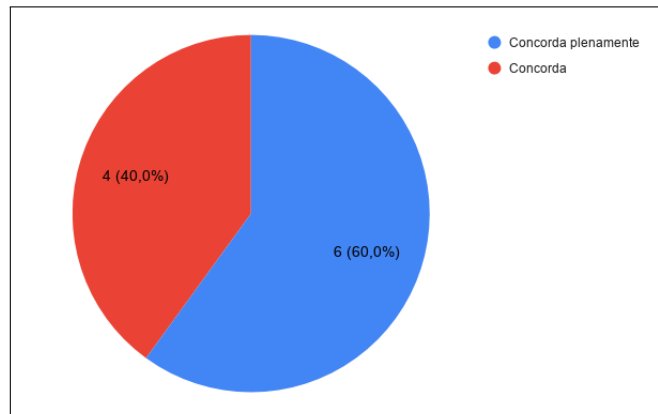
Na pergunta 7 abordamos se os participantes conseguiriam realizar a criação da infraestrutura do cluster sem o auxílio dos *scripts*, assim 80% discordou, 10% concordou e 10% dos participantes discordou plenamente. Isso mostra que para construir a infraestrutura *cluster* manualmente os participantes precisam ter um conhecimento mais aprofundado nas plataformas. Na pergunta 8 foi abordado se o uso da ferramenta é viável para implantações e serviços, assim 60% concordou plenamente, 40% concorda e nenhum dos participantes discorda. Veja os gráficos na figura 10 e 11.

Figura 10 – Questão 7: Implantação do *cluster* sem os *scripts*



Fonte – Elaborada pelo autor

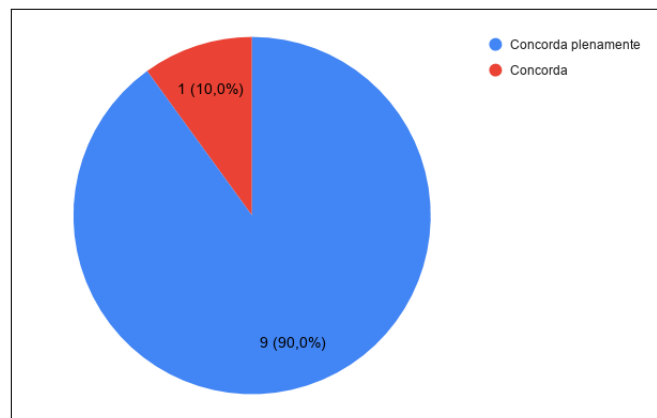
Figura 11 – Questão 8: Viabilidade para implantação de serviços e aplicações



Fonte – Elaborada pelo autor

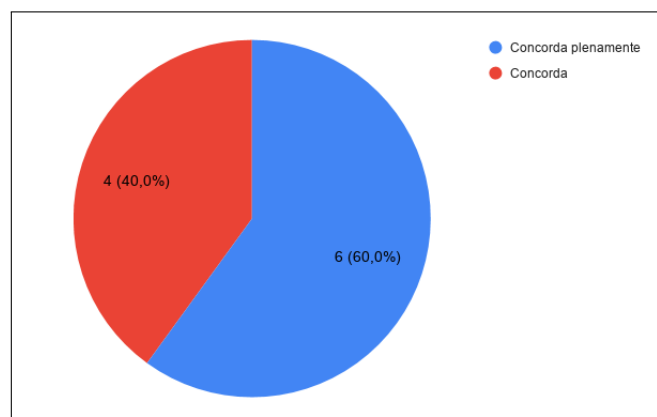
Nas perguntas das questões 9 a 10 tivemos como objetivo qualificar o uso e o desempenho da ferramenta, na pergunta 9 abordamos se o tempo para criação da infraestrutura do *cluster* era satisfatório, 90% concordou plenamente, 10% concordou e nenhum dos participantes discorda. Na pergunta 10 questionamos se a ferramenta é relevante para práticas e construções de ambientes de desenvolvimento em Nuvens, 60% concordou plenamente, 40% concordou e nenhum dos participantes discorda. Com tais resultados obtidos podemos concluir que os participantes estão satisfeitos com a usabilidade da ferramenta, considerando as proporções registradas nas Figuras 12 e 13.

Figura 12 – Questão 9: Tempo para criar a infraestrutura do *cluster*



Fonte – Elaborada pelo autor

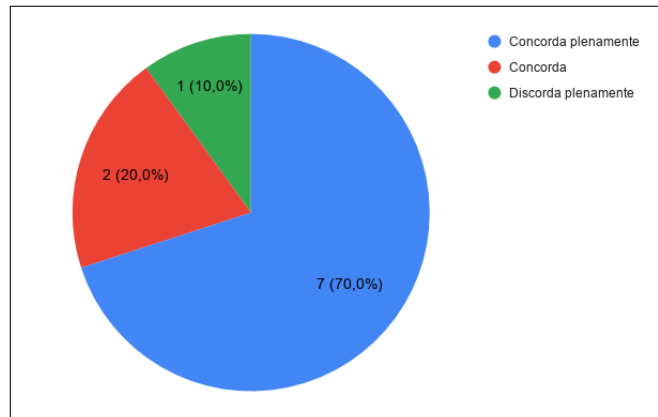
Figura 13 – Questão 10: Relevância para práticas em nuvens



Fonte – Elaborada pelo autor

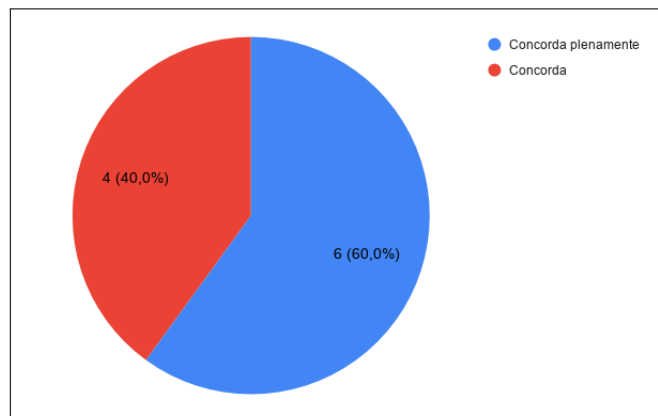
Nas perguntas das questões 11 a 12, o objetivo é mostrar que o *cluster* está configurado e os participantes conseguiriam obter acesso aos servidores do *cluster* para poder realizar o exercício. Na pergunta 11 questionamos se os participantes obtiveram as informações necessárias para realizar o exercício, 70% concordou plenamente, 20% concordou e 10% discordou plenamente. Na pergunta 12 perguntamos se os participantes conseguiram realizar a configuração dos nós do *cluster*, 60% concordou plenamente, 40% concordou e nenhum dos participantes discorda. Com tais informações obtidas podemos concluir que o gerenciamento e acesso aos nós do *cluster* não é complexo, sendo sua utilização viável no contexto das disciplinas do Campus.

Figura 14 – Questão 11: Obter as informações de rede para realizar o exercício



Fonte – Elaborada pelo autor

Figura 15 – Questão 12: Realização da configuração e execução dos nós



Fonte – Elaborada pelo autor

Na pergunta 13 abrimos espaço para a opinião subjetiva dos participantes sobre a ferramenta, os mesmos relataram que a ferramenta tem um potencial enorme para a criação de *clusters* de forma fácil e rápida. Houve afirmações que utilizar ferramentas que facilitem a configuração e uso de serviços é algo muito relevante, alguns pontos citados foram a facilidade e manipulação do *cluster* criado para o uso de grandes infraestruturas em nuvens, mesmo sem conhecimento avançados em ambientes AWS e AZURE podem realizar a criação de *cluster*.

7 CONCLUSÃO

Concluimos com este trabalho, que a ferramenta desenvolvida é capaz de criar *clusters* virtuais nas plataformas de nuvens públicas abordada através do modelo de serviço IaaS, no qual o usuário faz uso de infraestruturas virtuais para práticas de programação paralelas e distribuídas. Iniciamos com um levantamento dos provedores de nuvem públicas que apresentavam algum nível gratuito, para em seguida desenvolver a ferramenta em *shell script*, facilitando o uso da *interface* por linha de comando para acessar os recursos do provedor de nuvem.

Diante dos resultados obtidos através da prática realizada com alunos da Universidade Federal do Ceará, tivemos como objetivo uma análise quantitativa, a fim de mostrar que o *cluster* criado está apto para práticas paralelas e distribuídas, sendo avaliada pelos os participantes, os mesmo retrataram que a ferramenta tem um potencial enorme para a criação de *clusters* de forma fácil e rápida, notamos que a ferramenta atingiu o objetivo geral deste trabalho.

Independentemente dos resultados positivos obtidos no exercício, identificamos que a ferramenta ainda possui limitações, tais como uma opção para deletar os recursos do *cluster* caso a execução do *cluster* não tenha sido concluída completamente, uma vez que a ferramenta não tem uma avaliação de *logs*, em caso de erro de execução dos *scripts* na hora de executar a implantação do *cluster*.

Levando em consideração os resultados, como trabalhos futuros pretendemos melhorar as funcionalidades da ferramenta, como a mesma tem acesso aos recursos das nuvens públicas por linha de comando, tendo potencial para desenvolver novas funcionalidades com base nos recursos disponibilizados por cada provedor. Outra sugestão seria aumentar o número de provedores de nuvens públicas a ferramenta.

REFERÊNCIAS

- AMAZON. **AWS CloudFormation**. [S.I.]. 2021. Disponível em: https://docs.aws.amazon.com/pt_br/AWSCloudFormation/. [Acesso em : 20 fev. 2021].
- ARMBRUST, M.; FOX, A.; GRIFFITH, R.; JOSEPH, A. D.; KATZ, R.; KONWINSKI, A.; LEE, G.; PATTERSON, D.; RABKIN, A.; STOICA, I. et al. A view of cloud computing. **Communications of the ACM**, ACM New York, NY, USA, v. 53, n. 4, p. 50–58, 2010.
- BACELLAR, H. V. **Cluster: Computação de alto desempenho**. Instituto de Computação, Universidade Estadual de Campinas. Campinas, São Paulo, 2009.
- BHARDWAJ, S.; JAIN, L.; JAIN, S. Cloud computing: A study of infrastructure as a service (iaas). **International Journal of engineering and information Technology**, Mandi Gobindgarh, Punjab, Índia, v. 2, n. 1, p. 60–63, 2010.
- CABALLER, M.; ALFONSO, C. D.; ALVARRUIZ, F.; MOLTÓ, G. Ec3: Elastic cloud computing cluster. **Journal of Computer and System Sciences**, Elsevier, [S.I.], v. 79, n. 8, p. 1341–1351, 2013.
- CABALLER, M.; ALFONSO, C. D.; MOLTÓ, G.; ROMERO, E.; BLANQUER, I.; GARCÍA, A. Codecloud: A platform to enable execution of programming models on the clouds. **Journal of Systems and Software**, Elsevier, [S.I.], v. 93, p. 187–198, 2014.
- CHOPRA, N.; SINGH, S. Deadline and cost based workflow scheduling in hybrid cloud. In: IEEE. **Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on**. [S.I.], 2013. p. 840–846.
- DILLON, T.; WU, C.; CHANG, E. Cloud computing: issues and challenges. In: IEEE. **Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on**. [S.I.], 2010. p. 27–33.
- GOYAL, S. Public vs private vs hybrid vs community-cloud computing: a critical review. **International Journal of Computer Network and Information Security**, Modern Education and Computer Science Press, [S.I.], v. 6, n. 3, p. 20, 2014.
- JADEJA, Y.; MODI, K. Cloud computing-concepts, architecture and challenges. In: IEEE. **Computing, Electronics and Electrical Technologies (ICCEET), 2012 International Conference on**. [S.I.], 2012. p. 877–880.
- MICROSOFT. **Documentação do Azure Resource Manager**. [S.I.]. 2021. Disponível em: <https://docs.microsoft.com/pt-br/azure/azure-resource-manager/>. [Acesso em: 20 fev. 2021].
- RIGHI, R. da R.; RODRIGUES, V. F.; COSTA, C. A. D.; GALANTE, G.; BONA, L. C. E. D.; FERRETO, T. Autoelastic: Automatic resource elasticity for high performance applications in the cloud. **IEEE Transactions on Cloud Computing**, IEEE, [S.I.], v. 4, n. 1, p. 6–19, 2015.
- RUSCHEL, H.; ZANOTTO, M. S.; MOTA, W. d. C. **Computação em nuvem**. Pontifícia Universidade Católica do Paraná, Curitiba, Brazil, 2010.
- SADASHIV, N.; KUMAR, S. D. Cluster, grid and cloud computing: A detailed comparison. In: CITESEER. **Computer Science & Education (ICCSE), 2011 6th International Conference on**. [S.I.], 2011. p. 477–482.

SILVA, W. F. Utilizando virtualização baseada em containers para criação de laboratórios práticos de disciplinas na área de TI. [S.I.:s.n], 2017.

SOUSA, F. R.; MOREIRA, L. O.; MACHADO, J. C. Computação em nuvem: Conceitos, tecnologias, aplicações e desafios. II Escola Regional de Computação Ceará, Maranhão e Piauí (ERCEMAPI), p. 150–175, 2009.