



UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO  
DEPARTAMENTO DE ESTATÍSTICA E INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA APLICADA

CLAUDEMIR JEREMIAS DE LIMA

# AVALIAÇÃO DE DESEMPENHO E CONSUMO DE ENERGIA DO AMBIENTE MOODLE

RECIFE – PE

2020

**CLAUDEMIR JEREMIAS DE LIMA**

**AVALIAÇÃO DE DESEMPENHO E CONSUMO DE  
ENERGIA DO AMBIENTE MOODLE**

Dissertação submetida à Coordenação do Programa de Pós-Graduação em Informática Aplicada da Universidade Federal Rural de Pernambuco, como parte dos requisitos necessários para obtenção do grau de Mestre.

**ORIENTADOR: Prof. Dr. Gustavo Rau de Almeida Callou**

**RECIFE – PE**

**2020**

Dados Internacionais de Catalogação na Publicação  
Universidade Federal Rural de Pernambuco  
Sistema Integrado de Bibliotecas  
Gerada automaticamente, mediante os dados fornecidos pelo(a) autor(a)

---

- L732a Lima, Claudemir Jeremias de  
Avaliação de desempenho e consumo de energia do ambiente MOODLE / Claudemir Jeremias de Lima.  
- 2020.  
98 f. : il.
- Orientador: Gustavo Rau de Almeida Callou.  
Inclui referências.
- Dissertação (Mestrado) - Universidade Federal Rural de Pernambuco, Programa de Pós-Graduação em  
Informática Aplicada, Recife, 2021.
1. Computação em nuvem. 2. Avaliação de desempenho. 3. Moodle. 4. Consumo de energia. 5. Apache  
CloudStack. I. Callou, Gustavo Rau de Almeida, orient. II. Título

CLAUDEMIR JEREMIAS DE LIMA

# AVALIAÇÃO DE DESEMPENHO E CONSUMO DE ENERGIA DO AMBIENTE MOODLE

Dissertação submetida à Coordenação do  
Programa de Pós-Graduação em Informática  
Aplicada da Universidade Federal Rural  
de Pernambuco, como parte dos requisitos  
necessários para obtenção do grau de Mestre.

Aprovada em:

BANCA EXAMINADORA

---

Prof. Dr. Gustavo Rau de Almeida Callou (Orientador)  
Universidade Federal Rural de Pernambuco  
Departamento de Computação

---

Prof. Dr. Ermeson Carneiro de Andrade  
Universidade Federal Rural de Pernambuco  
Departamento de Computação

---

Profa. Dra. Juliana Regueira Basto Diniz  
Universidade Federal Rural de Pernambuco  
Unidade Acadêmica de Educação a Distância e Tecnologia



Dedico esta dissertação a minha família e amigos, em especial ao meu filho, Davi Jeremias Oliveira de Lima (in memoriam), que é o meu filho amado e desejado.

# Agradecimentos

Agradeço primeiramente a Deus, Jeová Samá, por ter me dado forças para prosseguir, mesmo diante das dificuldades encontradas no caminho e por chegar à conclusão deste trabalho e não desistir diante de tudo isso.

Agradeço ao Prof. Gustavo Callou pela orientação, comentários que foram fundamentais para o desenvolvimento deste trabalho.

Agradeço também à minha esposa Elidiane Gomes e à minha filha Letícia Vitória, que sempre estão comigo em todos os momentos. Ao meu filho Davi Jeremias Oliveira de Lima, que motivou a busca por qualificação profissional. Sereno em alguns momentos e extremamente alegre em tantos outros, Deus permitiu que, em uma consulta de emergência no hospital (HJM), devido a uma crise asmática, ele viesse a falecer. Muito obrigado, meu filho, por proporcionar nossa felicidade por 17 anos.

Também meu muito obrigado a todos os colegas e amigos do grupo de pesquisa, em especial a Alison Vinícius, que fez contribuições nesta pesquisa, também a Luiz Sérgio e Osvaldo pelas dicas e por estarem dispostos a ajudar.

Aos professores das disciplinas do PPGIA, em especial: Ermeson Carneiro, Fernando Aires, Gilberto Cysneiros, Gustavo Callou, Jorge da Silva e Tiago Alessandro; e também a todos os funcionários e colaboradores que sempre me atenderam muito bem, particularmente Eduardo Chaves, nosso secretário. Agradeço ao CNPq e à FACEPE, pelo apoio para a realização desta pesquisa. À UFRPE, pelo suporte com a infraestrutura necessária ao desenvolvimento deste trabalho.

Meu eterno agradecimento!

## Resumo

Devido às vantagens como alocação dinâmica de recursos e redução de custos, a computação em nuvem aparece cada vez mais em várias áreas da sociedade: educação, negócios e finanças. Com o aumento da utilização de *softwares* de ensino e aprendizagem, Ambiente Virtual de Aprendizagem (AVA), surgiu uma necessidade de atender várias requisições ao mesmo tempo, sem reduzir o desempenho dos serviços que requerem recursos dinâmicos, com elevado desempenho. Nesse contexto, o paradigma da computação em nuvem representa uma alternativa interessante por poder fornecer esses recursos sob demanda e com alta escalabilidade. O desempenho da computação em nuvem está relacionado diretamente à infraestrutura de Tecnologia da Informação, que é composta por dispositivos de *hardware* e *software*, além das tecnologias para realizar a gestão dos dados e, também, para permitir a comunicação entre os componentes. Sendo assim, existe uma demanda para se avaliar, de forma integrada, o desempenho e o consumo de energia dessas infraestruturas, a fim de minimizar, inclusive, o impacto negativo na sustentabilidade e no custo. Dessa forma, este trabalho apresenta uma estratégia de modelagem computacional, para comparar o consumo de energia e o desempenho de sistema em nuvem privada, utilizando um ambiente virtual de aprendizagem com as métricas de desempenho, como: vazão, tempo de resposta, utilização da CPU, utilização da memória RAM, do serviço hospedado nos dois sistemas de nuvens privadas, OpenStack e Apache CloudStack. Essas métricas são avaliadas a partir da entrada dos parâmetros de configuração e taxa de requisições esperada para o sistema. Para atingir este objetivo, o modelo proposto utiliza as redes de Petri estocásticas (SPN) para quantificar e avaliar o desempenho de tais sistemas de computação em nuvem. Em laboratório, um ambiente de teste foi configurado de modo a se poder avaliar esses dois sistemas de nuvens privadas. Estudos de caso são apresentados para análises da medição e simulação dos dois sistemas em nuvem. Os estudos de caso demonstraram que, no comparativo, os dois sistemas de nuvem possuem comportamentos distintos, com um consumo de energia diferente, utilização da memória RAM e utilização da CPU, vazão e tempo de resposta. A utilização da estratégia de modelagem permite ainda a realização das simulações de cenários difíceis de serem realizados via medições e, assim, podem prover estimativas aos projetistas desses ambientes antes mesmo de terem implementado tais sistemas.

**Palavras-chave:** Avaliação de Desempenho, Moodle, Computação em Nuvem, OpenStack, Apache CloudStack, Consumo de Energia

# Abstract

Due to advantages such as dynamic resource allocation and cost reduction, cloud computing is increasingly appearing in several areas of society: education, business and finance. With the increased use of teaching and learning software, Virtual Learning Environment (VLA), a need has arisen to meet multiple requests at the same time, without reducing the performance of services that require dynamic, high-performance resources. In this context, the cloud computing paradigm represents an interesting alternative because it can provide these resources on demand and with high scalability. The performance of cloud computing is directly related to the Information Technology infrastructure, which is composed of hardware and software devices, as well as technologies to perform data management and also to allow communication between components. Therefore, there is a demand to evaluate, in an integrated way, the performance and energy consumption of these infrastructures, in order to minimize, even, the negative impact on sustainability and cost. Thus, this work presents a strategy of computational modeling, to compare the power consumption and performance of a private cloud system, using a virtual learning environment with performance metrics such as: flow, response time, CPU utilization, RAM utilization, the service hosted in the two private cloud systems, OpenStack and Apache CloudStack. These metrics are evaluated from the input of configuration parameters and the expected request rate for the system. To achieve this goal, the proposed model uses stochastic Petri networks (SPN) to quantify and evaluate the performance of such cloud computing systems. In the laboratory, a test environment has been configured so that these two private cloud systems can be evaluated. Case studies are presented for measurement and simulation analysis of both cloud systems. The case studies demonstrated that, in comparison, the two cloud systems have different behaviors, with different power consumption, RAM utilization and CPU utilization, flow rate and response time. The use of the modeling strategy also allows the simulation of difficult scenarios to be performed via measurements and, thus, can provide estimates to the designers of these environments even before they have implemented such systems.

Translated with [www.DeepL.com/Translator](http://www.DeepL.com/Translator) (free version)

**Keywords:** Performance Evaluation, Moodle, Cloud Computing, OpenStack, Apache CloudStack, Energy consumption

## Lista de Figuras

Figura 1 – Infraestrutura em nuvem do Apache CloudStack . . . . .	30
Figura 2 – infraestrutura em nuvem do OpenStack . . . . .	32
Figura 3 – Classificação das técnicas de análises de desempenho (CALLOU, 2009)	34
Figura 4 – Elementos da Redes de Petri . . . . .	35
Figura 5 – Elemento da SPN . . . . .	36
Figura 6 – Exemplo de uma Rede de Petri (RdP) . . . . .	37
Figura 7 – Metodologia adotada. . . . .	41
Figura 8 – Tela inicial do Apache JMeter . . . . .	44
Figura 9 – Watts Up Pro meter - Coletor de dados do consumo de energia . . . . .	44
Figura 10 – Estrutura utilizada no laboratório de pesquisa do PPGIA . . . . .	45
Figura 11 – Infraestrutura montada para os experimentos. . . . .	46
Figura 12 – Modelo de desempenho proposto . . . . .	47
Figura 13 – Gráfico da Regressão linear - Consumo Energia Apache CloudStack . . . . .	50
Figura 14 – Gráfico da Regressão linear - Consumo Energia OpenStack . . . . .	51
Figura 15 – Regressão linear - Tempo de resposta do Apache CloudStack . . . . .	52
Figura 16 – Regressão linear - Tempo de resposta do OpenStack . . . . .	53
Figura 17 – Regressão linear - Tempo de processamento Apache CloudStack . . . . .	53
Figura 18 – Regressão linear - Tempo de processamento OpenStack . . . . .	54
Figura 19 – Regressão linear - Tempo de latência Apache CloudStack . . . . .	55
Figura 20 – Regressão linear - Tempo de latência OpenStack . . . . .	55
Figura 21 – Consumo de energia (Watts) . . . . .	61
Figura 22 – Utilização da CPU (%) . . . . .	62
Figura 23 – Comparativo da utilização da memória RAM. . . . .	64
Figura 24 – Comparação da vazão . . . . .	65
Figura 25 – Comparação do Tempo de resposta . . . . .	66
Figura 26 – Comparativo da medição e modelagem - nível de utilização da CPU. . . . .	69
Figura 27 – Resultado da medição e modelagem RAM - Apache CloudStack e OpenStack . . . . .	70
Figura 28 – Resultado da medição e modelagem Watts - Apache CloudStack e OpenStack . . . . .	71

Figura 29 – Resultado da simulação da vazão - Apache CloudStack e OpenStack .	73
Figura 30 – Resultado da simulação do tempo de resposta - Apache CloudStack e OpenStack . . . . .	74
Figura 31 – Resultado da simulação da utilização da CPU - Apache CloudStack e OpenStack . . . . .	79
Figura 32 – Resultado da simulação da utilização da Memória RAM - Apache CloudStack e OpenStack . . . . .	79
Figura 33 – Resultado da Utilização da CPU com variação do T1 - Apache CloudStack . . . . .	81
Figura 34 – Resultado da Utilização da RAM com variação do T1 - Apache CloudStack . . . . .	83
Figura 35 – Resultado da Utilização da CPU com variação do T1 - OpenStack . .	84
Figura 36 – Resultado da Utilização da RAM com variação do T1 - OpenStack . .	84
Figura 37 – Resultado da simulação da utilização da CPU do Apache CloudStack e OpenStack . . . . .	85
Figura 38 – Resultado da simulação da utilização da Memória RAM do Apache CloudStack e OpenStack . . . . .	86
Figura 39 – Resultado da simulação do consumo de energia(Watts) - OpenStack vs Apache CloudStack . . . . .	87
Figura 40 – Resultado da medição e simulação do consumo de energia(Watts) - OpenStack vs Apache CloudStack . . . . .	88

## Lista de tabelas

Tabela 1 – Características dos Trabalhos Relacionados. . . . .	26
Tabela 2 – Descrição dos lugares do modelo. . . . .	49
Tabela 3 – Descrição das transições do modelo. . . . .	49
Tabela 4 – Teste-t: duas amostras em par para médias (W). . . . .	61
Tabela 5 – Comparação do consumo de energia - OpenStack vs Apache CloudStack. . . . .	62
Tabela 6 – Teste-t: duas amostras em par para médias (CPU). . . . .	63
Tabela 7 – Comparação do nível de utilização da CPU - OpenStack vs Apache CloudStack. . . . .	63
Tabela 8 – Comparação da utilização da memória RAM - OpenStack vs Apache CloudStack. . . . .	64
Tabela 9 – Teste-t: duas amostras em par para médias Memória RAM da VW. . . . .	65
Tabela 10 – Comparação vazão - Apache CloudStack vs OpenStack. . . . .	66
Tabela 11 – Comparação do tempo de resposta Apache CloudStack vs OpenStack. . . . .	67
Tabela 12 – Parâmetros de entrada do modelo. . . . .	68
Tabela 13 – Resultados para validação do modelo com o nível de utilização da CPU da VM com Apache CloudStack . . . . .	69
Tabela 14 – Resultado para validação do modelo com uso da CPU da VM - OpenStack . . . . .	70
Tabela 15 – Resultado para validação do modelo com utilização da memória RAM - Apache CloudStack . . . . .	70
Tabela 16 – Resultado para validação do modelo com a utilização da memória RAM - OpenStack . . . . .	71
Tabela 17 – Resultado para validação do modelo com o consumo de energia (W) - Apache CloudStack . . . . .	72
Tabela 18 – Resultado para validação do modelo com o consumo em de energia (W) - OpenStack . . . . .	72
Tabela 19 – Parâmetros de variação da CPU (QC) e RAM (QM). . . . .	74
Tabela 20 – Resultados da Simulação utilizado QC=2 - Apache CloudStack. . . . .	75
Tabela 21 – Resultados da Simulação utilizado QC=4 - Apache CloudStack. . . . .	76
Tabela 22 – Resultados da Simulação utilizado QC=6 - Apache CloudStack. . . . .	76



Tabela 23 – Resultados da Simulação utilizado QC=8 - Apache CloudStack. . . . .	76
Tabela 24 – Resultados da Simulação utilizado QC=2 - OpenStack. . . . .	77
Tabela 25 – Resultados da Simulação utilizado QC=4 - OpenStack. . . . .	77
Tabela 26 – Resultados da Simulação utilizado QC=6 - OpenStack. . . . .	78
Tabela 27 – Resultados da Simulação utilizado QC=8 - OpenStack. . . . .	78
Tabela 28 – Parâmetros de variação do Tempo de Chegada. . . . .	80
Tabela 29 – Utilização da CPU (%) Apache CloudStack. . . . .	81
Tabela 30 – Utilização da memória RAM (%) Apache CloudStack. . . . .	82
Tabela 31 – Utilização da CPU (%) OpenStack. . . . .	83
Tabela 32 – Utilização da memória RAM (%) OpenStack. . . . .	84
Tabela 33 – Comparação do Consumo de energia OpenStack vs Apache CloudStack.	87
Tabela 34 – Teste-t: duas amostras em par para a medição do consumo de energia (W). . . . .	88

## Lista de abreviaturas e siglas

API	Application Programming Interface
AVA	Ambiente Virtual de Aprendizagem
AWS	Amazon Web Services
CSS	Cascading Style Sheets
CSV	Valores Separados por Vírgula
CPU	Central Process Uni
CTMC	Continuous Time Markov Chain
EaD	Educação a Distância
E/S	Entrada e Saída
FTP	File Transfer Protocol
GRASP	Greed Randomized Search Procedure
GPU	Graphics Processing Unit
HD	Hard Disk
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IaaS	Infrastructure as a Service
ISO	International Organization for Standardization
LCD	Liquid Crystal Display
KVM	Kernel-based Virtual Machine
VLE	Virtual Learning Environment
MB	megabyte

MLB	Método de Lattice Boltzmann
Moodle	Modular Object-Oriented Dynamic Learning Environment
NASA	National Aeronautics and Space Administration
PaaS	Platform as a Service
PPGIA	Programa de Pós-Graduação em Informática
PN	Petri Nets
RAM	Random Access Memory
RBD	Reliability Block Diagram
RdP	Redes de Petri
SaaS	Software as a Service
SED	Sistemas de Eventos Discretos
SPN	Stochastic Petri Nets
SSD	Solid-State Drive
TI	Tecnologia da Informação
VM	Virtual Machine
UFRPE	Universidade Federal Rural de Pernambuco
USB	Universal Serial Bus
W	Watts
YCSB	Yahoo! Cloud Serving Benchmarking

# Sumário

<b>1</b>	<b>Introdução</b>	<b>17</b>
1.1	Delimitação do Tema	19
1.2	Problema de Pesquisa	19
1.3	Objetivos	20
1.3.1	Objetivo Específico	20
1.4	Organização da Dissertação	20
<b>2</b>	<b>Trabalhos Relacionados</b>	<b>22</b>
2.1	Medição	22
2.2	Modelagem	24
2.3	Moodle	24
2.4	Considerações finais	27
<b>3</b>	<b>Fundamentação Teórica</b>	<b>28</b>
3.1	Computação em Nuvem	28
3.1.1	Apache CloudStack	30
3.1.2	OpenStack	31
3.2	Avaliação de Desempenho	32
3.2.1	Apache JMeter	34
3.3	Redes de Petri	35
3.3.1	Redes de Petri Estocástica	36
3.4	Regressão Linear	38
3.5	Moodle	39
3.6	Considerações finais	40
<b>4</b>	<b>Metodologia</b>	<b>41</b>
4.1	A metodologia	41
4.2	Coleta de Dados	43
4.3	Infraestrutura	45
4.4	Considerações finais	46
<b>5</b>	<b>Modelo</b>	<b>47</b>
5.1	Modelo de Desempenho	47
5.2	Regressão Linear	50

5.2.1	Consumo de Energia e Nível de Utilização da CPU . . . . .	50
5.2.2	Regressão do tempo de resposta e quantidade de clientes . . . . .	51
5.2.3	Regressão do tempo de processamento e quantidade de clientes . . . . .	53
5.2.4	Tempo de latência e quantidade de clientes . . . . .	54
5.3	Métricas . . . . .	56
5.3.1	Utilização de CPU . . . . .	56
5.3.2	Utilização da Memória . . . . .	56
5.3.3	Consumo de Energia . . . . .	56
5.3.4	Vazão . . . . .	57
5.3.5	Tempo de Resposta . . . . .	58
5.4	Considerações finais . . . . .	59
<b>6</b>	<b>Estudos de Caso . . . . .</b>	<b>60</b>
6.1	Estudo de Caso I . . . . .	60
6.1.1	Resultados . . . . .	60
6.2	Estudo de Caso II . . . . .	67
6.2.1	Resultados . . . . .	68
6.3	Estudo de Caso III . . . . .	72
6.3.1	Resultados . . . . .	73
6.4	Estudo de Caso IV . . . . .	74
6.4.1	Resultados . . . . .	75
6.5	Estudo de Caso V . . . . .	80
6.5.1	Resultados . . . . .	81
6.6	Estudo de Caso VI . . . . .	86
6.7	Considerações finais . . . . .	89
<b>7</b>	<b>Conclusão . . . . .</b>	<b>90</b>
7.1	Contribuições . . . . .	90
7.1.1	Produção Científica . . . . .	91
7.2	Limitações . . . . .	91
7.3	Trabalhos futuros . . . . .	92
	<b>Referências . . . . .</b>	<b>94</b>

# 1 Introdução

Surgido em 1961, o termo computação em nuvem propôs um modelo de computação que fornece serviços semelhantes à distribuição de energia (MANSUR et al., 2010). A computação em nuvem está cada vez mais presente na sociedade, em diversas áreas (ex.: educacional, comercial e financeira) devido às suas vantagens, como alocação dinâmica de recursos e baixo custo. Existem diversos exemplos de serviços de *e-commerce*, plataforma de *streaming* e ambientes virtuais de aprendizagem que necessitam de recursos dinâmicos e que são facilmente providos pelo paradigma de computação em nuvem.

A computação em nuvem é um exemplo de computação em grade e computação utilitária (KIM et al., 2010) com uso sob demanda, a partir de uma rede que compartilha recursos de computação configuráveis. Exemplo de recursos que podem ser providos por computação em nuvem são: redes, servidores, armazenamento, aplicativos e serviços, que podem ser adquiridos e liberados rapidamente com o mínimo de esforço, gerenciamento e interação com provedores de serviços (NIST, 2014). Suas principais características são: elasticidade, alta disponibilidade e confiabilidade, baixo custo e reduzido consumo de energia (BAUER; ADAMS, 2012).

A computação em nuvem oferece vantagens como: serviços compartilhados, sistemas operacionais e aplicativos de rede, facilitando o gerenciamento de negócios sem se preocupar com detalhes de infraestrutura (MACHIDA et al., 2011). Por estes motivos, a computação em nuvem é uma opção para a instalação do ambiente virtual de aprendizagem (AVA), que foi originalmente utilizado como meio de comunicação entre alunos e professores na Educação a Distância (EAD). Atualmente, os AVAs vêm sendo utilizados em salas de aula e cursos a distância, e contam com alguns recursos úteis para a educação. A vantagem do AVA para os discentes é que se pode estudar a qualquer hora, em qualquer lugar e só é necessária uma conexão de acesso à Internet. Esse tipo de ambiente permite que alunos e professores troquem ideias em fóruns e salas de *chat*, podendo inclusive aumentar a velocidade de aprendizagem.

O AVA é uma plataforma colaborativa que é utilizada no ensino a distância, podendo ser utilizada também no ensino presencial, além de um acompanhamento de grupos de trabalho. No entanto, tal ambiente demanda elevado desempenho, o qual está relacionado diretamente à infraestrutura de Tecnologia da Informação e Comunicação

(TICs), que é composta por dispositivos de *hardware* e *software*, além das tecnologias para realizar a gestão dos dados e, também, para permitir a comunicação entre os componentes. As infraestruturas desses sistemas são muitas vezes distribuídas e envolvem diversos componentes, existindo múltiplos fatores que podem influenciar tanto no desempenho como no consumo de energia de uma determinada aplicação hospedada em um sistema em nuvem.

É importante destacar que ambientes virtuais de aprendizagem e de computação em nuvem devem proporcionar um bom desempenho para seus usuários. Os projetistas desses ambientes devem planejar qual sistema e recursos que serão demandados para atender aos requisitos de desempenho estabelecidos (DANTAS, 2018). Além disso, existe uma demanda para se avaliar de forma integrada o desempenho e o consumo de energia de tais ambientes, a fim de não ter um impacto negativo na sustentabilidade e no custo de tais serviços. A utilização de técnicas com medição, modelagem analítica e simulação pode auxiliar os projetistas, pois por meio dessas avaliações será possível fazer a escolha que melhor atenda aos seus requisitos. Este trabalho utiliza também a regressão linear, que é uma técnica estatística usada para correlacionar a variável independente  $x$  com a variável dependente  $y$ , e que busca relação matemática entre os resultados obtidos e os valores futuros das variáveis. As equações lineares são importantes porque podem aproximar muitas relações da vida real (RODRIGUES et al., 2013).

Há vários AVAs, como um sistema de gestão do ensino e aprendizagem. O Amadeus, Moodle são plataformas de ensino a distância baseadas em *software* livre (FILHO et al., 2014). O Moodle é um *software* desenvolvido para auxiliar os professores a criar cursos *online* ou cursos presenciais, com uma variedade de recursos disponíveis. Com uma comunidade composta por centenas de programadores em todo o mundo, continua a ser desenvolvido por esta comunidade, que também formou um grupo de apoio aos usuários, para adicionar novas funcionalidades sob o conceito de *software* livre GNU.

Com mais de 14.800 mil alunos matriculados na graduação e pós-graduação em 2018 na UFRPE (UFRPE1, 2019), cada vez mais os ambientes virtuais de aprendizagem são usados no dia a dia de professores e alunos. Além de partes da Bahia, a instituição mantém escritórios em todas as regiões de Pernambuco, por meio de unidades acadêmicas, postos de pesquisa e centros de ensino a distância (EAD). Muitas instituições de ensino públicas no país utilizam ambiente virtual de aprendizagem (TORRES, 2017) e necessitam armazenar

dados de maneira elástica, de modo a manter o serviço sem perder o desempenho. Algumas soluções de *software* adotam a virtualização para fornecer serviços de IaaS, Infraestrutura como serviço em nuvem, bem como para melhorar o desempenho do serviço e dos dados, segurança, utilização de recursos, custos e outros benefícios (TORRES et al., 2016b; MACHIDA et al., 2011; TRIVEDI et al., 1996).

Nesse sentido, a hospedagem do ambiente de aprendizagem virtual, Moodle, em sistema de nuvem privada é uma alternativa que poderá possibilitar um maior desempenho, aumentando somente o consumo energético quando necessário. Isso ocorre devido ao fato de que a computação em nuvem fornece elasticidade e, assim, uma infraestrutura flexível para atender às suas cargas de trabalho de serviço em seus domínios de gerenciamento.

Dessa forma, esta pesquisa irá fazer uso de uma estratégia de avaliação de desempenho, que faz uso de modelagem computacional e que visa a comparação do ambiente Moodle hospedado em diferentes nuvens privadas.

## 1.1 Delimitação do Tema

A presente pesquisa trata sobre modelagem e análises de serviço em nuvens computacionais, a partir dos sistemas OpenStack e Apache CloudStack, utilizando a plataforma de aprendizagem *online* Moodle, a qual será o campo de pesquisa a ser explorado. A proposta de análise e uso desse modelo de serviço demandará um desempenho a partir de uma avaliação monitorada pelos sistemas, além de disponibilizar uma análise da utilização dos recursos computacionais, como: memória RAM, CPU.

## 1.2 Problema de Pesquisa

Devido ao aumento da utilização da tecnologia da informação em ambientes *online* para o ensino e aprendizagem, há uma necessidade crescente para o desempenho desses sistemas com a finalidade de atender as demandas das instituições e seus usuários. A falta de uma infraestrutura adequada para suportar o acesso de múltiplos usuários simultaneamente no ambiente virtual de aprendizagem AVA pode ocasionar um baixo desempenho. Dessa forma, a presente pesquisa tem como questão principal: qual é a nuvem privada em *software* de código aberto para suportar um ambiente de aprendizagem virtual com desempenho e



consumo de energia satisfatórios?

### 1.3 Objetivos

Esta pesquisa tem como objetivo a proposição de uma estratégia de avaliação de desempenho e consumo de energia que visa a comparação de um Ambiente Virtual de Aprendizagem hospedado em diferentes nuvens privadas. Dessa forma, nuvens como o OpenStack e Apache CloudStack serão comparadas quanto a métricas de desempenho e de consumo de energia. Para se atingir tal objetivo, a metodologia proposta irá usar as redes de Petri estocásticas (SPN) para quantificar e avaliar o desempenho e consumo de energia de tais sistemas computacionais.

#### 1.3.1 Objetivo Específico

Para se atingir o objetivo geral, foram definidos os seguintes objetivos específicos:

- Montar e avaliar o desempenho do Ambiente Virtual de Aprendizagem nas nuvens Apache CloudStack e OpenStack;
- Propor modelo de desempenho para representar o ambiente a ser avaliado;
- Propor modelo de consumo de energia para as nuvens Apache CloudStack e OpenStack com o sistema Moodle hospedado;
- Realizar experimentos de medição com o Apache JMeter para coletar informações sobre desempenho e consumo energético. Os resultados desses testes irão servir para fazer a validação do modelo proposto em SPN e, também, como parâmetros de entrada.

### 1.4 Organização da Dissertação

O restante desta dissertação está organizada da seguinte forma. O Capítulo 2 apresenta os trabalhos relacionados, destacando artigos da literatura relacionados à medição e modelagem de sistema em computação nas nuvens e de Moodle. O Capítulo 3 introduz os conceitos fundamentais para um melhor entendimento deste trabalho. O capítulo inicia com explicação sobre: computação em nuvem, Apache CloudSack, OpenStack, avaliação

de desempenho, Apache JMeter, redes de Petri, redes de Petri Estocástica, regressão linear, Moodle. O Capítulo 4 mostra a metodologia desta pesquisa, a coleta de dados e infraestrutura. O Capítulo 5 apresenta o modelo SPN proposto na pesquisa. Esse modelo faz uso de métricas e de regressão linear, também explicado neste capítulo. Em seguida, o Capítulo 6 apresenta os estudos de caso utilizados para mostrar a aplicabilidade do modelo proposto e da estratégia adotada. Por fim, o Capítulo 7 apresenta a conclusão do trabalho e mostra as contribuições, limitações e os futuros direcionamentos desta pesquisa.

## 2 Trabalhos Relacionados

Este capítulo apresenta os trabalhos relacionados à análise de desempenho de AVA em computação em nuvem, considerando as metodologias e as ferramentas utilizadas. O estado da arte possui várias pesquisas relacionadas com este trabalho. Para posicionar este trabalho e indicar suas contribuições, primeiramente serão apresentados os trabalhos relacionados com a área de computação em nuvem que fazem uso de estratégias de medição. Posteriormente, serão apresentados trabalhos que utilizam estratégias de modelagem e os trabalhos que fazem uso do Moodle. Em seguida, é feita uma comparação da literatura apresentada com o que é proposto nesta dissertação, deixando mais claras as contribuições desta pesquisa.

### 2.1 Medição

(BRUSCHI et al., 2016) apresentaram um estudo de caso com o sistema Apache CloudStack utilizando o *hypervisor* XenServer e armazenamento dos dados no sistema Openfiler. Foram utilizados testes de desempenho a partir de três diferentes tipos de perfil de instâncias em uma nuvem computacional privada. O objetivo era o de medir o nível de utilização da CPU, disco E/S e memória. Como resultado, foi observado um alto consumo de disco na camada de armazenamento de dados, em particular na gravação de dados de E/S. Também foi registrado um alto consumo de memória na camada de *hypervisor*, o que é explicado pelo próprio *hypervisor*, na alocação de VMs sendo empregadas e usadas no processo.

(VOGEL et al., 2017) realizaram uma pesquisa com experimentos em instâncias utilizando *hypervisor* KVM e contêiner, em um sistema de nuvens privadas de IaaS com o sistema Apache CloudStack. Para realizar as medições, foi utilizado o NetPIPE, obtendo-se as seguintes métricas: taxa de transferência e a latência da rede. Também foi utilizado o Iperf, que é uma ferramenta de *benchmarking*, sendo uma ferramenta multiprotocolo para avaliar o desempenho de redes. Como resultado, observou-se que as instâncias de nuvem baseadas em KVM apresentaram pequena degradação do desempenho da rede em relação à taxa de transferência quando comparado com as instâncias baseadas em contêiner.

(BARROS et al., 2017) propuseram uma análise conjunta de desempenho e consumo

de energia através da medição de um sistema de armazenamento de dados distribuídos. Para gerar a carga, foi utilizado o Yahoo! *Cloud Serving Benchmark* (YCSB), possibilitando a análise de cenários variando a quantidade de operações de inserção e a quantidade de nós no *cluster* do Cassandra. Pelos resultados, verificou-se que o aumento no número de nós causa um aumento na vazão, mas, conseqüentemente, também aumenta o consumo de energia elétrica.

Autores em (MARON et al., 2014) avaliaram o desempenho das nuvens OpenStack e OpenNebula. Para isso, foi utilizado o *benchmark Iperf* para medir a rede e coletar a vazão. Os testes eram nas unidades de armazenamento com um arquivo de 100 MB analisando a escrita, reescrita, leitura e releitura, e *Stream* para testes de memória. Ao final, foi apresentado um resumo comparativo do desempenho obtido para cada ambiente a partir dos *benchmarks* avaliados. É importante destacar que foi possível detectar que o OpenStack foi recomendado para aplicações que utilizam somente o disco e o OpenNebula teve melhor desempenho para usar rede, memória e processador.

Em (VOGEL ADRIANO E GRIEBLER, 2016)), foi apresentado um trabalho que fez um estudo sobre as ferramentas de gerenciamento de IaaS, discutindo e comparando suas diferenças e desafios. Também, foi analisada a eficiência e o desempenho da computação em nuvem, além dos modelos de implantação e suporte para flexibilidade e avaliação de resiliência de três ferramentas de gerenciamento de IaaS: OpenNebula, CloudStack e OpenStack. O estudo indicou que a vantagem do CloudStack é que ele é mais flexível para implantar um serviço privado IaaS nuvem, ao passo que o ponto positivo do OpenStack é que ele é mais resiliente.

(RATNA ANAK AGUNG PUTRI E WIRIANATA, 2017) apresentaram uma estratégia de *design* de rede na nuvem, infraestrutura baseada na implementação do *Neutron* com medição da vazão, perda de pacotes e latência. A pesquisa foi realizada para investigar o posicionamento da instância no nó e a segmentação de rede que influenciam o desempenho na comunicação. Os testes para obter a taxa de transferência de parâmetros e perda de pacotes foram feitos usando o IPerf.

Autores em (WANG QIANG E XU, 2017) apresentaram um estudo sobre um *benchmark EPPMiner*, que é um protótipo com o objetivo de medir a eficiência energética de CPUs multi-core. Nesse estudo, dois processadores Intel x86, dois GPUs da Nvidia e uma GPU AMD foram analisadas, realizando-se um comparativo de consumo energético.

Uma das técnicas adotadas foi testar no *kernel* rodando por, pelo menos, 5 minutos e, em seguida, calcular a média da execução total do tempo ativo para obter os dados de desempenho.

## 2.2 Modelagem

Nos últimos anos, muitos estudos foram realizados para analisar nuvens computacionais. Em (FÉ, 2017), modelos em SPN foram propostos para auxiliar no planejamento de sistemas de transcodificação de vídeo em nuvem privada e pública. Modelos em SPN foram definidos para se estimar as seguintes métricas: vazão, tempo de resposta e custo. Para validar os resultados obtidos através dos modelos SPN, testes foram realizados no ambiente de nuvem Apache CloudStack, usando o Apache JMeter na máquina cliente para enviar as requisições de transcodificações. De acordo com os resultados, observou-se que o tempo de resposta diminuiu de 30 para 15 segundos, e o custo teve um ajuste de cerca de 300%.

(TORRES et al., 2016a) propuseram modelos em SPN, RBD e CTMC para a realização de uma avaliação integrada das métricas de desempenho e disponibilidade de um serviço de armazenamento em uma nuvem privada, gerenciada pelo sistema Eucalyptus. Os autores utilizaram as ferramentas JMeter e o Apache *benchmarking Tool* para as medições e, conseqüentemente, para a validação dos modelos propostos. Tais modelos foram validados através das métricas de vazão e nível de utilização do sistema.

(BORBA, 2017) realizou um estudo de caso baseado na plataforma OpenStack, realizando um comparativo entre duas formas de armazenamento: SSD e HD. Porém, esse estudo não trata sobre um comparativo de consumo de energia entre os dois meios de armazenamento. Um modelo SPN foi proposto para representar o sistema de armazenamento, representando as operações de gravação e leitura. Para computar a disponibilidade, foi proposto um modelo RBD.

## 2.3 Moodle

Em (SOUSA, 2015), foi realizado um planejamento para infraestruturas de nuvens privadas. Modelos e uma técnica de otimização com GRASP foram propostos para

otimizar o custo e o desempenho. Os modelos RBD e SPN propostos são capazes de realizar a estimativa de desempenho e disponibilidade. Nos estudos de caso, a autora avaliou o impacto da variação de carga, ocorrência de falhas e atividades de recuperação no ambiente de nuvem Eucalyptus, utilizando diversos cenários de configuração de *software* e de *hardware*. Na validação do modelo de desempenho, foi utilizado o AVA Moodle configurado em uma máquina virtual na plataforma Eucalyptus.

(SOUSA et al., 2017) apresentaram uma metodologia que recorre a diagramas de confiabilidade (RBD) e redes de Petri estocásticas (SPN), aliada a uma abordagem de otimização com GRASP, para auxiliar no planejamento de infraestruturas de nuvens privadas. Utilizando o Moodle hospedado em uma máquina virtual na plataforma Eucalyptus, foram computadas as seguintes métricas: disponibilidade e custo.

No trabalho de (TORRES, 2017), é proposto um método de avaliação para o planejamento da infraestrutura em nuvem privada, utilizando plataforma Eucalyptus e o ambiente de educação a distância Moodle. Ele também propõe uma arquitetura e um modelo básicos para representar e quantificar a disponibilidade e o custo. Uma análise de sensibilidade foi realizada para priorizar os componentes mais frágeis. Uma estratégia de modelagem envolvendo o uso de diagramas de blocos de confiabilidade e redes de Petri estocásticas é estabelecida. Um estudo de caso foi apresentado para analisar o método de avaliação do planejamento de infraestrutura em uma nuvem privada.

Diferentemente dos estudos anteriores, este trabalho propõe uma estratégia integrada para avaliação tanto do desempenho como do consumo de energia do AVA hospedado em diferentes infraestruturas de nuvens privadas (ex., Apache CloudStack e OpenStack). A Tabela 1 apresenta uma visão geral dos trabalhos relacionados previamente explicados. Pode-se perceber que não foi encontrado um estudo com uma abordagem integrada de desempenho e consumo de energia de AVAs hospedados em diferentes nuvens privadas. Existem muitas plataformas de nuvem computacional de *software* livre, como OpenStack, Apache CloudStack, OpenNebula e Eucalyptus (DESHMUKH; AMDANI, 2019). Para este trabalho de pesquisa foi usado o Moodle instalado em VM nas plataformas OpenStack e o Apache CloudStack.

Tabela 1 – Características dos Trabalhos Relacionados.

Trabalho	Estratégia	Métricas de Desempenho	Sistema em Nuvem	AVA
(BRUSCHI et al., 2016)	Medição	Memória, CPU, leitura em disco	Apache CloudStack	Não
(VOGEL et al., 2017)	Medição	Vazão, latência	Apache CloudStack	Não
(BARROS et al., 2017)	Medição	Latência e Consumo de energia	YCSB	Não
(FÉ, 2017)	Medição / Modelagem	Vazão, Temp. Resp. Consumo energia	Cloudstack	Não
(TORRES et al., 2016a)	Medição / Modelagem	Vazão, Número de falhas, número de atendimentos	Eucalyptus	Não
(TORRES, 2017)	Modelagem -		Eucalyptus	MOODLE
(SOUSA, 2015)	Medição / Modelagem	tempo de resposta, utilização de processador e memória	Eucalyptus	MOODLE
(SOUSA et al., 2017)	Medição / Modelagem	Memória, CPU	Eucalyptus	MOODLE
(MARON et al., 2014)	Medição	rede, memória e CPU	OpenStack e OpenNebula	Não
(VOGEL ADRIANO E GRIEBLER, 2016)	Medição	CPU, memória, armazenamento e rede	OpenNebula, CloudStack e OpenStack	Não
(RATNA ANAK AGUNG PUTRI E WIRIANATA, 2017)	Medição	CPU, RAM	OpenStack	Não
(WANG QIANG E XU, 2017)	Medição	CPU	Não	Não
(BORBA, 2017)	Medição	armazenamento SSD e HD	OpenStack	Não
Este trabalho	Medição e Modelagem	Memória, CPU, Consumo energia, vazão e tempo de resposta	Apache CloudStack e OpenStack	MOODLE

## 2.4 Considerações finais

Este capítulo apresentou os principais trabalhos correlatos ao estudo proposto. Embora existam vários trabalhos na literatura que focam em avaliação de desempenho de nuvens computacionais através de modelos analíticos, nenhum desses trabalhos foca na avaliação de desempenho com modelos analíticos e expressões matemáticas para os dois sistemas em nuvem que fazem parte desta pesquisa: Apache CloudStack e OpenStack. Alguns trabalhos apresentam uma estratégia de modelagem e dependabilidade de nuvens computacionais, mas poucos trabalhos fazem o comparativo de desempenho em nuvem computacional. Embora existam trabalhos que apresentem métricas diferentes, nenhum trabalho estuda as métricas que este trabalho utiliza com as duas nuvens computacionais utilizando o Moodle como serviço.



## 3 Fundamentação Teórica

Este capítulo apresenta os conceitos para uma melhor compreensão deste trabalho. Inicialmente, são apresentados os conceitos sobre computação em nuvem, a classificação de serviços fornecidos em computação em nuvem, a classificação de modelos de computação em nuvem, e uma visão geral. Em seguida, são mostradas as definições sobre o Apache CloudStack e OpenStack, os conceitos básicos sobre avaliação de desempenho. Na sequência, são abordados conceitos sobre e Apache JMeter e redes de Petri. Por fim, são abordados conceitos sobre regressão linear e o Ambiente Virtual de Aprendizagem Moodle.

### 3.1 Computação em Nuvem

O *data center* é o elemento central de processamento e armazenamento na nuvem. A virtualização é a tecnologia central do data center, pode-se dizer que a virtualização é uma das principais tecnologias da computação em nuvem. Graças a esta tecnologia, a abstração de recursos físicos pode ser realizada de forma que vários *hosts* virtuais podem ser criados como um único *hosts* físico ou como vários *hosts* físicos. O *hosts* virtual possui características semelhantes ao *hosts* físico, isso pode otimizar os recursos de computação, de modo que os recursos do *hosts* físico possam ser particionados para fornecer serviços para vários clientes (PEIXOTO, 2012).

A computação em nuvem pode ser vista como um modelo para acesso ubíquo, conveniente e sob demanda. Ela representa uma rede compartilhada de recursos computacionais configuráveis (por exemplo, redes, servidores, armazenamento, aplicações e serviços) que podem ser rapidamente adquiridos e liberados com o mínimo esforço gerencial ou a mínima interação com o provedor dos serviços (NIST, 2014). A computação em nuvem oferece, então, vantagens como compartilhamento de serviços, sistemas operacionais e aplicativos em rede, facilitando o gerenciamento dos modelos de negócio sem se preocupar com detalhes da infraestrutura (MACHIDA et al., 2011).

Os três modelos de serviços em nuvem computacional podem ser classificados de acordo com o seu modelo de negócio (ZOTA; PETRE, 2014):

- Infraestrutura como Serviço (*Infrastructure as a Service - IaaS*): É um modelo de serviço que fornece recursos de computação como hardware, armazenamento e rede.

- Plataforma como Serviço (*Platform as a Service - PaaS*): É um modelo de serviço que o usuário pode utilizar para o desenvolvimento e execução de aplicações.
- Software como Serviço (*Software as a Service - SaaS*): É um modelo de serviço onde são fornecidos aplicativos executados em uma infraestrutura de nuvem.

A grande disponibilidade de recursos e serviços, além do crescente acesso de usuários à computação em nuvem, vêm favorecendo o aumento acelerado de hospedagem em diversas soluções de TI. No entanto, existe uma variedade de configurações possíveis e, assim, é importante avaliar tais configurações.

Esses serviços compartilhados são divididos em: nuvem pública, privada, comunitária e híbrida (ZHANG et al., 2010) (NIST, 2014).

- Nuvem pública: é aplicada quando sua finalidade é compartilhar serviços e recursos pela internet;
- Nuvem privada: é aplicada quando sua infraestrutura está totalmente (ou maior parte) dentro de uma estrutura privada, como serviços disponíveis que tenham a finalidade somente para uma única organização;
- Nuvem comunitária: a infraestrutura da nuvem é compartilhada por várias organizações. Neste modelo, um grupo de pessoas ou empresas que geralmente têm interesses comuns se reúnem para criar uma nuvem que é compartilhada apenas entre os membros dessa comunidade. Os recursos estão nas mãos de um ou mais participantes da comunidade;
- Nuvem híbrida: é aplicada quando sua infraestrutura está disponível agregando modelos de nuvens pública e privadas. Essas arquiteturas podem ser utilizadas de acordo com a finalidade de disponibilização de serviços, podendo utilizar apenas um dos modelos ou mais de um.

Assim sendo, para o gerenciamento desses serviços, plataformas de computação em nuvem como o Apache CloudStack (APACHE1, 2020), OpenNebula (SYSTEMS, 2020) e OpenStack (OPENSTACK, 2020) foram criadas. A seguir, apresentamos mais informações sobre Apache CloudStack e, em seguida, sobre o OpenStack, que são os ambientes avaliados neste trabalho pelo seu amplo nível de utilização (MULLERIKKAL; SASTRI, 2015).

### 3.1.1 Apache CloudStack

O Apache CloudStack é um *software* de código aberto, projetado para implantar e gerenciar amplas redes de máquinas virtuais, com uma IaaS altamente escalável e disponível (PAURO, 2016). O Apache CloudStack gerencia a computação, a rede e os recursos de armazenamento. O CloudStack utiliza diversos *hypervisors* como KVM, vSphere e XenServer para virtualização, e também tolera a API do AWS, além de suas próprias APIs (KUMAR et al., 2014). Os recursos dentro da infraestrutura em nuvem do Apache CloudStack são denominados por: regiões, zonas, *Pods*, clusters, *hosts*, armazenamento (SABHARWAL, 2013). A Figura 1 ilustra a relação entre cada um desses itens. A seguir, uma explicação mais detalhada também é fornecida.

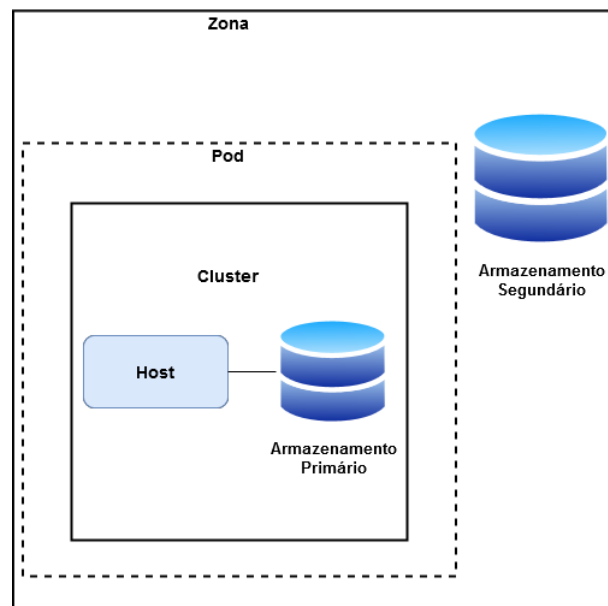


Figura 1 – Infraestrutura em nuvem do Apache CloudStack

- Regiões: uma coletânea de uma ou mais zonas geograficamente próximas gerenciadas por um ou mais servidores. Cada região é controlada pelo seu próprio conjunto de servidores de gerenciamento, que precisam executar em uma das zonas. Regiões é uma tecnologia útil para fornecer tolerância a falhas e recuperação de desastres;
- Zonas: é análogo a um único *data center*. Uma zona consiste em uma ou mais *Pods* e armazenamento secundário. A vantagem de organizar a infraestrutura em Zonas é fornecer isolamento físico e redundância. Zonas são visíveis para o usuário final e podem ser privadas ou públicas. Zona pública é visível para todos os usuários na nuvem, enquanto que a zona privada é visível apenas para os usuários de um

determinado domínio;

- *Pods*: um *pod* de regra é um rack ou fileira de racks que inclui um *switch* de camada 2 e um ou mais clusters. Ele possui um ou mais *hosts*, e um ou mais servidores de armazenamento primários. Os *Pods* estão inseridos dentro de uma Zona, e cada Zona pode conter um ou mais *Pods*;
- Clusters: um cluster incide em um ou mais *hosts* homogêneos e com armazenamento primário. Assumindo que os nós de computação estão no mesmo cluster, eles possuem o mesmo *hardware* e *hypervisor*, e se estiverem na mesma sub-rede, poderão acessar o armazenamento primário compartilhado;
- Hosts: um único nó de computação dentro de um cluster; muitas vezes um *hypervisor*. O host físico é virtualizado pelo *hipervisor*. CloudStack oferece suporte a vários *hipervisores*, para gerenciar máquinas virtuais entre eles: Xen, KVM, vSphere, o Hyper-V, VMWare;
- Armazenamento Primário: é uma solução de armazenamento fornecido a um único cluster para a execução real de imagens de disco da instância.
- Armazenamento Secundário: sendo um recurso de toda a zona que armazena imagens de disco (ex., imagens ISO).

### 3.1.2 OpenStack

O OpenStack é uma plataforma com conjunto crescente de soluções de *software* colaborativo. Foi criado pela NASA para gerenciar e armazenar dados em uma infraestrutura virtualizada tanto em nuvem privada como em nuvem pública. Esses serviços oferecidos pelo OpenStack permitem a virtualização do processamento (computação) e da rede como Serviço (IaaS), formando um conjunto de serviços de *software* distribuídos e interconectados.

O OpenStack apresenta as seguintes características: escalabilidade, flexibilidade, e é *open source*. Dessa forma, é possível redefinir os patamares de velocidade de implementação, custo e escalabilidade dos serviços, oferecendo flexibilidade em suas decisões tecnológicas. A Figura 2 apresenta a infraestrutura dos principais componentes do sistema em nuvem OpenStack (BRUSCHI et al., 2016). A seguir, mais detalhes sobre os seus componentes são apresentados.

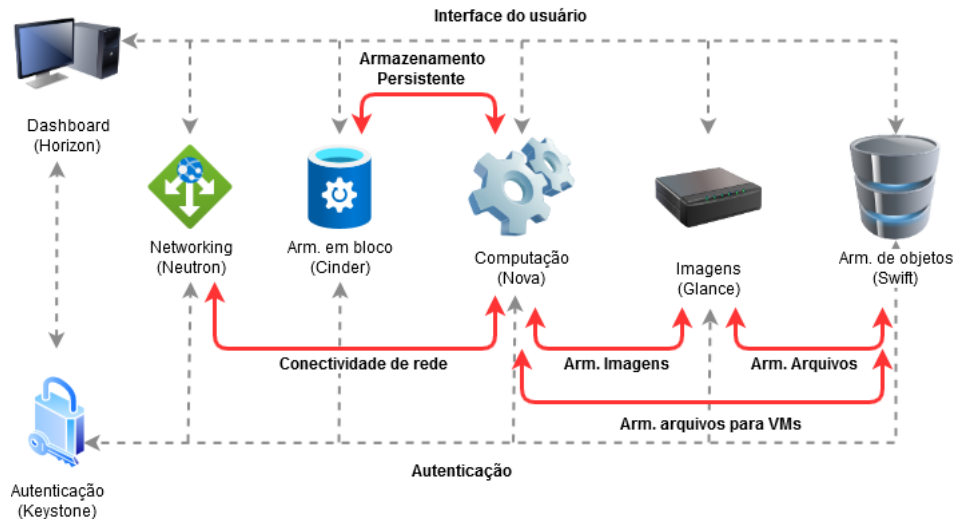


Figura 2 – infraestrutura em nuvem do OpenStack

- *Keystone* (Serviço de Autenticação): disponibiliza a autenticação e autorização para todos os serviços;
- *Glance* (serviço de imagem): garantindo um catálogo e repositório de imagens de discos virtuais, que são utilizados pelas VM;
- *Nova* (serviço de computação): disponibiliza as máquinas virtuais à medida que são demandadas pelos clientes;
- *Neutron* (serviço de rede): proporciona conectividade de rede como um serviço, permite acesso a uma API para requerer e configurar redes virtuais;
- *Horizon* (serviço de painel-*Dashboard*): é uma interface para visualizar todos os serviços;
- *Cinder* (bloco de armazenamento): fornece armazenamento em nível de bloco, que pode ser montado como volume pelas instâncias (servidor virtual);
- *Swift* (Objeto de armazenamento): dispendo um serviço de armazenamento de objetos, por via os clientes podem armazenar ou buscar seus arquivos. (Vogel et al., 2016).

### 3.2 Avaliação de Desempenho

A avaliação de desempenho pode ser realizada via medições, modelagem e simulações (CALLOU et al., 2011). A mensuração de um sistema é o método mais direto para a avaliação de desempenho. As técnicas de medição, apesar de fornecerem respostas exatas sobre o desempenho do sistema, não podem ser utilizadas (ou são muito caras) se um

sistema estiver em operação. Além disso, embora a carga de trabalho real seja uma boa escolha por representar fielmente o sistema, às vezes essa opção não é viável. Isso pode acontecer, por exemplo, quando o tamanho da carga não é representativo com os dados coletados sujeitos a muitas interferências externas. A fim de facilitar a vida dos projetistas que realizam medições, existem alguns mecanismos usados para auxiliar na geração da carga de trabalho (LILJA, 2005; SOUSA, 2015).

A escolha da carga de trabalho é tão importante quanto a definição da estratégia de medição a ser seguida. Diferentes estratégias de medição são baseadas no conceito de eventos, que são mudanças no estado do sistema. A definição exata do evento depende de Métricas. Por exemplo, os eventos podem ser definidos como acessos a disco (*upload* de um arquivo), acesso de memória ou comunicação de rede de computadores. Assim, as ferramentas utilizadas para avaliação de desempenho de sistemas computacionais via medição podem modificar o comportamento do objeto em teste. Quanto maior a informação e resolução que as ferramentas de medição podem fornecer, maior pode ser a interferência introduzida por esta ferramenta de medição. Essa interferência introduzida pela ferramenta de medição pode reduzir a qualidade dos dados aferidos (LILJA, 2005; SOUSA, 2015). No entanto, também existe a possibilidade de se realizarem medições via *hardware* externo ao ambiente que está sendo analisado. E, assim, adotando essa estratégia, o impacto sob o sistema analisado não ocorre.

Já a modelagem analítica é utilizada para a abstração do sistema. Assim, o modelo deve conter detalhes do sistema que são essenciais para o seu comportamento. Por fim, a simulação do modelo poderá fornecer resultados não tão precisos quanto os fornecidos pela medição, mas é possível calcular as estimativas considerando as precisões dos erros desejados. Vale ressaltar que existem diversos tipos de modelos formais que podem ser utilizados para quantificar as métricas de desempenho desejadas.

A simulação pode ser usada para avaliação de desempenho e verificação de modelos analíticos. Ao contrário das medições, as simulações são baseadas em modelos abstratos do sistema, portanto, podem ser aplicadas sem a necessidade de implantar totalmente o sistema. Portanto, o modelo utilizado no processo de simulação é concluído abstraindo as características básicas do sistema, e sua complexidade, e o grau de abstração adotado pode variar de um sistema para outro. No processo de simulação, os valores assumidos dos parâmetros do sistema podem ser controlados. Isso facilita obter informações relacionadas

à avaliação de desempenho (LILJA, 2005).

A Figura 3 mostra que existem várias formas de se realizar a análise de desempenho. Dependendo das condições de tempo e recursos, o uso de mais de uma técnica pode permitir a compreensão melhor do sistema e fornecer informações mais precisas para auxiliar nas tomadas de decisões. Com a ajuda de ferramentas de modelagem, simulação e medição, informações detalhadas sobre o sistema podem ser obtidas. Usando a tecnologia de medição, os resultados mais precisos podem ser obtidos porque é a tecnologia que melhor corresponde ao ambiente real sendo analisado. No entanto, a modelagem fornece vários métodos alternativos de avaliação de desempenho, que podem ser obtidos através de simulação ou modelagem analítica, onde os modelos analíticos lidam com métodos probabilísticos, teoria de filas, modelos de Markov ou redes de Petri (CALLOU et al., 2011). Vale destacar que a modelagem e simulação permite a avaliação do sistema para diferentes cenários, ainda não disponíveis no sistema em análise.

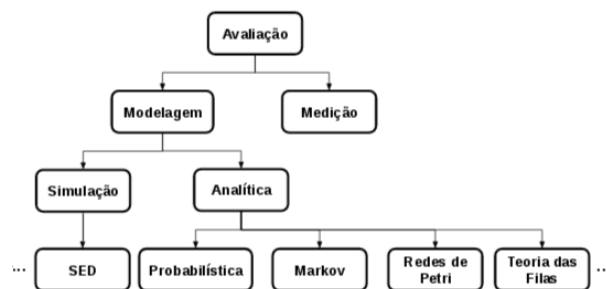


Figura 3 – Classificação das técnicas de análises de desempenho (CALLOU, 2009)

### 3.2.1 Apache JMeter

O Apache JMeter é um aplicativo Java de código aberto, cuja função é gerar carga ao sistema, analisando e medindo o desempenho de recursos estáticos e dinâmicos. Tal técnica permite que o aplicativo Apache JMeter gere requisições, realizando simulações de grupos de servidores, redes e objetos, por exemplo, analisando o desempenho do sistema em diferentes situações de cargas (JMETER, 2019). O Apache JMeter permite medir o tratamento para a execução dos experimentos e coleta das métricas de desempenho. Esse gerador pode ser utilizado, por exemplo, para gerar carga de requisições de usuário, simulando diversos níveis de carga nas atividades requeridas no Moodle, que representa a plataforma virtual de aprendizagem adotada neste trabalho. O JMeter é capaz de analisar

o desempenho e possíveis interrupções ocasionado pelo acesso múltiplo de usuários ao Moodle (SOUSA, 2015).

Neste trabalho, são utilizadas técnicas de modelagem com redes de Petri estocásticas e técnicas de medição através da ferramenta Apache JMeter. A medição foi adotada com o objetivo de validar o modelo proposto ao se obterem valores reais, utilizados para comparar o desempenho e o consumo de energia obtidos a partir do modelo.

### 3.3 Redes de Petri

O conceito de redes de Petri foi introduzido por Carl Adam Petri, em 1962, como uma técnica gráfica e formal para representar e simular o comportamento de sistemas (PETRI, 1962). Desde então, tem sido utilizada para modelagem e análise dos sistemas em várias áreas do conhecimento. Essas redes permitem descrever e analisar sistemas de processamento de informações que são caracterizados como sistemas concorrentes, assíncronos, distribuídos, paralelos, não determinísticos e ou estocásticos. As redes de Petri representam uma técnica visual bastante utilizada para auxiliar no entendimento do comportamento de sistemas, que faz uso de *Tokens* para simular a dinâmica e as atividades concorrentes do sistema (MURATA, 1989).

A rede de Petri é uma técnica de análise e modelagem de sistemas complexos amplamente difundida. Esta técnica possui representação matemática e mecanismos de análise que permitem a verificação de características dos sistemas. As redes de Petri básicas são compostas por lugares, transições, arcos e tokens (ver Figura 4). Contudo, a necessidade de atender a outras áreas do conhecimento fez com que outras extensões deste modelo surgissem ao longo do tempo, como as redes de Petri temporizadas (Merlin; Farber, 1976; NETO, 2015).

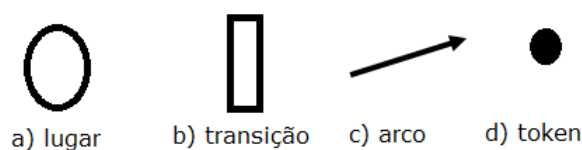


Figura 4 – Elementos da Redes de Petri

A Rede de Petri, além de permitir a representação matemática dos sistemas,



possibilita a análise dos modelos, proporcionando informações úteis sobre a estrutura da rede. Além disso, ela é capaz de analisar comportamentos dinâmicos e permite a realização de sincronismos, proporcionando mecanismos de comunicação, bem como a representação de atrasos determinísticos e probabilísticos que podem ser naturalmente representados.

### 3.3.1 Redes de Petri Estocástica

As redes de Petri estocásticas (SPN) são uma extensão das redes de Petri, e podem ser empregadas, por exemplo, para a modelagem de desempenho e dependabilidade (TORRES et al., 2016a). As SPN passam a ter duas representações distintas para as transições, como pode ser visto na Figura 5. Existe a transição estocástica, Figura 5 (a), onde um tempo é associado à transição para representar o período de execução da atividade. A Figura 5 (b) mostra a transição imediata que não demanda tempo para o disparo (tempo igual a zero), ou seja, o disparo é instantâneo (SILVA, 2012).

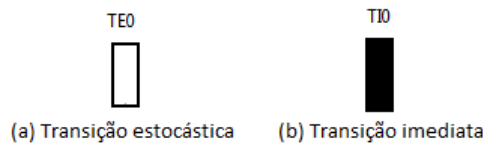


Figura 5 – Elemento da SPN

A definição de uma rede de Petri estocástica é uma 9-tupla  $SPN = \{P, T, I, O, H, \Pi, G, M_0, Atts\}$  (GERMAN, 2000), onde:

- $P = \{p_1, p_2, \dots, p_n\}$  é um conjunto finito de lugares;
- $T = \{t_1, t_2, \dots, t_n\}$  é um conjunto finito de transições;
- $I \in (\mathbb{N}^n \rightarrow \mathbb{N})^{m \times n}$  é a matriz que representa a multiplicidade dos arcos de entrada (que podem ser dependentes de marcações);
- $O \in (\mathbb{N}^n \rightarrow \mathbb{N})^{m \times n}$  é a matriz que representa a multiplicidade dos arcos de saída (que podem ser dependentes de marcações);
- $H \in (\mathbb{N}^n \rightarrow \mathbb{N})^{m \times n}$  é a matriz que representa os arcos inibidores (que podem ser dependentes de marcações).
- $\Pi \in \mathbb{N}^m$  é um vetor vincula o nível o nível de prioridade para cada transição;
- $G \in (\mathbb{N}^n \rightarrow \{true, false\})^m$  é o vetor que associa uma condição de guarda relacionada à marcação do lugar a cada transição;

- $M_0 \in \mathbb{N}^n$  é o vetor que associa uma marcação inicial de cada lugar (estado inicial);
- $Atts = (Dist, W, Markdep, Policy, Concurrency)^m$  é o vetor que associa uma marcação inicial de cada lugar (estado inicial), onde:
  - $Dist \in \mathbb{N}^m \rightarrow f$  é uma função de distribuição de probabilidade associada ao tempo de uma transição, onde o domínio de  $f[0, \infty)$  (esta distribuição pode ser dependente de marcação);
  - $W \in \mathbb{N}^m \rightarrow \mathbb{R}^+$  é uma função de peso (esta distribuição pode ser dependente de marcação);
  - $Markdep \in \{constant, enabdep\}$  é definido como a distribuição de probabilidade associada ao tempo de uma transição constante (*constant*) ou dependente de marcação (*enabdep*);
  - $Policy \in \{prd, prs\}$  é a política de memória adotada pela transição (*prd* - *preemptive repeat different*, corresponde a *race enabling policy*; *prs* - *preemptive resume*, corresponde a *age memory policy*).
  - $Concurrency \in \{ss, is\}$  é o grau de concorrência das transições, onde *ss* representa a semântica de *single server* e *is* representa a semântica de *infinity server*.

A Figura 6 mostra um exemplo de uma SPN que representa o funcionamento de um sistema operacional de um computador. Para isso, definimos dois lugares que representam os estados que o sistema pode ter (*Sistema\_Ligado* ou *Sistema\_Desligado*), e duas transições que modelam as ações para alterar o estado do sistema (*Ligar* e *Desligar*).

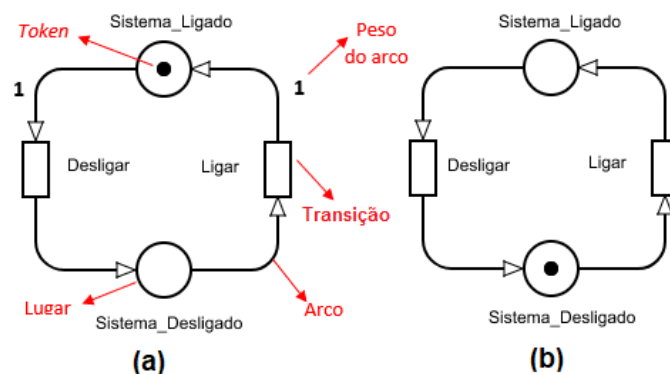


Figura 6 – Exemplo de uma Rede de Petri (RdP)

Inicialmente, temos o sistema em funcionamento, o que é representado por um Token no lugar *Sistema\_Ligado*, conforme pode ser visto na Figura 6 (a). No estado atual

do modelo, a única transição que pode ser disparada é *Desligar*. Assim que esta transição for disparada, o modelo passará do estado Ligado, como representado na Figura 6 (a), para o estado Desligado, conforme pode ser visto na Figura 6 (b). Vale destacar que as transições *Ligar* e *Desligar* tem um tempo associados, em outras palavras, esse tempo representa a duração de cada atividade.

### 3.4 Regressão Linear

A regressão linear é uma técnica estatística usada para relacionar uma variável  $x$ , independente, com uma variável  $y$  que é dependente (MATOS, 1995). Este trabalho fará uso da regressão linear no modelo SPN proposto para a simulação do consumo de energia, vazão e tempo de resposta, por exemplo. A regressão linear simples é dada pela seguinte Equação 3.1.

$$y = a + bx \quad (3.1)$$

onde  $y$  é o resultado da saída prevista,  $x$  é a variável de entrada e  $a$  e  $b$  são os parâmetros de regressão.

O cálculo do parâmetro de regressão  $b$  é dada pela seguinte Equação 3.2.

$$b = \frac{n \sum_{i=1}^n x_i y_i - (\sum_{i=1}^n x_i)(\sum_{i=1}^n y_i)}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \quad (3.2)$$

onde  $n$  é quantidade de pontos  $(x_i; y_i)$ .

Já o cálculo do parâmetro de regressão  $a$  é dada pela seguinte Equação 3.3.

$$a = \bar{y} - b\bar{x} \quad (3.3)$$

onde  $\bar{y}$  e  $\bar{x}$  é a média aritmética da variável de saída e de entrada, respectivamente.

Existe um grau de erro associado à regressão que é chamado de coeficiente de determinação e é representado por  $R^2$  (LILJA, 2005). Esse grau de erro varia entre zero e um. Quanto mais próximo o valor do coeficiente de determinação estiver de um, maior

será a validade da regressão. O cálculo do coeficiente de determinação é dado pela seguinte Equação 3.4.

$$R^2 = \frac{b \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (3.4)$$

### 3.5 Moodle

Um dos exemplos do crescimento acelerado de sistemas hospedados na nuvem é o Moodle, ambiente virtual de aprendizagem colaborativo, que facilita o acompanhamento e permite a gestão do ensino e aprendizagem para alunos e professores. Muitas instituições e pessoas estão utilizando o AVA Moodle, tanto em cursos online como em cursos presenciais, seja em universidades ou em cursos profissionalizantes. Esse crescimento na utilização pode ser explicado pela busca frequente de flexibilidade de horário e espaço para realização dos cursos, tornando-se cada vez mais popular entre educadores e estudantes. Fora isso, em momentos de pandemia, o ensino a distância cresceu em importância.

Em (SOUSA, 2015), é apresentada uma descrição sobre o Moodle explicando a função do sistema aberto de gerenciamento de aprendizagem, bem como suas ferramentas. O Moodle é um AVA amplamente utilizado no meio acadêmico, tanto nos cursos a distância quanto nos presenciais. O Moodle é um sistema *open source* de gerenciamento de aprendizagem e de trabalho colaborativo em ambiente virtual que permite a criação e administração de cursos *online*, grupo de trabalho e comunidades de aprendizagem. Esse AVA também fornece ferramentas de avaliação específicas, como, por exemplo: discussões de fórum, lições com questões, entradas de glossário, entre outros. As principais funcionalidades são: fórum, *chat*, glossário, lição, questionário e Wiki (MOODLE, 2020).

Portanto, o crescente número de usuários em cursos a distância em instituições acadêmicas que abrange vários municípios e estados no país e, também, no mundo pode demandar um estudo mais aprofundado sobre o desempenho de tais sistemas para poder suportar a demanda crescente, por exemplo. Além disso, vão sendo previsíveis possíveis indisponibilidades de acessos para recursos disponíveis em nuvem. No entanto, esses problemas podem ser minimizados a partir da utilização de ferramentas que realizam monitoramento de forma determinística e probabilística para gerenciamento de possíveis soluções em TI com a utilização de estratégias de modelagem.

### 3.6 Considerações finais

Este capítulo apresentou uma introdução sobre computação em nuvem, considerando aspectos relacionados aos modelos dos serviços oferecidos, à classificação dos modelos e às plataformas de nuvens privadas adotadas. Conceitos básicos sobre o Apache CloudStack e OpenStack foram apresentados. Em seguida, apresentou conceitos sobre avaliação de desempenho, Apache JMeter e redes de Petri. Posteriormente, foram apresentados conceitos sobre regressão linear e, por fim, este capítulo apresentou o Moodle.

## 4 Metodologia

Este capítulo apresenta a metodologia utilizada, para a avaliação de desempenho do AVA hospedado em nuvens privadas. A metodologia descreve as atividades realizadas e os objetos gerados em cada uma das atividades. Em seguida, como foi realizada a coleta de dados. Por fim, apresenta a infraestrutura utilizada para os experimentos.

### 4.1 A metodologia

A metodologia é ilustrada na Figura 7. A seguir, cada atividade é detalhada.

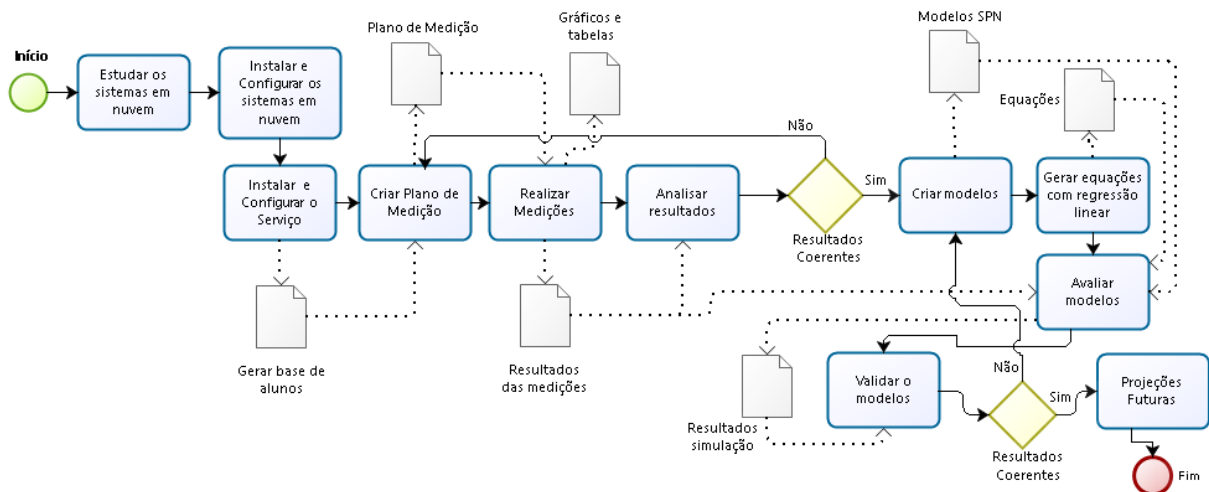


Figura 7 – Metodologia adotada.

- **Estudar os sistemas em nuvem:** o propósito desta fase é fazer um estudo com a finalidade de se obter uma compreensão da infraestrutura em nuvem a ser analisada e seus componentes. Além disso, nessa fase também devem ser analisados os serviços que serão configurados na nuvem (ex., Moodle) e as métricas que serão usadas para avaliar o desempenho.
- **Instalar e configurar os sistemas em nuvem:** nesta atividade são realizadas a instalação e configuração da infraestrutura em nuvem (ex., Apache CloudStack e o OpenStack).
- **Instalar e configurar o serviço:** esta fase corresponde à atividade de instalação e configuração de um serviço na máquina virtual de cada um dos sistemas em nuvem a serem comparados.

- **Criar o plano de medição:** nesta atividade, um plano de medição é criado, na ferramenta Apache JMeter, com o objetivo de realizar os experimentos no ambiente em análise. Essa atividade é responsável, ainda, pela criação das bases de dados que serão utilizadas; por exemplo, pode-se criar a base de alunos, professores, cursos, turmas. Note-se que todos os dados inseridos são hipotéticos. O plano de medição é usado para definir as atividades a serem executadas, como, por exemplo, acessar o Moodle, realizar *login*, acessar o curso, acessar a atividade, realizar *upload* de arquivos com 1 Mb e sair do AVA. É nessa atividade que devem ser definidas a quantidade de usuários que irão acessar o sistema e o tempo de inicialização para cada um.
- **Realizar medições:** esta atividade corresponde às medições utilizando JMeter e/ou NMON, de acordo com o plano de medição criado na atividade anterior. Note-se que o Apache JMeter é responsável pela coleta de métricas como vazão e tempo de resposta. Já o NMON pode ser utilizado para quantificar a utilização da CPU e da memória RAM, por exemplo. As medições devem ser repetidas, no mínimo, trinta vezes para aumentar a confiabilidade dos resultados obtidos.
- **Analisar resultados:** nesta etapa, devem ser feitos os cálculos estatísticos da média, desvio padrão e intervalo de confiança. E, ao final, são gerados gráficos de comparação dos resultados, por exemplo, do tempo de resposta, vazão, utilização de CPU, utilização da memória RAM e consumo de energia.
- **Resultados coerentes:** se o resultado for Não, volta-se para a fase de Criar Plano de Medição. Caso o resultado seja Sim, segue-se para a fase Criar modelos.
- **Criar modelos:** esta fase refere-se à criação de modelos de desempenho. Vale salientar que estes modelos podem ser feitos, por exemplo, utilizando o formalismo das SPNs. Vale destacar que esse modelo poderá ser refinado na fase seguinte, com o emprego de técnicas de regressão linear. Note-se que, através da regressão linear, é possível conseguir representar o tempo demandado para diferentes atividades no sistema. Sendo assim, a transição do modelo SPN que representa tal atividade irá possuir uma expressão resultado da regressão linear realizada.
- **Gerar equações com regressão linear:** é nesta fase que as equações de regressão linear são geradas. Por exemplo, pode-se fazer o cálculo da regressão linear entre a quantidade de clientes no sistema e o tempo de atendimento, ou ainda, entre o nível

de utilização da CPU e o consumo de energia decorrente. Então, é nesta fase que são geradas as equações que serão utilizadas para refinar o modelo de desempenho em SPN.

- **Avaliação do modelo:** esta etapa é responsável por realizar a avaliação do modelo proposto, considerando os cenários adotados no plano de medição.
- **Validar o modelo:** para se realizar a validação do modelo proposto, é adotada a seguinte estratégia: (i) deve-se observar o intervalo de confiança para cada métrica de interesse, tanto nos resultados do modelo quanto nos resultados das medições; (ii) comparam-se os intervalos de confiança e, se estes estiverem se sobrepondo, pode-se confirmar que não existe diferença estatística significativa, com 95% de confiança, por exemplo, entre os resultados do modelo e os resultados dos experimentos.
- **Resultados coerentes:** se o resultado for Não, volta-se para a fase de Criar Modelos. Caso o resultado seja Sim, segue-se para a fase Projeções futuras.
- **Projeções futuras:** uma vez validado o modelo, pode-se utilizá-lo para fazer estimativas inviáveis de serem feitas em medições. Assim, o modelo pode ser utilizado para análises mais complexas. Esta é uma das grandes vantagens de se utilizar a modelagem.

Dessa forma, este trabalho propõe expor uma modelagem que irá avaliar a métrica de desempenho, no *software* de nuvem privada, utilizando o Apache CloudStack e o OpenStack através de Redes estocásticas de Petri (SPN). Assim, busca analisar as possibilidades de uma infraestrutura adequada, para suportar acesso de múltiplos usuários no Ambiente Virtual de Aprendizagem . Para isso, serão descritos, a seguir, os métodos e as técnicas necessários para a validação e execução da pesquisa.

## 4.2 Coleta de Dados

Para a coleta dos dados da presente pesquisa, é utilizado o Apache JMeter, um *software* de código aberto, projetado para medir e coletar experimentos. A Figura 8 mostra um exemplo da tela inicial do JMeter, que pode ser usado para testar o desempenho em recursos estáticos e dinâmicos ([APACHE, 2019](#)).

O Apache JMeter foi utilizado para testar o desempenho dos *softwares* de nuvens, com Ambiente Virtual de Aprendizagem hospedado em uma máquina virtual e em diferentes



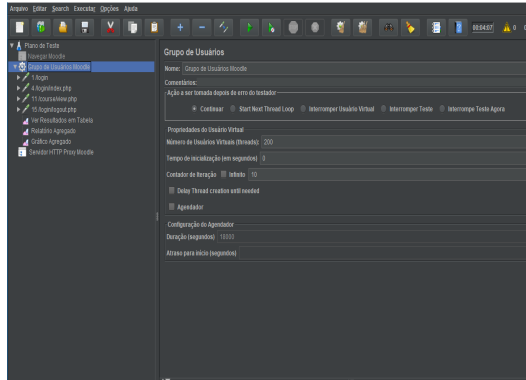


Figura 8 – Tela inicial do Apache JMeter

cenários. No Cenário 01, Apache CloudStack foi utilizado e testada a sua capacidade de suportar múltiplos usuários simultaneamente, levando-se em consideração o ambiente Moodle. No Cenário 02, o OpenStack foi utilizado levando-se em consideração os mesmos experimentos realizados com o Apache CloudStack. Para chegar a um valor limite que o ambiente hospedado na nuvem privada suporta, o Apache JMeter foi configurado de maneira crescente em 25, 50, 75 e 100 usuários simultâneos acessando o ambiente de teste. Durante os experimentos, foi observado que, com mais de 100 usuários no sistema, e com a configuração do *hardware* utilizado, ocorria uma sobrecarga, ocasionado assim perdas de pacotes. Então, dadas as limitações, definimos o número máximo de usuários que poderíamos trabalhar. Já a observação, para ser considerada um instrumento de coleta de dados, deve servir a um objeto preestabelecido de pesquisa, ser planejada, ser registrada de forma sistemática e ser passível de verificação quanto ao seu grau de precisão (PRODANOV; FREITAS, 2013). A observação foi utilizada buscando-se identificar o momento em que o ambiente não mais suportará os acessos dos usuários ao ambiente de Aprendizagem virtual de aprendizagem.



Figura 9 – Watts Up Pro meter - Coletor de dados do consumo de energia

A Figura 9 mostra o equipamento utilizado para coletar o consumo de energia. Nesse ambiente, somente a máquina com a nuvem privada em execução permanece ligada diretamente a este dispositivo. Sendo assim, apenas o consumo de energia da máquina que receberá as requisições do Apache JMeter será coletado por este equipamento. Vale destacar que são executadas, no mínimo, 30 coletas para cada experimento. Através deste número de repetições, o experimento terá um intervalo de confiança maior, ou seja, o resultado ficará mais próximo da situação real (MORAIS et al., 2020).



Figura 10 – Estrutura utilizada no laboratório de pesquisa do PPGIA

A Figura 10 ilustra a estrutura física utilizada no laboratório de pesquisa do PPGIA na UFRPE. Esse ambiente é composto por cinco máquinas, sendo uma utilizada como cliente, duas com o *software* Apache CloudStack e outras duas com o *software* OpenStack, conectadas por um *switch* Gigabit e um roteador para disponibilizar o acesso à *internet*.

### 4.3 Infraestrutura

A Figura 11 ilustra a visão geral do ambiente montado para a realização dos experimentos. O ambiente é composto por três máquinas, sendo (i) uma cliente, com Apache JMeter para gerar a carga de trabalho e quantificar as métricas de desempenho; (ii) uma máquina com o ambiente de nuvem gerenciado pelo Apache CloudStack; e (iii) uma máquina com o ambiente de nuvem gerenciado pelo OpenStack. Em cada máquina com sistema de nuvem foi criada uma VM, configurado o Moodle e a ferramenta de medição NMON. É importante destacar que a comunicação entre essas máquinas é feita por um *switch* Gigabit. Além disso, pode-se observar a existência de um medidor para quantificar

o consumo de energia das máquinas com o ambiente de nuvem.

Todas as máquinas utilizadas são equivalentes e são compostas por CPU Intel Core i5 de 3,4 GHz, 8 MB Cache, 8 GB RAM, NIC 1 Gigabit Ethernet 10/100/1000, 500 HD. O sistema operacional adotado foi o CentOS 7. Para ambas as nuvens foram instalados o hypervisor KVM. A VM utilizada foi configurada com processador de 2 GHz, com 2 CPUs e 4 GB de memória RAM, 20 GB de espaço em disco e com o sistema operacional Linux CentOS 7 e banco de dados MySQL.

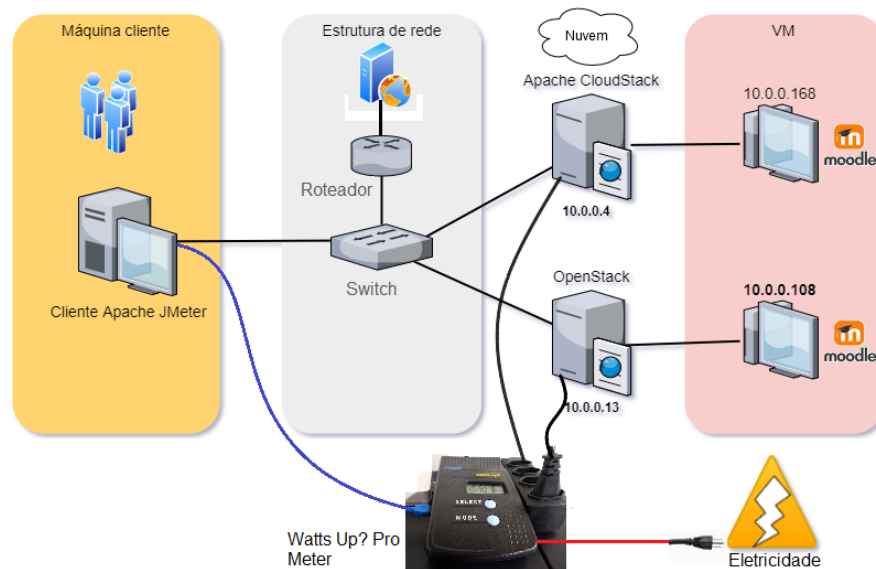


Figura 11 – Infraestrutura montada para os experimentos.

#### 4.4 Considerações finais

Este capítulo apresentou a metodologia e as atividades para a realização deste trabalho. A descrição das atividades da metodologia. Também foram apresentadas as ferramentas de coleta de dados. Finalmente, infraestruturas do ambiente em nuvem no laboratório utilizada para a realização dos experimentos.

## 5 Modelo

Este capítulo inicia com a apresentação do modelo proposto, em SPN, para a quantificação do desempenho dos sistemas: Apache CloudStack e OpenStack com o Moodle hospedado. Na descrição do modelo, são apresentados os lugares e as descrições das transições. Em seguida, são mostradas as regressões lineares para a quantificação da utilização da CPU, do consumo de energia e do tempo de resposta para cada sistema de nuvem, tempo de processamento, tempo de latência. Por último, são apresentadas as métricas para utilização de CPU, utilização da memória RAM, vazão, tempo de resposta, do Apache CloudStack e OpenStack.

### 5.1 Modelo de Desempenho

A Figura 12 mostra o modelo SPN de desempenho proposto para representar o comportamento do Moodle da infraestrutura apresentada na Figura 11. O modelo representa o funcionamento da chegada de requisições dos usuários até a resposta do sistema. Dessa forma, usuários enviam requisições, as quais passam pela rede, acessam a memória e são processadas na CPU.

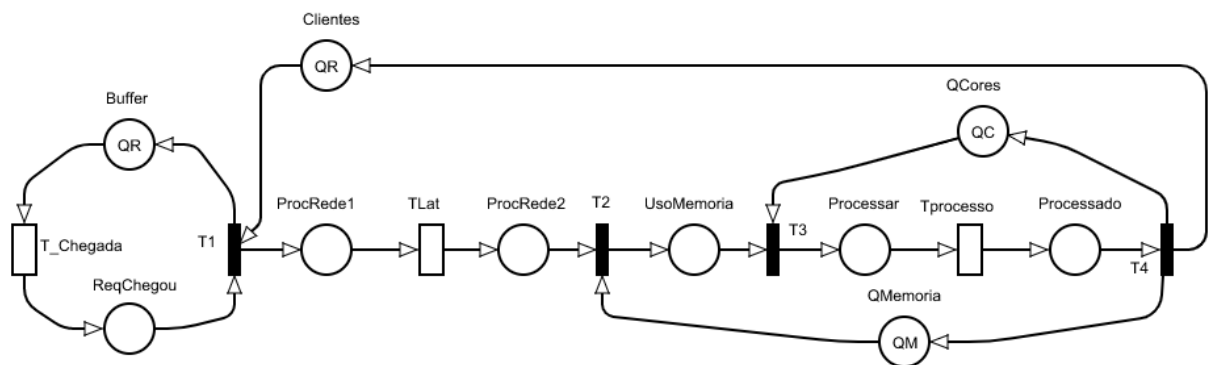


Figura 12 – Modelo de desempenho proposto

O número de tokens ( $QR$ ) presente no lugar *Clientes* representa a quantidade de clientes utilizando o sistema Moodle, simultaneamente. O lugar *Buffer* armazena os clientes, a transição  $T\_Chegada$  possui o tempo de chegada e, ao ser disparada, consome um token de *Buffer* e dispara um token para *ReqChegou*. A transição imediata  $T1$ , ao ser disparada, consome um token de *Clientes* e um token de *ReqChegou* e gera um token

no lugar *ProcRede1*. Os lugares *ProcRede1* e *ProcRede2* representam a infraestrutura de rede. O tempo da latência para cada requisição dentro do sistema é associado à transição exponencial *T<sub>Lat</sub>*, valor que pode variar de acordo com a quantidade de clientes (*QR*). O disparo da transição *T<sub>Lat</sub>* consome um token de *ProcRede1* e gera um token no lugar *ProcRede2*.

A transição imediata *T<sub>2</sub>*, ao ser disparada, consome um token de *ProcRede2* e também um token de *QMemoria*, gerando um token no lugar *UsoMemoria* que indica que a requisição já foi carregada na memória para poder ser executada. A seguir, o disparo da transição *T<sub>3</sub>* consome um token de *UsoMemoria* e outro de *QCores*, e deposita um token no lugar *Processar*, indicando que o processamento será iniciado. O tempo de processamento na CPU de cada requisição é associado à transição exponencial *T<sub>Processo</sub>*, e o disparo desta transição consome um token do lugar *Processar* e gera um token em *Processado*, que indica o término do processamento da requisição. Por fim, o disparo da transição imediata *T<sub>4</sub>* consome um token de *Processado* e libera os recursos consumidos (tokens) para *QCores*, *QMemoria* e *Cientes*. O label *QC*, associado ao lugar *QCores*, representa o número de Cores de CPU que o sistema em análise possui. Já o label *QM*, associado ao lugar *QMemoria*, representa a quantidade de memória RAM disponível para ser utilizada pelas requisições. As Tabelas 2 e 3 apresentam, respectivamente, um resumo das descrições dos lugares e das transições do modelo SPN proposto.

É importante destacar que, com esse modelo, é possível fazer a variação de alguns parâmetros, como, por exemplo, pode-se variar a quantidade de clientes *QR* no sistema e também o *T<sub>Lat</sub>*, que é o tempo de latência da rede e corresponde ao tempo necessário para que a carga de trabalho seja enviada pela rede ao sistema em análise. Essa latência pode variar de acordo com a quantidade de requisições, gerando impacto diretamente no tempo total de processamento. Outro parâmetro é o *T<sub>Processo</sub>*, que corresponde ao tempo de processamento das requisições na CPU, e o seu valor pode variar conforme a quantidade de clientes. Existe ainda o parâmetro *QC*, que representa a quantidade de Cores que a CPU possui. Por exemplo, o valor 2 significa que a VM faz uso de duas cores de CPUs. O último parâmetro é a quantidade de memória no sistema, sendo representado por *QM*. Por exemplo, o valor 4 pode representar que a memória RAM da VM possui 4 Gigabytes.

O modelo da Figura 12 faz uso de três métricas: (i) utilização da CPU, (ii) quantidade de memória RAM utilizada e, por fim, (iii) consumo de energia em Watts.

Tabela 2 – Descrição dos lugares do modelo.

Lugar	Descrição
Buffer (QR)	Quantidade de clientes para atendimento no sistema
ReqChegou	Quantidade esperando atendimento no sistema
Clientes (QR)	Quantidade de clientes sendo atendidos no sistema
ProcRede1	Requisições na rede antes de serem enviadas ao sistema
ProcRede2	Solicitações que foram enviadas ao sistema e se encontram esperando.
UsoMemoria	Requisições na memória aguardando serem escalonados para o processador.
Processar	Requisições em processamento pelo sistema.
Processado	Finalização do processamento.
QCpu (QC)	Quantidade de CPUs disponíveis para serem usados.
QMemoria (QM)	Quantidade de Memória RAM disponível para ser utilizada na VM.

Tabela 3 – Descrição das transições do modelo.

Transição	Tipo	Tempo	Tipo Serv.	Descrição
T_Chegada	Exponencial	-	Single Server	Chegada de 1 cliente
T1	Imediata	-	-	Chegada de 1 cliente
TLat	Exponencial	TLatencia	Single Server	Latência da rede.
T2	Imediata	-	-	Alocação de Memória
T3	Imediata	-	-	Alocação de Processador
TProcesso	Exponencial	TProc	Infinite Server	Tempo de processamento
T4	Imediata	-	-	Liberação dos recursos

Mais detalhes dessas métricas são apresentados na Seção 5.3.

## 5.2 Regressão Linear

Este trabalho faz uso de regressões lineares com o objetivo de propor equações e serão utilizadas no modelo SPN proposto. Por exemplo, os resultados das regressões podem ser utilizados como o *delay* de transições ou, ainda, na definição de métricas dentro do próprio modelo SPN.

### 5.2.1 Consumo de Energia e Nível de Utilização da CPU

A Figura 13 apresenta a regressão linear utilizada para definir a relação do consumo de energia com o nível de utilização da CPU para o Apache CloudStack. Nessa figura, o eixo Y representa o consumo de energia ( $W$ ) e o X nível de utilização da CPU ( $UC$ ). Os resultados dessas regressões serão utilizados e melhor ilustrados na Seção 5.3.

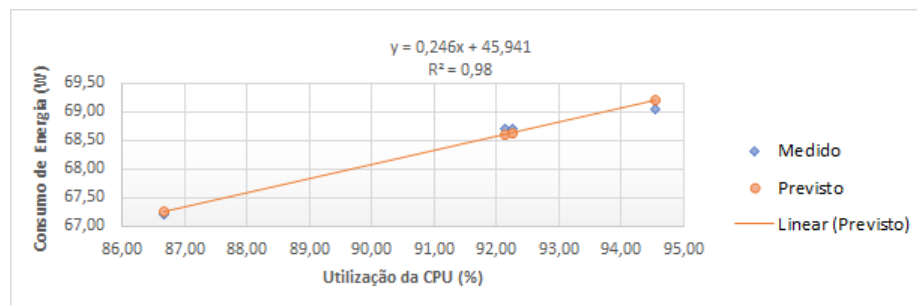


Figura 13 – Gráfico da Regressão linear - Consumo Energia Apache CloudStack

Ainda sobre a Figura 13, pode-se observar que a regressão praticamente se sobrepõe aos pontos. Vale destacar que a cor azul representa os dados consumidos de fato, e a cor vermelha representa a previsão obtida através da regressão. Cada um dos pontos consumidos representam o resultado da média de 30 medições realizadas no ambiente em laboratório. A qualidade da regressão pode ser comprovada pelo valor de  $R^2$  que foi 0,98 para o Apache CloudStack.  $R^2$  bem próximo ao valor um (1), indica a validade da regressão. A equação obtida pela regressão linear entre o consumo de energia e a utilização da CPU, na nuvem Apache CloudStack, é mostrada a seguir.

$$y = (0,246 \times UC) + 45,941 \quad (5.1)$$

Onde  $y$  é a previsão do consumo de energia em Watts e  $UC$  é a utilização da CPU. Assim, é possível estimar o consumo de energia do *controller*, que corresponde à máquina com a nuvem instalada. Essa equação é utilizada no consumo de energia do modelo de desempenho (Figura 12).

Já para o OpenStack, a Figura 14 apresenta a regressão linear utilizada para definir a relação do consumo de energia e a utilização da CPU para o OpenStack. Similarmente, o eixo Y representa o consumo de energia ( $W$ ) e o X representa a utilização da CPU ( $UC$ ). O resultado dessa regressão é mostrado na Equação 5.2.

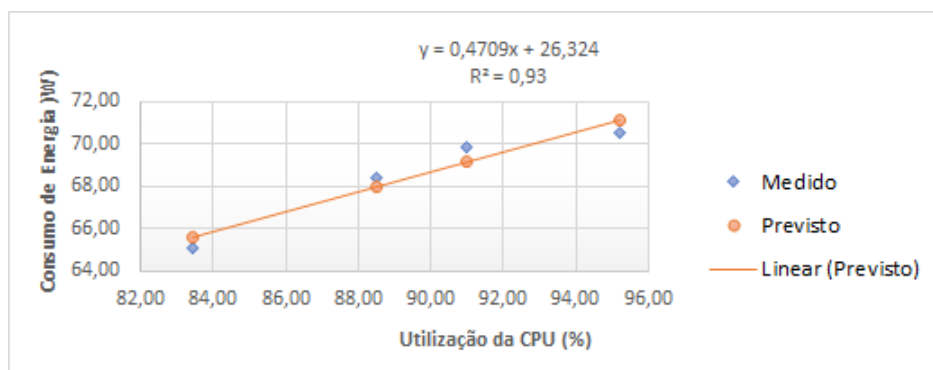


Figura 14 – Gráfico da Regressão linear - Consumo Energia OpenStack

$$y = (0,4709 \times UC) + 26,324 \quad (5.2)$$

Onde  $y$  é a previsão do Consumo de energia em Watts ( $W$ ) e  $UC$  é o nível de utilização da CPU por quantidade de usuários no sistema. Essa equação é utilizada no consumo de energia do modelo de desempenho (Figura 12) para a nuvem OpenStack.

Sobre a regressão ilustrada na Figura 14, pode-se observar que a regressão obteve um valor de  $R^2$  de 0,93, indicando assim a validade da regressão. É importante destacar que cada ponto da regressão representa um valor médio aferido no sistema para amostras de tamanho 30.

### 5.2.2 Regressão do tempo de resposta e quantidade de clientes

Outra regressão linear foi feita para relacionar o tempo de resposta com a quantidade de clientes fazendo requisições ao ambiente Moodle hospedado no Apache CloudStack e no



OpenStack. Esta regressão linear representa a relação entre a quantidade de clientes ( $x$ , independente) e o tempo de resposta para atender  $N$  requisições por cliente ( $y$ , dependente).

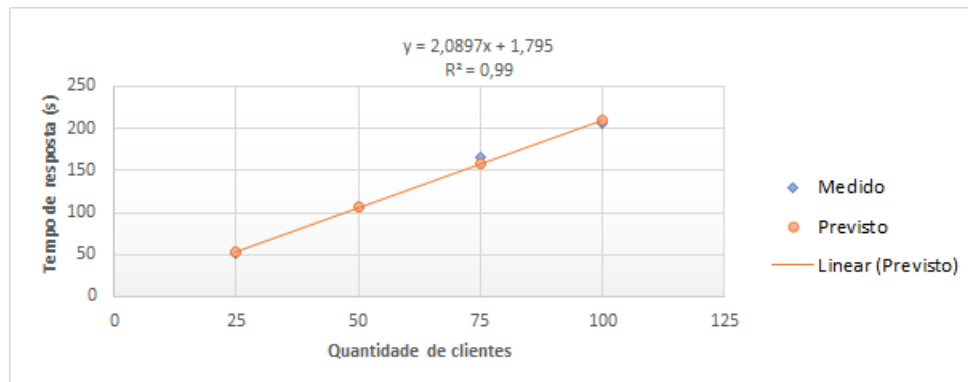


Figura 15 – Regressão linear - Tempo de resposta do Apache CloudStack

Os valores de  $y$  previstos (cor vermelha na Figura 15) pela regressão estão praticamente sobrepondo os valores de tempo de resposta obtidos pela medição. A qualidade da regressão pode ser confirmada com o valor computado para  $R^2$  que foi de 0,99. Essa regressão também fez uso de medições com amostras de tamanho 30. A regressão linear entre a quantidade clientes e o tempo de resposta, para o Apache CloudStack, é dada pela Equação 5.3.

$$y = (2,0897 \times QR) + 1,795 \quad (5.3)$$

Onde  $y$  é a previsão do tempo de resposta e  $QR$  é a quantidade de clientes fazendo solicitações no sistema.

Para o OpenStack, também foi desenvolvida uma regressão linear para relacionar o tempo de resposta com a quantidade de clientes fazendo requisições. Essa regressão linear representa a quantidade de clientes ( $x$ , independente) e o tempo de resposta para atender  $N$  requisições por cliente ( $y$ , dependente).

Similarmente, a Figura 16 mostra que os valores de  $y$  previstos pela regressão praticamente se sobrepõem aos valores do tempo de resposta aferidos experimentalmente. A qualidade da regressão pode ser confirmada com o valor computado para  $R^2$ , que foi de 0,998. Essa regressão linear entre a quantidade clientes e o tempo de resposta, para o

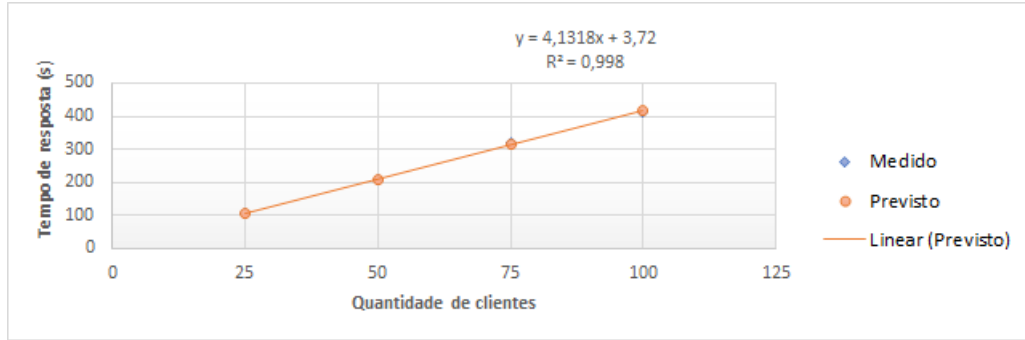


Figura 16 – Regressão linear - Tempo de resposta do OpenStack

OpenStack, é dada pela Equação 5.4.

$$y = (4,1318 \times QR) + 3,72 \quad (5.4)$$

Onde  $y$  é a previsão do tempo de resposta e  $QR$  é a quantidade de clientes fazendo solicitações no sistema.

### 5.2.3 Regressão do tempo de processamento e quantidade de clientes

Outra relação importante e que também foi analisada através de regressão linear é a do tempo de processamento com a quantidade de clientes fazendo requisições no sistema com Apache CloudStack e OpenStack. Esta regressão linear representa a quantidade de clientes ( $x$ , independente) e o tempo de processamento para atender  $N$  requisições por cliente ( $y$ , dependente).

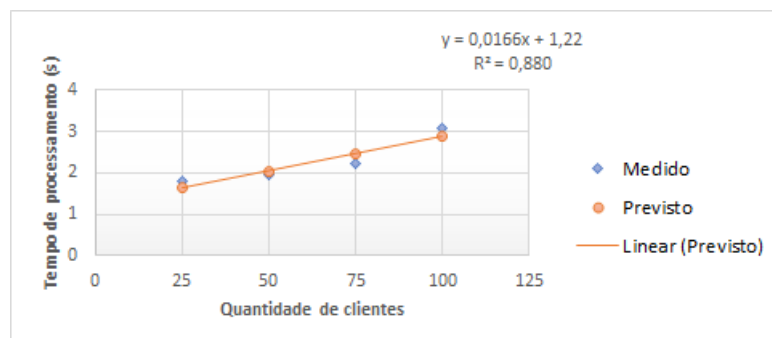


Figura 17 – Regressão linear - Tempo de processamento Apache CloudStack

A Figura 17 mostra que os valores para  $y$  previstos pela regressão são próximos aos valores obtidos via medição. Destacamos que cada ponto da medição representa a média de uma amostra com tamanho 30. A qualidade da regressão pode ser confirmada com o

valor computado para  $R^2$ , que foi de 0,880. Essa regressão para o Apache CloudStack é dada pela Equação 5.5.

$$y = (0,0166 \times QR) + 1,22 \quad (5.5)$$

Onde  $y$  é a previsão do tempo de resposta e  $QR$  é a quantidade de clientes fazendo solicitações no sistema.

Similarmente, foi desenvolvida uma regressão linear para o OpenStack. A Figura 18 mostra os valores de  $y$  previstos pela regressão para o OpenStack. A qualidade da regressão pode ser confirmada com o valor computado para  $R^2$ , que foi de 0,94. A regressão linear da quantidade clientes e do tempo de processamento, para o OpenStack, é dada pela Equação 5.6.

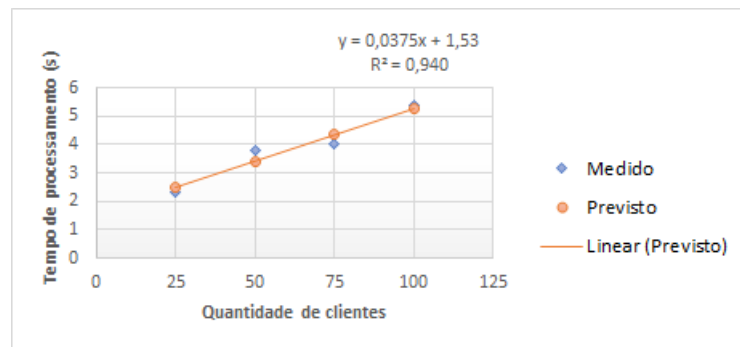


Figura 18 – Regressão linear - Tempo de processamento OpenStack

$$y = (0,0375 \times QR) + 1,53 \quad (5.6)$$

onde  $y$  é a previsão do tempo de resposta e  $QR$  é a quantidade de clientes fazendo solicitações no sistema.

#### 5.2.4 Tempo de latência e quantidade de clientes

Uma regressão linear foi realizada para relacionar o tempo de latência com a quantidade de clientes. Essa regressão linear representa a quantidade de clientes ( $x$ , independente) e o tempo de latência na rede para atender  $N$  requisições por cliente ( $y$ , dependente).

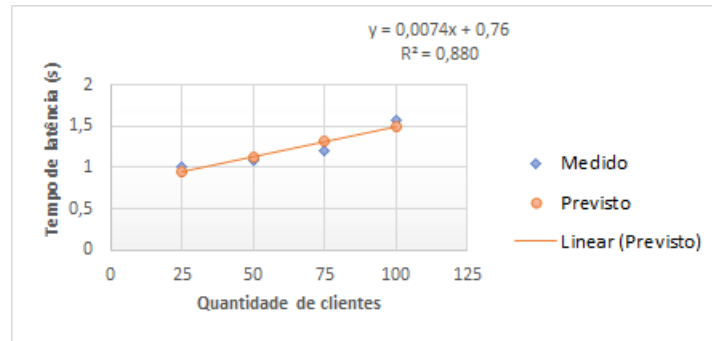


Figura 19 – Regressão linear - Tempo de latência Apache CloudStack

A Figura 19 apresenta os valores de  $y$  previstos pela regressão. Pode-se observar que tais valores praticamente sobrepõem os valores de tempo de resposta feito pela medição. O valor computado para  $R^2$ , foi de 0,88. Essa regressão linear entre a quantidade clientes e o tempo de processamento, para o Apache CloudStack, é dada pela Equação 5.7.

$$y = (0,0074 \times QR) + 0,76 \quad (5.7)$$

onde  $y$  é a previsão do tempo de latência e  $QR$  é a quantidade de clientes fazendo solicitações no sistema.

A mesma estratégia de regressão linear foi aplicada para relacionar o tempo de latência com a quantidade de clientes fazendo requisições para o OpenStack. A Figura 20 ilustra os resultados obtidos para a previsão de valores para  $y$ . O valor computado para  $R^2$ , foi de 0,91. A regressão linear entre a quantidade clientes e o tempo de processamento, para o OpenStack, é dada pela Equação 5.8.

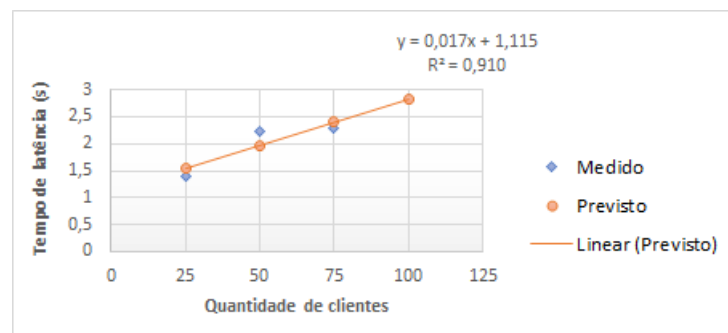


Figura 20 – Regressão linear - Tempo de latência OpenStack

$$y = (0,017 \times QR) + 1,115 \quad (5.8)$$

onde  $y$  é a previsão do tempo de latência e  $QR$  é a quantidade de clientes fazendo solicitações no sistema.

### 5.3 Métricas

Esta seção apresenta as métricas adotadas no modelo SPN proposto (Figura 12) e que são aplicadas para os dois sistemas de nuvem, Apache CloudStack e OpenStack.

#### 5.3.1 Utilização de CPU

A primeira métrica, *UtilizacaoCPU*, é responsável pelo cálculo da utilização da CPU na VM. A fórmula da utilização da CPU é definida pela Equação 5.9.

$$UtilizacaoCPU = \frac{(E\{\#Processar\})}{QC} \times 100 \quad (5.9)$$

onde  $E\{\#Processar\}$  indica o número médio de tokens no lugar *Processar* e  $QC$  corresponde a quantidade de CPUs.

#### 5.3.2 Utilização da Memória

A métrica do modelo *UtilizacaoMemoria* é responsável pelo cálculo da utilização da memória RAM na VM. A utilização da Memória é representada em % conforme definida na Equação 5.10.

$$UtilizacaoMemoria = \frac{QM - (E\{\#QMemoria\})}{QM} \times 100 \quad (5.10)$$

onde  $QM$  indica a quantidade de memória disponível na VM e  $E\{\#QMemoria\}$  corresponde ao número médio de tokens no lugar *QMemoria*.

#### 5.3.3 Consumo de Energia

Utilizando o modelo, a variável *ConsumoEnergia* é responsável por calcular o consumo de energia da máquina com a nuvem que hospeda o Moodle. Esse cálculo

consiste em verificar, através do ambiente de medição, o consumo elétrico pelo uso do *Controller* com instâncias ativas e com  $N$  clientes demandando solicitações à VM. As Equações 5.11 e 5.12 definem como é feito o cálculo do consumo de energia para o Apache CloudStack e OpenStack. É importante destacar que, em ambos os casos, foram utilizadas as regressões lineares mostradas anteriormente. Sendo assim, o consumo de energia do Apache CloudStack faz uso da Equação 5.1, e o CloudStack utiliza a regressão mostrada na Equação 5.2. Mais detalhes sobre essas regressões estão disponíveis na Seção 5.2. A regressão linear simples é dada pela seguinte Equação  $y = bx + a$ .

### Consumo de energia do Apache CloudStack

$$ConsumoEnergia = \left( 0,246 \times \frac{(E\{\#Processar\})}{QC} \times 100 \right) + 45,941 \quad (5.11)$$

onde  $b = 0,246$ ,  $E\{\#Processar\}$  indica o número médio de tokens no lugar *Processar* e  $QC$  é a quantidade de CPUs disponível e  $a = 45,941$ .

### Consumo de energia do OpenStack

$$ConsumoEnergia = \left( 0,4709 \times \frac{(E\{\#Processar\})}{QC} \times 100 \right) + 26,324 \quad (5.12)$$

onde  $b = 0,4709$ ,  $E\{\#Processar\}$  indica o número médio de tokens no lugar *Processar* e  $QC$  é a quantidade de CPUs disponível e  $a = 26,324$ .

#### 5.3.4 Vazão

A métrica do modelo *VZ* é responsável pelo cálculo da vazão do sistema. A vazão representa a quantidade de requisições que o sistema atende durante um período de tempo. A fórmula da vazão é definida nas Equações 5.13 e 5.14 para o OpenStack e Apache CloudStack, respectivamente.

#### Vazão OpenStack:

$$vz = \left( \frac{(E\{\#Processar\})}{(0,0375 \times QR) + 1,53} \right) \quad (5.13)$$

onde  $E\{\#Processar\}$  indica o número médio de tokens no lugar *Processar*. O tempo que leva para processar a requisição (*delay* da transição  $TProcesso$ ) foi substituído pela

Equação 5.6 da regressão linear, que indica o tempo que o sistema leva para atender as requisições.  $b = 0,0375$ ,  $QR$  Quantidade de usuários e  $a = 1,53$ .

**Vazão Apache CloudStack:**

$$vz = \left( \frac{(E\{\#Processar\})}{(0.0166 \times QR) + 1,22} \right) \quad (5.14)$$

onde  $E\{\#Processar\}$  indica o número médio de tokens no lugar *Processar*, e o denominador indica o tempo que o sistema leva para atender as requisições (Equação 5.5).  $b = 0,0166$ ,  $QR$  Quantidade de usuários e  $a = 1,22$ .

### 5.3.5 Tempo de Resposta

A métrica do modelo  $TR$  é responsável pelo cálculo do tempo de resposta que representa o período entre o envio da requisição para o sistema até a chegada da resposta para o cliente (BARROS et al., 2017). A fórmula do tempo de resposta é definida para o Apache CloudStack na Equação 5.15, e a Equação 5.16 define o mesmo tempo para o OpenStack.

**Tempo de resposta Apache CloudStack:**

$$TR = \left( \frac{(E\{\#ProcRede1\}) + (E\{\#ProcRede2\}) + (E\{\#UsoMemoria\}) + (E\{\#Processar\}) + (E\{\#Processado\})}{\left( \frac{(E\{\#Processar\})}{(0.0166 \times QR) + 1,22} \right)} \right) \quad (5.15)$$

**Tempo de resposta OpenStack:**

$$TR = \left( \frac{(E\{\#ProcRede1\}) + (E\{\#ProcRede2\}) + (E\{\#UsoMemoria\}) + (E\{\#Processar\}) + (E\{\#Processado\})}{\left( \frac{(E\{\#Processar\})}{(0,0375 \times QR) + 1,53} \right)} \right) \quad (5.16)$$

onde  $E\{\#\text{ProcRede1}\}$  indica o número médio de tokens no lugar ProcRede1,  $E\{\#\text{ProcRede2}\}$  indica o número médio de tokens no lugar ProcRede2,  $E\{\#\text{UsoMemoria}\}$  indica o número médio de tokens no lugar UsoMemoria,  $E\{\#\text{Processar}\}$  indica o número médio de tokens no lugar Processar,  $E\{\#\text{Processado}\}$  indica o número médio de tokens no lugar Processado. A equação  $(0.0166 \times QR) + 1,22$  corresponde à regressão linear do  $T\text{Processo}$  para o Apache CloudStack. A equação  $(0,0375 \times QR) + 1,53$  é a regressão linear do  $T\text{Processo}$  para o OpenStack.

#### 5.4 Considerações finais

Este capítulo apresentou o modelo SPN proposto para a quantificação do desempenho e do consumo de energia para o Moodle hospedado em dois sistemas de nuvem: Apache CloudStack e OpenStack. E a descrição do modelo, em que são apresentados os lugares e as descrições das transições. Em seguida, foram mostradas as regressões lineares para a quantificação da utilização da CPU, do consumo de energia e do tempo de resposta para cada sistema de nuvem, tempo de processamento, tempo de latência. Por último, foram apresentadas as métricas para utilização de CPU, utilização da memória RAM, vazão, tempo de resposta, para cada sistema de nuvem.



## 6 Estudos de Caso

Este capítulo apresenta seis estudos de caso que ilustram a aplicabilidade da estratégia proposta. O primeiro estudo faz o comparativo experimental do desempenho, utilização da CPU, utilização da memória RAM, vazão e tempo de resposta e também do consumo de energia entre o OpenStack e o Apache CloudStack. O segundo estudo de caso faz a validação do modelo SPN proposto, utilizando o nível de confiança de 95% e o erro relativo da simulação com 10%. Para a validação, foram adotadas as seguintes métricas: utilização da CPU, utilização da memória RAM, e por fim, o consumo de energia para os dois sistemas de nuvem. O estudo de caso 3 apresenta os resultados obtidos da vazão e do tempo de resposta através do modelo proposto. O estudo de caso 4 tem como objetivo analisar cenários que não foram possíveis de medir experimentalmente. Sendo assim, a variação da quantidade de CPUs ( $QC$ ) e da quantidade de memória RAM ( $QM$ ) são analisadas. O estudo de caso 5 tem como objetivo mostrar os resultados obtidos, via modelo, para a utilização da CPU e memória RAM com a variação do tempo de chegada e da quantidade de clientes ( $QR$ ). Finalmente, o estudo de caso 6 apresenta o comparativo do consumo de energia entre os dois sistemas de nuvem.

### 6.1 Estudo de Caso I

O objetivo deste estudo de caso é comparar o desempenho e o consumo de energia entre o OpenStack e o Apache CloudStack. Para isso, cenários considerando o diferente número de clientes no sistema (25, 50, 75 e 100) foram adotados, de modo a se fazer medições com a infraestrutura montada (ver Figura 11). Em todos os cenários, o Moodle foi acessado para a realização das seguintes atividades: acessar o site AVA, logar no ambiente Moodle, acessar o curso, acessar uma atividade, selecionar um arquivo de 1 MB, clique no arquivo de *upload*, *upload* do arquivo, salvar o envio e sair do AVA.

#### 6.1.1 Resultados

##### Consumo de energia

A Figura 21 exibe o resultado comparativo do consumo de energia nas duas nuvens privadas,

Apache CloudStack e OpenStack, para cada cenário analisado. Pode-se observar que o consumo de energia do OpenStack inicia inferior para o cenário com 25 clientes no sistema. No entanto, aumentando-se o número de clientes, percebe-se que o consumo de energia do Apache CloudStack passa a ser menor que o do OpenStack.

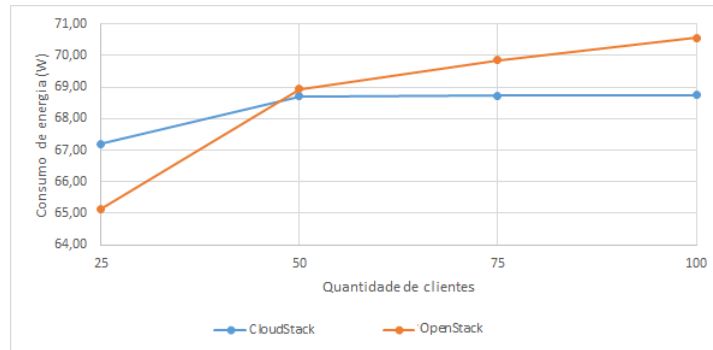


Figura 21 – Consumo de energia (Watts)

Para provar que a diferença no consumo de energia entre os dois ambientes de nuvem privada é significativo, aplicou-se o Teste T Pareado (MONTGOMERY; RUNGER, 2003). A Tabela 4 apresenta o resumo estatístico desse teste. Neste comparativo, o valor de  $P$  ficou em 0,001. Dessa forma, como o valor de  $P$  ficou inferior ao  $\alpha$  adotado (0,05), pode-se afirmar que as duas amostras são estatisticamente distintas, com 95% de confiança. Logo, o Apache CloudStack possui o consumo de energia diferente do OpenStack.

Tabela 4 – Teste-t: duas amostras em par para médias (W).

<b>Estatística</b>	<b>Resultado</b>
Quantidade (Número)	30
Desvio Padrão	1,008407188
Nível de Confiança	95%
Margem de Erro	0,361
Limite Inferior	-2,7457
Limite Superior	-1,2956
Valor de P	0,001

A Tabela 5 mostra, em maiores detalhes, os resultados da medição do consumo de

energia de cada computador com o sistema de nuvem Apache CloudStack e OpenStack. São apresentadas, ainda, as diferenças, em percentual, no consumo energético de cada sistema em nuvem, para cada cenário analisado. Além disso, pode-se perceber que o consumo médio do OpenStack (69,73W) é superior ao do Apache CloudStack (68,72W).

Tabela 5 – Comparação do consumo de energia - OpenStack vs Apache CloudStack.

QR	CloudStack(W)	OpenStack(W)	Diferença(%)
25	67,20	65,14	-3,16%
50	68,71	68,93	0,32%
75	68,72	69,85	1,62%
100	68,74	70,55	2,57%
Média	68,72	69,73	-
Desv.Pad	0,600	0,914	-

### Utilização da CPU

A Figura 22 mostra os resultados da utilização da CPU (ou seja, da VM) em cada nuvem privada. Pode-se observar que a utilização da CPU do Apache CloudStack inicia superior nos cenários com 25 e 50 clientes. No entanto, com o aumento dos clientes no sistema, observou-se que o nível de utilização passou a se inverter, mas com diferença mínima.

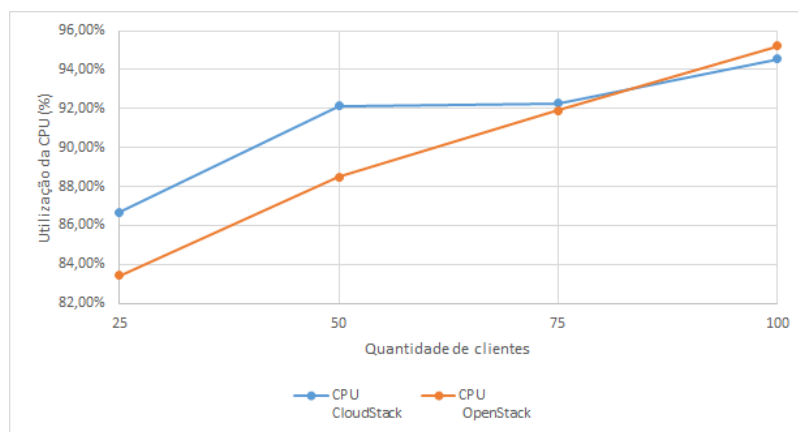


Figura 22 – Utilização da CPU (%)

Similarmente ao estudo do consumo de energia, também foi aplicado o Teste T Pareado. A Tabela 6 apresenta o resultado do resumo estatístico desse teste. Neste comparativo, o valor de  $P$  ficou em 0,001. Sendo assim, existe diferença significativa, com 95% de confiança, entre os resultados dos dois ambientes analisados.

Tabela 6 – Teste-t: duas amostras em par para médias (CPU).

<b>Estatística</b>	<b>Resultado</b>
Quantidade (Número)	30
Desvio Padrão	1,026774
Nível de Confiança	95%
Margem de Erro	0,010
Limite Inferior	-0,027
Limite Superior	-0,0071
Valor de P	0,001

A Tabela 7 mostra, em maiores detalhes, os resultados da medição do nível de utilização da CPU da VM, para cada cenário, considerando o Apache CloudStack e o OpenStack. Destaca-se que o nível de utilização médio da CPU obtido para o OpenStack foi superior.

Tabela 7 – Comparação do nível de utilização da CPU - OpenStack vs Apache CloudStack.

QR	Apache CloudStack	OpenStack	Diferença(%)
25	86,67%	83,45%	-3,86%
50	92,13%	88,49%	-4,11%
75	92,26%	91,91%	-0,38%
100	94,55%	95,21%	0,69%
Média	93,5%	95,2%	-
Desv.Pad	0,019	0,020	-

### Utilização da memória RAM

A Figura 23 mostra um gráfico comparativo da utilização de memória RAM para cada nuvem e contempla os cenários analisados. O OpenStack, nos cenários até 75 clientes, apresenta uma utilização de memória menor, quando comparado ao Apache CloudStack. No entanto, com 100 clientes, esse cenário se inverteu. O OpenStack passou a demandar mais memória RAM para realizar as mesmas atividades.

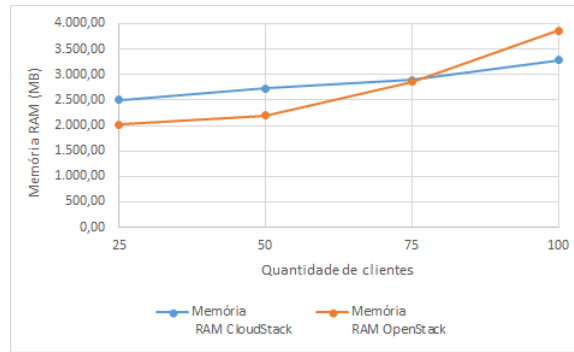


Figura 23 – Comparativo da utilização da memória RAM.

A Tabela 8 mostra, em maiores detalhes, os resultados da medição da utilização da memória RAM de cada nó com o sistema de nuvem Apache CloudStack e OpenStack. A tabela ressalta, ainda, o percentual da diferença da utilização da memória RAM das máquinas virtuais com OpenStack em relação com CloudStack. Todavia, pode-se perceber que, na média, para os cenários analisados, o Apache CloudStack teve uma utilização da memória RAM superior.

Tabela 8 – Comparação da utilização da memória RAM - OpenStack vs Apache CloudStack.

QR	CloudStack(MB)	OpenStack(MB)	Diferença(%)
25	2506,75	2015,23	-24,39%
50	2732,03	2197,91	-24,30%
75	2894,64	2855,32	-1,38%
100	3282,53	3860,89	14,98%
Média	2854,00	2732,30	-
Desv.Pad	326,98	834,39	-

Similar aos resultados anteriores, também foi utilizado o teste T Pareado. A Tabela 9 apresenta o resumo estatístico desse teste, onde percebe-se que as duas amostras são distintas, com 95% de confiança.

Tabela 9 – Teste-t: duas amostras em par para médias Memória RAM da VW.

<b>Estatística</b>	<b>Resultado</b>
Quantidade (Número)	30
Desvio Padrão	1,10782
Nível de Confiança	95%
Margem de Erro	0,039
Limite Inferior	-0,181
Limite Superior	-0,099
Valor de P	0,001

### Vazão

A Figura 24 mostra o resultado da comparação da vazão. Percebe-se que o Apache CloudStack apresenta uma vazão superior, em todos os cenários analisados, com uma média de 4.40 requisições atendidas por segundo. Já para o OpenStack, os resultados médios para a vazão foram de 2.76 requisições atendidas por segundo. A Tabela 10 mostra, em maiores detalhes, os resultados da medição da vazão de cada nó com o sistema de nuvem Apache CloudStack e OpenStack, apresentando o percentual da diferença na utilização da vazão das máquinas virtuais.

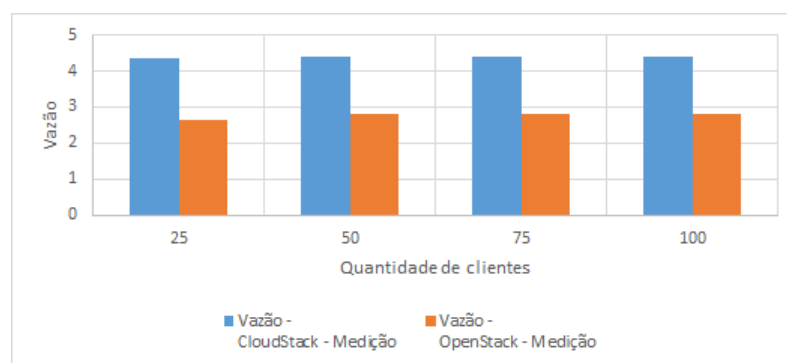


Figura 24 – Comparação da vazão

Assumindo o cenário com 25 clientes, o Apache CloudStack apresentou uma vazão bem superior em relação ao OpenStack (65%). É importante destacar que esse melhor desempenho do Apache CloudStack permaneceu para os demais cenários analisados. No

Tabela 10 – Comparação vazão - Apache CloudStack vs OpenStack.

QR	Vazão- CloudStack.	Vazão- OpenStack	Diferença(%)
25	4,37	2,65	-65,91%
50	4,40	2,79	-58,71%
75	4,40	2,80	-57,14%
100	4,41	2,81	-57,94%
Média	4,395	2,762	-
Desv.Pad	0,017	0,075	-

entanto, à medida que se aumenta a quantidade de clientes no sistema, o OpenStack obteve uma melhor vazão. No entanto, mesmo assim, permaneceu inferior em relação ao Apache CloudStack, passando para um desempenho da vazão inferior em 57,9%, assumindo o cenário com 100 clientes. Isso pode ser comprovado pelos dados mostrados na Tabela 10.

### Tempo de Resposta

A Figura 25 mostra o resultado da comparação do tempo de resposta das duas VMs. O Apache CloudStack apresentou sempre um tempo de resposta melhor do que o OpenStack, em todos os cenários analisados.

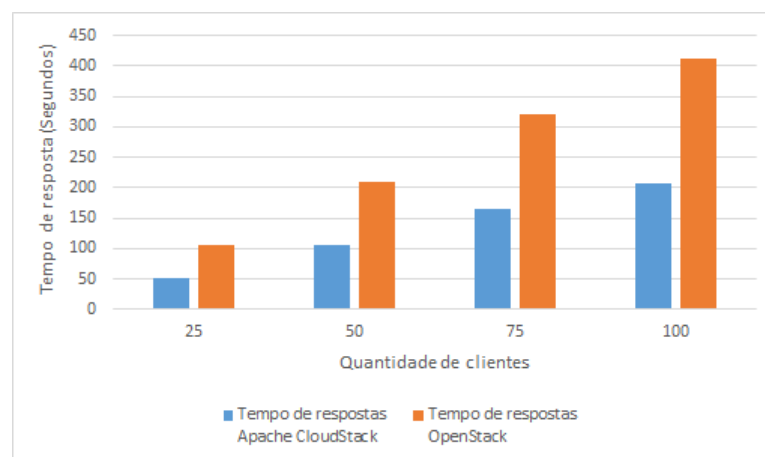


Figura 25 – Comparação do Tempo de resposta

A Tabela 11 apresenta com mais detalhes os resultados da medição obtidos para o tempo de resposta dos sistemas em nuvem Apache CloudStack e OpenStack. O Tempo de resposta da VM com OpenStack apresenta uma significativa diferença, para 25 clientes,

por exemplo, um percentual de 51,1% maior em relação ao mesmo cenário para o Apache CloudStack. Em resumo, o Apache CloudStack apresentou um tempo de resposta menor em todas as medições, como média de 132,40 segundos. Já o OpenStack teve um tempo médio de resposta de 261,95 segundos.

Tabela 11 – Comparação do tempo de resposta Apache CloudStack vs OpenStack.

QR	Apache CloudStack.	OpenStack	Diferença(%)
25	51,32	104,94	51,10%
50	106,86	210,34	49,20%
75	165,46	319,76	48,25%
100	205,95	412,78	50,11%
Média	132,40	261,95	-
Desv.Pad	67,64	133,43	-

De uma forma geral, é possível observar que, referente ao consumo de energia entre as duas nuvens, o Apache CloudStack obteve um menor consumo quando utilizamos a quantidade máxima de clientes simultaneamente no AVA. Já na utilização da CPU da VM para uma quantidade inicial de usuários, o OpenStack obteve melhor desempenho. À medida que expandimos a quantidade de clientes, o Apache CloudStack chega a uma utilização -1,78% menor. Para a Memória RAM, o OpenStack, nas primeiras quantidades de clientes, demonstrou uma menor utilização desse recurso. No entanto, à medida em que se aumentou a quantidade de requisições, o Apache CloudStack obteve 4,45% a mais de utilização da memória RAM, na média das medições. Já em relação à vazão, foi revelado que o Apache CloudStack tem a vazão média de 59,06% maior quando comparado com o OpenStack. Por fim, o tempo de resposta do Apache CloudStack foi de -49,45%, evidenciando um melhor desempenho.

## 6.2 Estudo de Caso II

Esta seção apresenta o segundo estudo de caso, que tem como objetivo validar o modelo SPN proposto na Seção 5.1. Para isso, foram coletados dados via medições da utilização da CPU e memória RAM, e do consumo de energia nos dois sistemas de nuvem avaliados neste trabalho. Conforme explicado anteriormente, a coleta foi realizada



utilizando-se *softwares* (JMeter, NMON) e *scripts* de medição no *hardware* do ambiente montado em laboratório.

Para cada simulação, utilizamos o modelo SPN com os parâmetros utilizados nas medições para cada sistema de nuvem (Apache CloudStack e OpenStack). Na avaliação do modelo SPN, foi adotada simulação estacionária, com o nível de confiança de 95% e com o erro relativo da simulação 10%.

Tabela 12 – Parâmetros de entrada do modelo.

	Apache CloudStack		OpenStack	
QR	T <sub>Lat</sub> (s)	T <sub>processo</sub> (s)	T <sub>Lat</sub> (s)	T <sub>processo</sub> (s)
25	1,01	1,78	1,41	2,33
50	1,08	1,95	2,22	3,78
75	1,21	2,23	2,28	4,01
100	1,58	3,07	2,81	5,38

A Tabela 12 apresenta os parâmetros de entrada para o modelo proposto, que representa o comportamento do Apache CloudStack e do OpenStack. Em ambos os casos, foram adotados 2 CPUs ( $QC=2$ ), 4GB de memória RAM ( $QM=4$ ) e o tempo de chegada entre um cliente e outro ao sistema de 0,1 segundos ( $TChegada=0,1$ ).

### 6.2.1 Resultados

#### Utilização da CPU

A Figura 26 apresenta um comparativo dos resultados da medição e dos obtidos via simulação do modelo proposto para a utilização da CPU em ambos os ambientes, Apache CloudStack e OpenStack.

A Tabela 13 mostra, em maiores detalhes, os resultados da medição e da modelagem e seus respectivos intervalos de confiança, para o nível de utilização da CPU da VM com Apache CloudStack. Todas as medições foram executadas, no mínimo, 30 vezes. Para validação do modelo, pode-se perceber que os intervalos de confiança dos dados medidos se sobrepõem aos valores estimados pelo modelo, conforme pode ser visto na Tabela 13.

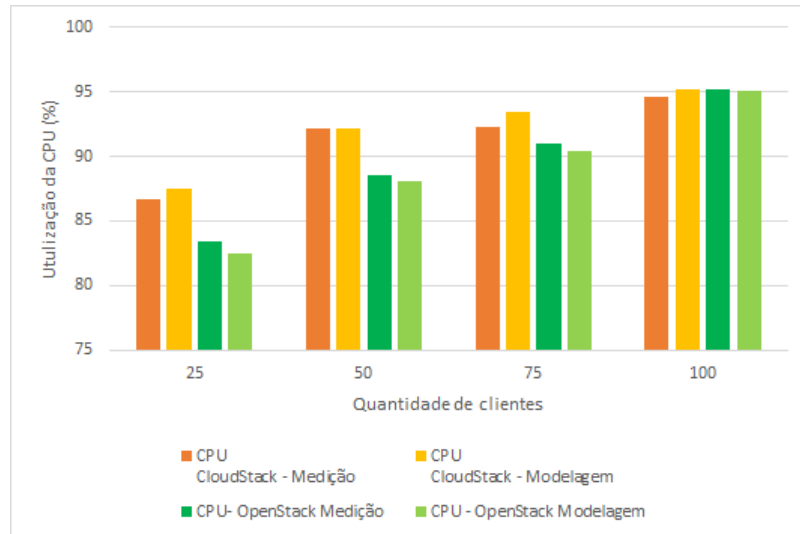


Figura 26 – Comparativo da medição e modelagem - nível de utilização da CPU.

Por exemplo, assumindo 25 clientes, o intervalo de confiança medido foi [83,03; 90,30], e o obtido com o modelo foi [85,39; 88,45]. Sendo assim, o valor estimado pelo modelo se encontra dentro do quantificado no sistema com a aferição.

Tabela 13 – Resultados para validação do modelo com o nível de utilização da CPU da VM com Apache CloudStack

QR	Medição		Modelagem	
	CPU	Int. Conf.	CPU	Int. Conf.
25	86,67	[83,03; 90,30]	87,46	[85,39; 88,45]
50	92,13	[91,49; 92,17]	92,11	[90,60; 93,78]
75	92,26	[90,66; 93,66]	93,40	[90,47; 93,57]
100	94,55	[93,84; 95,26]	95,18	[89,88; 96,33]

Similarmente, a Tabela 14 mostra os resultados para o nível de utilização da CPU da VM com OpenStack. A tabela também mostra que os intervalos de confiança dos valores medidos se sobrepõem aos valores dos dados estimados pelo modelo.

### Utilização da memória RAM

A Figura 27 apresenta um comparativo dos resultados entre a medição e os dados estimados pela modelagem da utilização da memória RAM tanto no sistema Apache CloudStack como no OpenStack. A Tabela 15 mostra, em maiores detalhes, os resultados da medição e do modelo proposto. É possível perceber que os intervalos de confiança se sobrepõem, ou seja, também foi possível realizar a validação para este cenário da utilização da memória RAM no Apache CloudStack.

Tabela 14 – Resultado para validação do modelo com uso da CPU da VM - OpenStack

QR	Medição		Modelagem	
	CPU	Int. Conf.	CPU	Int. Conf.
25	83,45	[81,02; 85,87]	82,51	[81,69; 83,52]
50	88,49	[87,24; 89,74]	88,10	[87,76; 88,45]
75	91,01	[89,15; 92,92]	90,37	[88,74; 92,00]
100	95,21	[94,48; 95,95]	95,08	[92,62; 97,33]

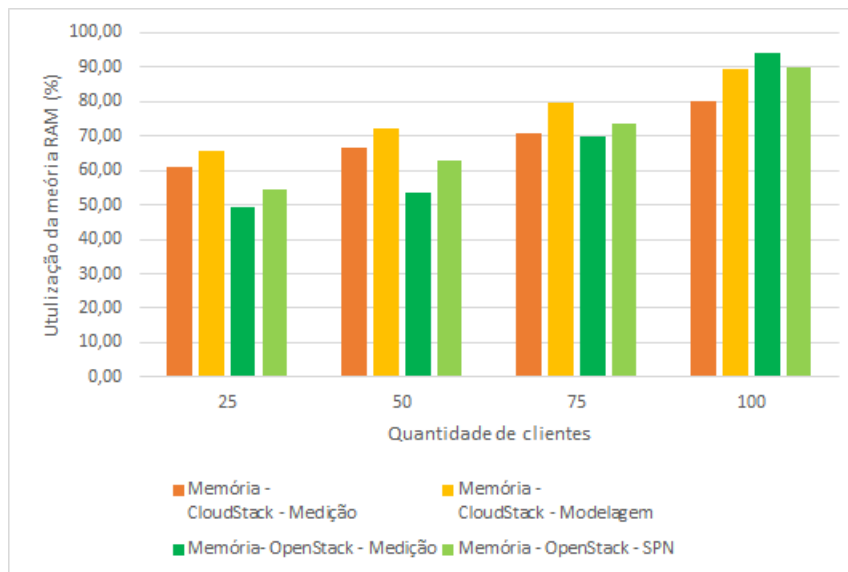


Figura 27 – Resultado da medição e modelagem RAM - Apache CloudStack e OpenStack

Tabela 15 – Resultado para validação do modelo com utilização da memória RAM - Apache CloudStack

QR	Medição		Modelagem	
	M. RAM(%)	Int. Conf.	M. RAM(%)	Int. Conf.
25	61,20	[57,62; 64,78]	65,46	[62,42; 71,38]
50	66,70	[64,11; 67,34]	72,04	[65,93; 76,52]
75	70,67	[68,48; 72,91]	79,71	[71,77; 79,73]
100	80,14	[76,16; 84,11]	89,52	[80,89; 89,74]

O mesmo também foi feito para os sistemas que fazem uso do OpenStack. Conforme pode ser visto também na Figura 27, os resultados obtidos para medição e modelagem da utilização da memória RAM para o sistema com OpenStack são próximos. A Tabela 16 mostra, em maiores detalhes, tais resultados. Como os intervalos de confiança dos dados medidos e estimados pelo modelo se sobrepõem, pode-se afirmar que os dados obtidos pelo modelo representam o sistema também para o cenário de utilização da memória RAM.

Tabela 16 – Resultado para validação do modelo com a utilização da memória RAM - OpenStack

Medição			Modelagem	
QR	M. RAM(%)	Int. Conf.	M. RAM(%)	Int. Conf.
25	49,20	[45,53; 52,15]	54,44	[50.86; 56.97]
50	53,66	[52,28; 59.03]	63,01	[58.97; 64.92]
75	69,71	[66,70; 71,63]	73,45	[70.65; 75.93]
100	94,26	[90,21; 94,50]	90,03	[87.03; 91.16]

### Consumo de energia

Os resultados comparativos das duas nuvens privadas para o consumo de energia entre os dados obtidos via modelo e os auferidos no ambiente real podem ser vistos na Figura 28. A Tabela 17 mostra, em maiores detalhes, os resultados da medição e da modelagem e seus respectivos intervalos de confiança para o CloudStack. A validação do modelo ocorre quando os intervalos de confiança se sobrepõem, como acontece na Tabela 17. Similarmente, a Tabela 18 mostra os resultados para o OpenStack.

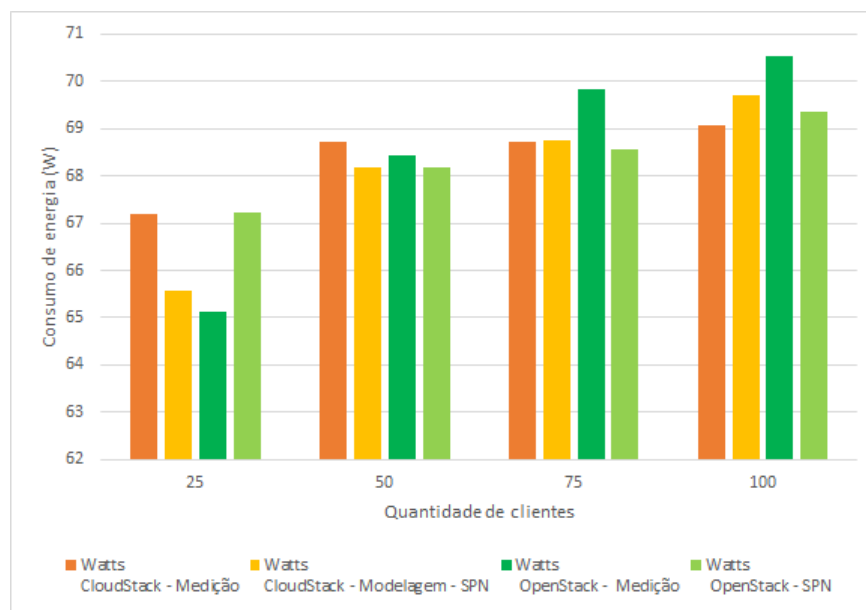


Figura 28 – Resultado da medição e modelagem Watts - Apache CloudStack e OpenStack

Sendo assim, pode-se destacar que esse estudo de caso comprovou que o modelo SPN proposto representa o sistema analisado quanto ao consumo de energia, utilização de

Tabela 17 – Resultado para validação do modelo com o consumo de energia (W) - Apache CloudStack

QR	Medição		Modelagem	
	Cons. E. (W)	Int. Conf.	Cons. E. (W)	Int. Conf.
25	67,20	[57,62; 67,78]	65,58	[65,52; 67,87]
50	68,71	[68,12; 68,74]	68,19	[67,38; 68,99]
75	68,72	[68,53; 68,95]	68,76	[66,53; 68,80]
100	69,06	[68,32; 69,06]	69,70	[69,02; 70,20]

Tabela 18 – Resultado para validação do modelo com o consumo em de energia (W) - OpenStack

QR	Medição		Modelagem	
	Cons. E. (W)	Int. Conf.	Cons. E. (W)	Int. Conf.
25	65,14	[60,86; 67,28]	67,22	[64,08; 67,36]
50	68,43	[66,80; 69,72]	68,18	[68,12; 68,84]
75	69,85	[65,64; 70,98]	68,57	[68,29; 68,85]
100	70,55	[69,35; 71,16]	69,36	[68,96; 69,77]

CPU e Memória RAM.

### 6.3 Estudo de Caso III

Esta seção apresenta o terceiro estudo de caso, cujo objetivo é apresentar os resultados obtidos para a vazão e o tempo de resposta utilizando o modelo SPN proposto (ver Seção 5.1). Sendo assim, comparações do tempo de resposta e da vazão obtidas pelo modelo foram utilizadas para comparar o desempenho dos dois ambientes, Apache CloudStack e OpenStack.

Para cada simulação, utilizamos o modelo SPN com os parâmetros para cada sistema de nuvem. A configuração utilizada na máquina virtual é a mesma do estudo de caso anterior. Esse estudo de caso adota os seguintes parâmetros de entrada para o modelo proposto representando o Apache CloudStack e o OpenStack (ver Tabela 12). Em ambos os casos, foram adotados 2 CPUs ( $QC=2$ ), 4 GB de memória RAM ( $QM=4$ ) e o tempo de chegada entre um cliente e outro ao sistema foi de 0.1 segundos ( $TChogada=0.1$ ).

### 6.3.1 Resultados

#### Vazão

A Figura 29 mostra um gráfico comparativo dos resultados para a vazão entre o Apache CloudStack e OpenStack que foram obtidos via simulação do modelo proposto. Nessa figura, é possível observar que a vazão do Apache CloudStack apresenta um desempenho superior. Além disso, é possível observar que a vazão de cada um desses sistemas permaneceu praticamente constante. Por exemplo, assumindo o CloudStack, pode-se dizer que a vazão foi praticamente 0,70 requisições (ou clientes) atendidos por segundo, ao passo que para o OpenStack foi de 0,42 requisições por segundo. Logo, estes resultados mostram que o Apache CloudStack possui uma vazão superior quando comparado com o outro sistema em estudo. É importante destacar, ainda, que esse comportamento da vazão praticamente constante ocorreu em virtude do limite do sistema. No entanto, pode-se perceber, no tempo de resposta mostrado a seguir, que com mais clientes, o tempo demandado para fazer as mesmas operações foi bem superior.

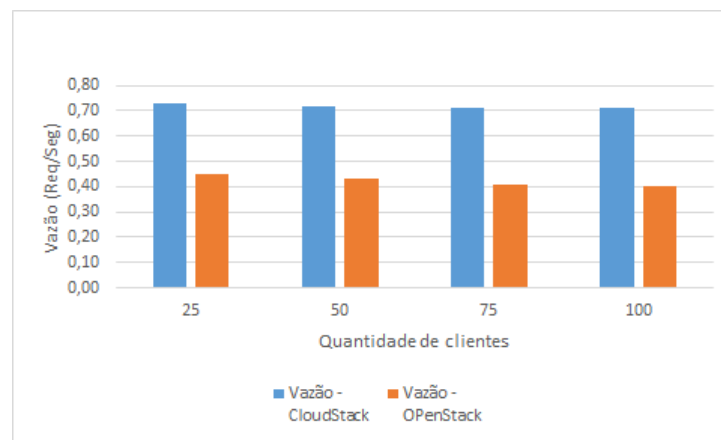


Figura 29 – Resultado da simulação da vazão - Apache CloudStack e OpenStack

#### Tempo de resposta

A Figura 30 mostra o resultado da simulação e do tempo de resposta para os sistemas Apache CloudStack e OpenStack utilizando o modelo proposto. Conforme esperado, o tempo de resposta no sistema acaba aumentando à medida que se adicionam mais clientes no sistema. A partir desse gráfico, é possível perceber também que o sistema OpenStack apresenta um tempo de resposta significativamente maior quando comparado com o tempo de resposta do CloudStack. Esse comportamento se deve pelo fato de que o OpenStack possui uma vazão média bem inferior, conforme mostrado anteriormente na Figura 29.

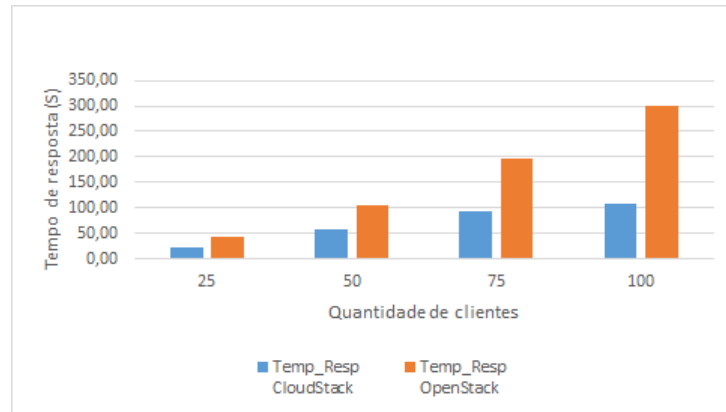


Figura 30 – Resultado da simulação do tempo de resposta - Apache CloudStack e OpenStack

#### 6.4 Estudo de Caso IV

Esta seção apresenta o quarto estudo de caso, que tem como objetivo mostrar o comportamento do desempenho dos sistemas ao serem realizadas variações da quantidade de recursos presentes para a CPU e memória RAM. As simulações foram realizadas a partir do modelo SPN da Figura 12.

Vários cenários foram definidos com a variação do número de CPUs ( $QC$ ) e da quantidade de memória RAM disponível no sistema ( $QM$ ). A Tabela 19 mostra os cenários que foram adotados. Por exemplo, com  $QC=2$  foram realizados experimentos com a memória variando de 4, 8, 16, 32 e 64 GB. Além desses cenários, também foi variada a quantidade de CPUs, que assumiram os valores  $QC=2, 4, 6, e 8$ .

Tabela 19 – Parâmetros de variação da CPU ( $QC$ ) e RAM ( $QM$ ).

$QC$	$QM$
2	4 8 16 32 64
4	4 8 16 32 64
6	4 8 16 32 64
8	4 8 16 32 64

Esse estudo de caso adotou os seguintes parâmetros de entrada para o modelo proposto representando o Apache CloudStack e o OpenStack (ver Tabela 12). O tempo de chegada entre um cliente e outro ao sistema foi de 0.1 segundos ( $T_{Chegada}=0.1$ ) e a

quantidade de clientes ( $QR = 100$ ).

#### 6.4.1 Resultados

##### Apache CloudStack

A Tabela 20 apresenta os resultados onde foram adotados 2 CPUs ( $QC=2$ ) variando a quantidade de memória RAM ( $QM$ ). É importante lembrar que foi utilizado o  $QR=100$ . Nesses resultados, ficou evidenciado que a utilização da CPU não sofreu impacto significativo. Porém, à medida que foi disponibilizado mais recurso de memória ( $QM$ ), ocorreu uma redução do percentual da memória RAM utilizada.

Tabela 20 – Resultados da Simulação utilizado  $QC=2$  - Apache CloudStack.

QM	CPU (%)	Memória RAM (%)
4	95,18 %	89,52 %
8	97,43 %	89,94 %
16	96,90 %	78,23 %
32	97,26 %	65,94 %
64	96,98 %	42,68 %

Neste segundo bloco de experimentos, foram adotados 4 CPUs ( $QC=4$ ), fazendo a variação da memória RAM ( $QM$ ), conforme mostrado na Tabela 19. A Tabela 21 apresenta os resultados das simulações utilizando  $QR=100$  e  $QC=4$ , e para estes parâmetros ficou evidenciado que a utilização da CPU sofreu impacto quando comparado com os resultados obtidos na Tabela 20. Esta mesma observação foi evidenciada também na medida em que foi disponibilizado mais recurso de memória RAM, ocorrendo uma redução do percentual da utilização da memória RAM.

Neste terceiro resultado, foram adotados 6 CPUs ( $QC=6$ ), novamente fazendo a variação da memória RAM ( $QM$ ). A Tabela 22 apresenta os resultados das simulações ainda fixando o  $QR=100$ . Para estes parâmetros, ficou evidenciado que a utilização da CPU sofreu impacto quando comparado com o apresentado na Tabela 21. Esta mesma observação foi evidenciada também na medida em que foi disponibilizado mais recurso para memória RAM, onde se observou uma redução do nível de utilização da memória.



Tabela 21 – Resultados da Simulação utilizado QC=4 - Apache CloudStack.

QM	CPU (%)	Memória RAM (%)
4	48,52%	48,52 %
8	48,37%	25,99 %
16	48,56%	13,10 %
32	48,63%	6,55 %
64	48,68%	3,28 %

Tabela 22 – Resultados da Simulação utilizado QC=6 - Apache CloudStack.

QM	CPU (%)	Memória RAM (%)
4	32,28%	48,42%
8	32,18%	24,21%
16	32,40%	13,48%
32	32,45%	6,10%
64	32,41%	3,05%

No cenário assumindo que foram adotados 8 CPUs ( $QC=8$ ), fazendo a variação da memória RAM ( $QM$ ), e com o  $QR$  fixado em 100 clientes, ficou evidenciado que a utilização da CPU reduziu ainda mais. Esta mesma observação, foi evidenciada também na medida em que foi disponibilizado mais recurso de memória RAM, onde acabou acontecendo uma redução no percentual de utilização da memória. A Tabela 23 ilustra tais resultados obtidos.

Tabela 23 – Resultados da Simulação utilizado QC=8 - Apache CloudStack.

QM	CPU (%)	Memória RAM (%)
4	24,36%	48,73%
8	24,07%	24,07%
16	24,81%	12,40%
32	24,18%	6,04%
64	24,10%	3,01%

## OpenStack

Similarmente, os mesmos cenários foram analisados considerando o OpenStack. Neste primeiro cenário, foram adotadas 2 CPUs ( $QC=2$ ), variando a quantidade de memória RAM ( $QM$ ). A Tabela 24 apresenta os resultados obtidos. Pode-se perceber que a utilização

da CPU com  $QM=64$  sofreu uma pequena variação para baixo quando comparado com  $QM=4$  impacto, porém, à medida que foi disponibilizado mais recurso de memória ( $QM$ ), houve uma redução da utilização significativa na memória RAM.

Tabela 24 – Resultados da Simulação utilizado  $QC=2$  - OpenStack.

QM	CPU (%)	Memória RAM (%)
4	96,35 %	91,75 %
8	95,32 %	83,77 %
16	95,41 %	71,96 %
32	95,21 %	52,00 %
64	95,38 %	31,42 %

Agora, assumindo 4 CPUs ( $QC=4$ ) e variando a quantidade de memória RAM ( $QM$ ), temos como resultados (ver Tabela 25) que a utilização da CPU sofreu impacto quando comparado com a apresentada na Tabela 24. Esta mesma observação foi evidenciada também na medida em que foi disponibilizado mais recurso  $QM$ , onde ocorreu uma redução da utilização da memória RAM.

Tabela 25 – Resultados da Simulação utilizado  $QC=4$  - OpenStack.

QM	CPU (%)	Memória RAM (%)
4	47,53%	48,43 %
8	47,62%	24,41 %
16	47,72%	12,85 %
32	47,85%	6,48 %
64	47,84%	3,18 %

O processo se repete com a utilização de 6 CPUs ( $QC=6$ ), repetindo a mesma variação da memória RAM ( $QM$ ). A Tabela 26 apresenta os resultados, onde ficou evidenciado que a utilização da CPU sofreu impacto quando comparado com os resultados apresentados na Tabela 25. A utilização da memória também sofreu redução à medida em que mais memória foi considerada.

Por fim, considerando agora 8 CPUs ( $QC=8$ ) e variando a quantidade de memória RAM ( $QM$ ) disponível, foi possível obter os dados conforme mostra a Tabela 27. Ficou evidenciado que a utilização da CPU sofreu impacto na redução quando comparado com

Tabela 26 – Resultados da Simulação utilizado  $QC=6$  - OpenStack.

QM	CPU (%)	Memória RAM (%)
4	32,10%	48,23%
8	32,20%	24,24%
16	32,00%	12,00%
32	32,12%	6,05%
64	32,05%	3,03%

todas as simulações anteriores. Esta mesma observação foi evidenciada também na medida em que foi disponibilizado mais recurso de memória  $QM$ , onde uma redução do nível de utilização foi observada.

Tabela 27 – Resultados da Simulação utilizado  $QC=8$  - OpenStack.

QM	CPU (%)	Memória RAM (%)
4	23,79,%	47,50%
8	23,80%	23,80%
16	23,81%	11,97%
32	23,80%	5,93%
64	24,10%	2,95%

### Comparando os Resultados do OpenStack vs Apache CloudStack

A Figura 31 mostra uma comparação da quantidade de CPUs utilizadas e o seu impacto no nível de utilização da CPU tanto no OpenStack como no Apache CloudStack. Pode-se perceber que o aumento da quantidade da CPU disponível, nos dois sistemas em nuvem, ocasionou uma redução significativa na utilização média das CPUs. Por exemplo, considerando o cenário com 2 CPUs ( $QC = 2$ ) e 4GB de memória RAM ( $QM=4$ ), em comparação com o de 8 CPUs, tivemos um resultado para o OpenStack saindo de 96% para 23% de utilização e, assumindo o CloudStack, a redução foi de 95% para 24%. É importante ressaltar que a variação da memória RAM e o impacto no nível de utilização da CPU não foram tão representativos. Sendo assim, os valores acabaram se sobrepondo a este gráfico da Figura 31.

Outro fator que também é relevante: a média do nível de utilização das CPUs considerando o cenário com 2 CPUs disponíveis, para o Apache CloudStack, foi de 96,75%. Esse valor médio leva em consideração os experimentos com diferentes quantidades de

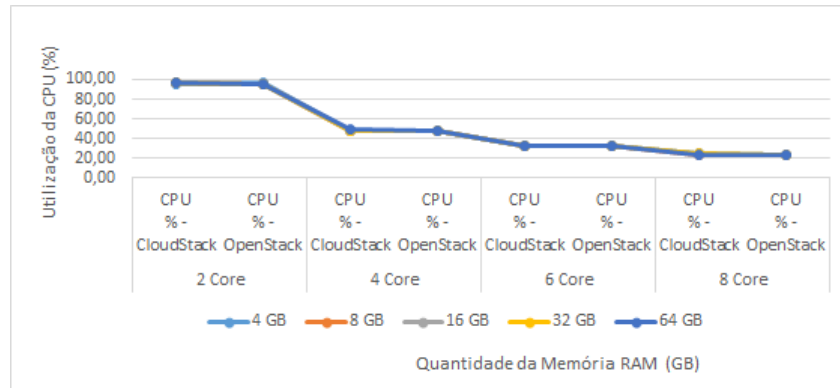


Figura 31 – Resultado da simulação da utilização da CPU - Apache CloudStack e OpenStack

memória RAM disponíveis (4, 8, 16, 32 e 64GB). Já no cenário com 8 CPUs disponíveis, a utilização da CPU foi de 24,30%. Assumindo ainda o OpenStack, o valor médio de utilização da CPU para 2 CPUs foi de 95,93%. Com 8 CPUs a utilização da CPU foi de 23,86% e de 18,43% para utilização da memória RAM.

A Figura 32 mostra o resultado da comparação na simulação da utilização da memória RAM ( $QM$ ), e o aumento da quantidade da CPU ( $QC$ ) no OpenStack e CloudStack. É possível perceber que o aumento da quantidade de CPUs disponíveis ocasionou uma redução na utilização da memória RAM apenas no aumento de 2 para 4 cores de CPU. Nas demais situações, o aumento da quantidade de cores no sistema acabou não reduzindo o percentual de utilização da memória RAM. Ainda neste gráfico, é possível destacar que à medida que se aumenta a quantidade de memória RAM, nos dois sistemas em nuvem, o nível de utilização da memória RAM teve uma redução.

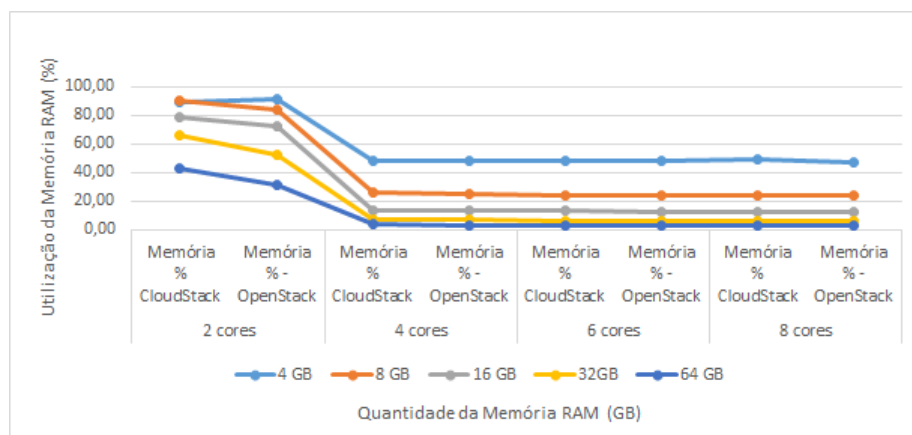


Figura 32 – Resultado da simulação da utilização da Memória RAM - Apache CloudStack e OpenStack

É importante evidenciar que a média da utilização da Memória RAM com 2 CPUs

para o Apache CloudStack foi de 73,26% e a média da utilização da CPU para 8 CPUs foi de 24,30%. Esse valor médio se refere aos valores com o número de cores fixo, variando somente a memória RAM (4, 8, 16, 32 e 64GB). Já com o OpenStack assumindo 8 CPUs, a média foi de 18,43% da memória RAM. Para o OpenStack, a média de utilização da memória RAM com 2 CPUs foi de 66,18% e a média para 8 CPUs foi de 18,43%.

Com base nos dados apresentados, pode-se concluir que o sistema OpenStack acabou apresentando um menor nível de utilização tanto para a memória RAM quanto para a CPU. Vale destacar que isso se deu para os cenários analisados.

## 6.5 Estudo de Caso V

Esta seção apresenta o estudo de caso V, que tem como objetivo mostrar os resultados obtidos para a utilização da CPU e memória RAM com variação do tempo de chegada de clientes ao sistema. Para isso, foi utilizado o modelo SPN de desempenho proposto, onde foram feitas comparações aplicando variações no tempo de chegada dos clientes.

A Tabela 28 mostra os cenários que são adotados neste experimento. Por exemplo, foram feitos experimentos com 25, 50, 75 e 100 clientes ao mesmo tempo realizando requisições ao sistema. Para cada um desses cenários com  $n$  clientes, adotaram-se tempos de chegada diferentes (0,01, 0,05, 0,1, 0,2, 0,3 0,4, 0,5, 1,0 s). Esses experimentos foram realizados tanto no Apache CloudStack, como no OpenStack. Para cada simulação, utilizamos o modelo SPN com os parâmetros de latência e tempo de processamento para cada sistema de nuvem, conforme mostrado na Tabela 12.

Tabela 28 – Parâmetros de variação do Tempo de Chegada.

QR	T_Chegada(s)								
25	0,01	0,05	0,1	0,2	0,3	0,4	0,5	1,0	
50	0,01	0,05	0,1	0,2	0,3	0,4	0,5	1,0	
75	0,01	0,05	0,1	0,2	0,3	0,4	0,5	1,0	
100	0,01	0,05	0,1	0,2	0,3	0,4	0,5	1,0	

### 6.5.1 Resultados

#### Apache CloudStack

A Tabela 29 apresenta os resultados da simulação no sistema Apache CloudStack, em que foi evidenciado que a variável Tempo de Chegada ( $T1$ ) gera impacto na utilização da CPU. Ao realizar a comparação da média de utilização da CPU em relação com o tempo de chegada, pode-se visualizar que, com um tempo  $T1=0,01s$ , obtém-se uma média de utilização da CPU em 91,91%. Já assumindo um  $T1=1,0s$ , tem-se a utilização da CPU média em 90,43%.

Tabela 29 – Utilização da CPU (%) Apache CloudStack.

T1 (s)	25 Clientes	50 Clientes	75 Clientes	100 Clientes	Média
0,01	88,00	89,79	92,91	96,92	91,91
0,05	87,76	89,87	92,71	96,98	91,83
0,1	87,65	89,80	92,55	96,25	91,56
0,2	87,74	89,70	91,91	96,21	91,39
0,3	87,72	89,66	91,35	95,78	91,13
0,4	87,51	89,64	91,29	95,66	91,03
0,5	87,43	89,55	91,17	95,64	90,95
1,0	86,24	89,07	90,55	95,86	90,43
Média	87,51	89,64	91,81	96,16	91,28

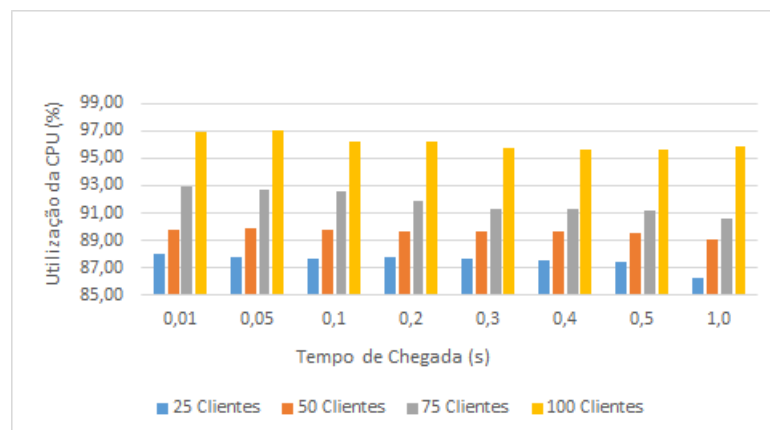


Figura 33 – Resultado da Utilização da CPU com variação do T1 - Apache CloudStack

A Figura 33 mostra a variação da utilização da CPU, no sistema Apache CloudStack, quando temos um aumento do tempo de chegada de clientes ao sistema. Pode-se perceber

que o aumento da frequência de chegada de clientes ao sistema, mesmo limitando em 25, 50, 75, ou 100 clientes, acaba por também aumentar o nível de utilização da CPU. Por exemplo, considerando 25 clientes e o tempo de chegada  $T1=0,01s$ , tem-se a utilização da CPU em 88%. Considerando o mesmo tempo de chegada, e ao seu aumentar o número máximo de clientes simultâneos realizando requisições ao sistema para 100, temos o nível de utilização da CPU em 96,92%.

Ainda considerando o Apache CloudStack, a Tabela 30 apresenta os resultados desse mesmo experimento, ocasionando impacto na utilização da Memória RAM. Ao se comparar com a média da utilização por tempo de chegada, é possível visualizar que, para um menor tempo  $T1=0,01s$ , a utilização média da memória RAM é 84,84%. Já assumindo o  $T1=1,0s$ , temos a utilização em 82,46%. Sendo assim, pode-se perceber que, ao se afastar a chegada de clientes ao sistema, tem-se uma redução na utilização tanto da CPU como da memória RAM.

Tabela 30 – Utilização da memória RAM (%) Apache CloudStack.

T1 (s)	25 Clientes	50 Clientes	75 Clientes	100 Clientes	Média
0,01	77,91	81,00	86,30	94,14	84,84
0,05	77,56	80,50	85,80	94,08	84,49
0,1	77,50	80,45	85,05	92,85	83,96
0,2	76,42	80,36	85,25	92,00	83,51
0,3	77,25	80,27	84,77	91,64	83,48
0,4	77,24	80,17	83,86	91,62	83,22
0,5	77,01	80,10	83,40	91,54	83,01
1,0	75,22	79,74	83,00	91,86	82,46
Média	77,01	80,32	84,68	92,47	83,62

A Figura 34 mostra esses mesmos resultados graficamente, onde a variação do tempo de chegada de clientes ao sistema impacta na utilização da memória RAM no sistema Apache CloudStack. Similarmente à utilização da CPU, um espaçamento maior do tempo de chegada de clientes ao sistema faz com que se reduza o nível de utilização da memória RAM. Por exemplo, assumindo 100 clientes, com o  $T1 = 0,01$  se tem uma utilização da memória acima de 94%. Já para o  $T1=1,0$  e com o mesmo número de clientes no sistema, se obteve uma redução para baixo de 92% na utilização da memória.

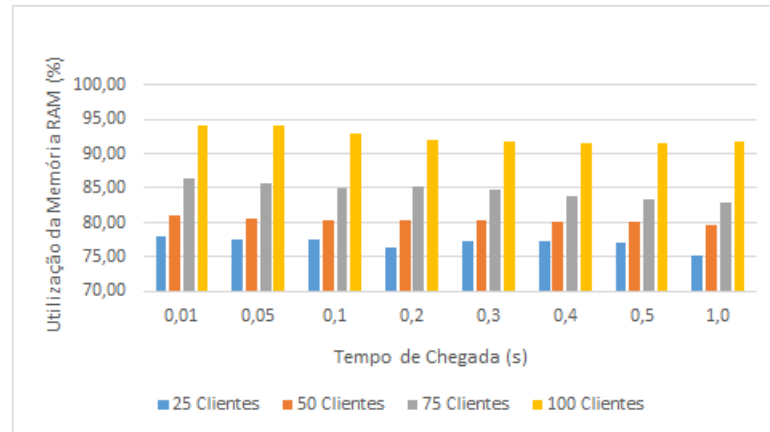


Figura 34 – Resultado da Utilização da RAM com variação do T1 - Apache CloudStack

### OpenStack

O mesmo experimento acima foi realizado assumindo o OpenStack. A Tabela 31 apresenta os resultados obtidos mostrando o impacto no nível de utilização da CPU. Por exemplo, a média da utilização da CPU caiu de 88,79%, assumindo 0,01s como tempo de chegada, para 86,14% com o  $T1=1,0s$ .

Tabela 31 – Utilização da CPU (%) OpenStack.

T1 (s)	25 Clientes	50 Clientes	75 Clientes	100 Clientes	Média
0,01	82,78	85,74	89,41	97,24	88,79
0,05	81,62	85,02	88,22	96,15	87,75
0,1	81,39	84,87	87,69	95,70	87,41
0,2	81,37	84,90	87,66	95,66	87,40
0,3	81,34	84,74	87,65	95,35	87,27
0,4	81,19	84,35	87,49	95,21	87,06
0,5	81,15	84,27	86,84	95,02	86,82
1,0	80,10	83,93	85,80	94,73	86,14
Média	81,37	84,73	87,60	95,63	87,33

A Figura 35 ilustra o impacto da variação do tempo de chegada na utilização da CPU no sistema OpenStack. Pode-se perceber que, quando há um aumento do tempo de chegada ao sistema dos clientes, acaba ocasionando uma redução na utilização da CPU.

A Tabela 32 apresenta os resultados impactando a utilização da memória RAM. Esse impacto pode ser percebido quando se compara, por exemplo, a média da utilização com o tempo de chegada em 0,01s (utilização da memória RAM em 79,68%) com o



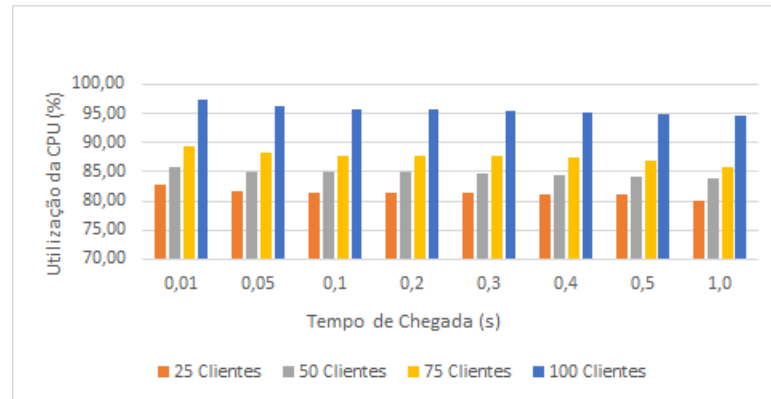


Figura 35 – Resultado da Utilização da CPU com variação do T1 - OpenStack

resultado obtido para o  $T1=1,0$  (76,10%).

Tabela 32 – Utilização da memória RAM (%) OpenStack.

T1 (s)	25 Clientes	50 Clientes	75 Clientes	100 Clientes	Média
0,01	71,21	73,27	79,56	94,26	79,58
0,05	70,00	72,79	78,41	92,33	78,38
0,1	69,10	73,13	77,84	91,62	77,92
0,2	68,65	72,53	77,76	91,51	77,61
0,3	67,55	72,38	77,64	90,90	77,12
0,4	67,53	72,08	77,34	90,88	76,96
0,5	67,25	71,51	76,21	90,17	76,29
1,0	67,12	71,21	76,06	90,00	76,10
Média	68,55	72,36	77,60	91,46	77,49

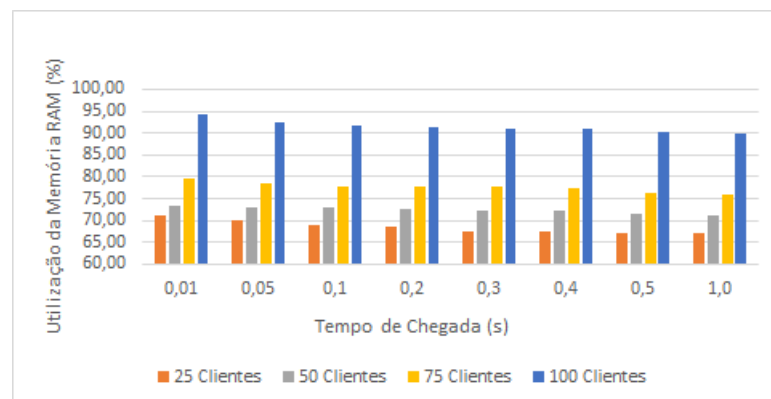


Figura 36 – Resultado da Utilização da RAM com variação do T1 - OpenStack

A Figura 36 mostra a variação da utilização da memória RAM, no sistema Apache CloudStack, quando se tem uma variação do tempo de chegada de clientes ao sistema.

É possível notar que uma redução da utilização da RAM ocorre sempre que se espaça a chegada de clientes ao sistema. Além disso, o aumento da quantidade de clientes no sistema realizando requisições simultaneamente também acaba por impactar em um aumento na utilização da memória RAM.

### Comparação Apache CloudStack vs OpenStack

A Figura 37 mostra um comparativo dos resultados obtidos para utilização da CPU do ambiente Apache CloudStack vs OpenStack. O aumento do tempo de chegada ( $T1$ ) e da quantidade de clientes ( $QR$ ) no sistema acabam por impactar na utilização da CPU. Ficou evidenciado que, com o aumento da quantidade de clientes, há uma maior utilização da CPU. Já quando se aumenta o espaçamento entre a chegada de clientes, ocorre uma redução da utilização da CPU. Esse mesmo comportamento ocorre para os dois sistemas de nuvem Apache CloudStack e OpenStack.

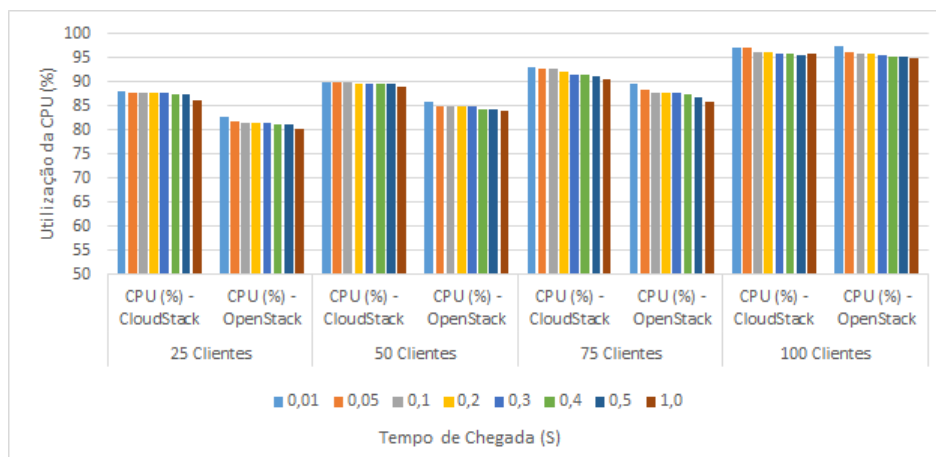


Figura 37 – Resultado da simulação da utilização da CPU do Apache CloudStack e OpenStack

A Figura 38 ilustra os resultados obtidos comparando o Apache CloudStack vs OpenStack para a utilização da Memória RAM. O aumento do tempo de chegada ( $T1$ ) e da quantidade de clientes ( $QR$ ) impactam a utilização da memória RAM. Por exemplo, nessa figura é possível perceber que o aumento da quantidade de clientes no sistema ocasiona uma maior utilização da Memória RAM. Além disso, assumindo uma redução da frequência da chegada de clientes ao sistema, ocorre uma redução no nível de utilização da memória. Esse mesmo padrão de comportamento ocorreu nos dois sistemas de nuvem Apache CloudStack e OpenStack.

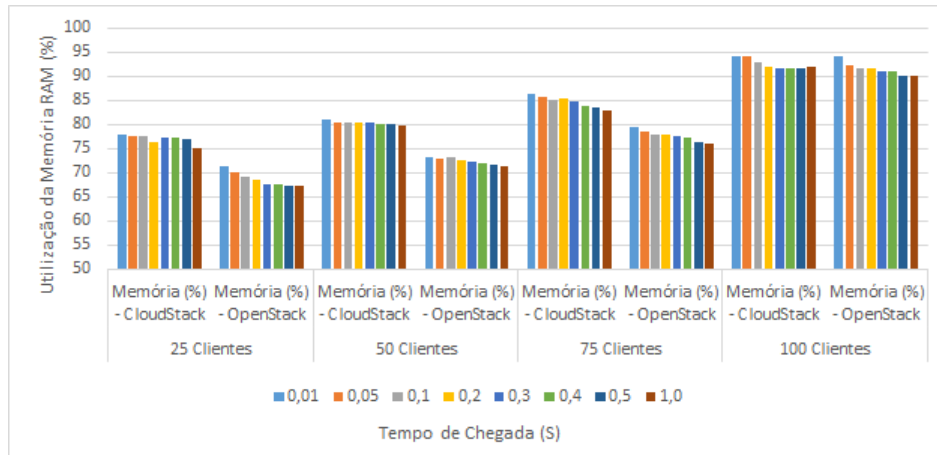


Figura 38 – Resultado da simulação da utilização da Memória RAM do Apache CloudStack e OpenStack

Com 4,33% na média a menos na utilização da CPU e 7,44% a menos na utilização da memória RAM, o OpenStack acabou aparentando ser a melhor solução em nuvem para tais cenários adotados.

## 6.6 Estudo de Caso VI

Esta seção apresenta o último estudo de caso, cujo objetivo é o de mostrar os resultados obtidos para o consumo de energia ao se utilizar o modelo SPN proposto (ver Seção 5.1). Para isso, foram realizadas comparações do consumo de energia entre os dois sistemas em nuvem avaliados. Foi utilizada Equação 5.1 para o Apache CloudStack e a Equação 5.2 para o OpenStack.

O modelo SPN foi utilizado com os parâmetros para cada sistema de nuvem (Apache CloudStack e OpenStack). Esse estudo de caso adota os seguintes parâmetros de entrada para o modelo proposto representando o Apache CloudStack e o OpenStack (ver Tabela 12). Em ambos os casos, foram adotados 2 CPUs ( $QC=2$ ), 4 GB de memória RAM ( $QM=4$ ) e o tempo de chegada entre um cliente e outro ao sistema de 0.1 segundos ( $TChegada=0.1$ ).

A Tabela 33 apresenta os resultados obtidos para o consumo de energia (em Watts) para os dois sistemas em nuvem avaliados, Apache CloudStack e OpenStack. O consumo de energia do *Controller* (máquina física) para o OpenStack apresentou uma diferença de 2,5%, para 25 clientes, em comparação com o OpenStack. Isso foi obtido ao se avaliar o mesmo cenário para cada um dos ambientes. Por outro lado, ao se comparar a média de todos os resultados avaliados (com 25, 50, 75 e 100 clientes), o Apache CloudStack

apresentou um menor consumo energético, com uma média de 68,06 (watts) vs 68,33 (watts) do OpenStack.

Tabela 33 – Comparação do Consumo de energia OpenStack vs Apache CloudStack.

QR	CloudStack.	Int. Conf.	OpenStack	Int. Conf.	Diferença(%)
25	65,58	[65,52; 67,87]	67,22	[64,08; 67,36]	2,50 %
50	68,19	[67,38; 68,99]	68,18	[68,12; 68,84]	-0,01 %
75	68,76	[66,53; 68,80]	68,57	[68,29; 68,85]	-0,28 %
100	69,70	[69,02; 70,20]	69,36	[68,96; 69,77]	-0,49 %
Média	68,06		68,33	-	

A mesma Tabela apresenta os intervalos de confiança obtidos no modelo SPN com a ferramenta Mercury, para o consumo de energia (em Watts) dos sistemas em nuvem Apache CloudStack e OpenStack, evidenciando que os valores estão dentro do intervalo de confiança esperado.

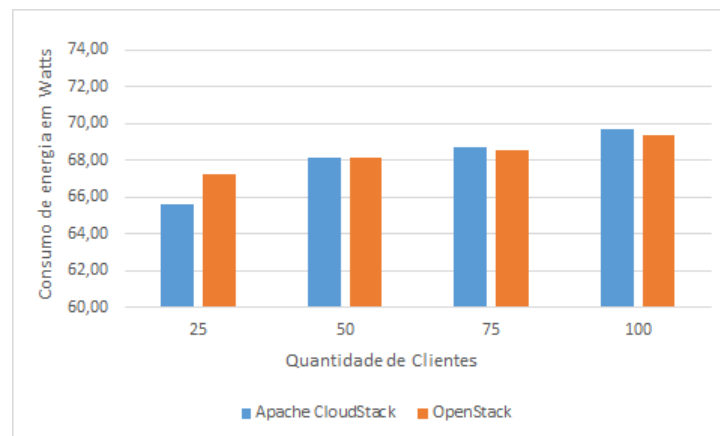


Figura 39 – Resultado da simulação do consumo de energia(Watts) - OpenStack vs Apache CloudStack

A Figura 39 mostra o resultado da comparação do consumo de energia entre o sistema OpenStack e Apache CloudStack. Neste comparativo foi observado que, o sistema OpenStack utiliza mais energia, quando comparamos a média dos quatro cenários estudados.

A Figura 40 mostra o resultado da comparação do consumo de energia medido e da simulação, utilizando o modelo SPN para os sistemas OpenStack e Apache CloudStack. Neste comparativo, foi observado que o sistema OpenStack demanda um maior consumo de energia, quando comparamos a média dos quatro cenários estudados.

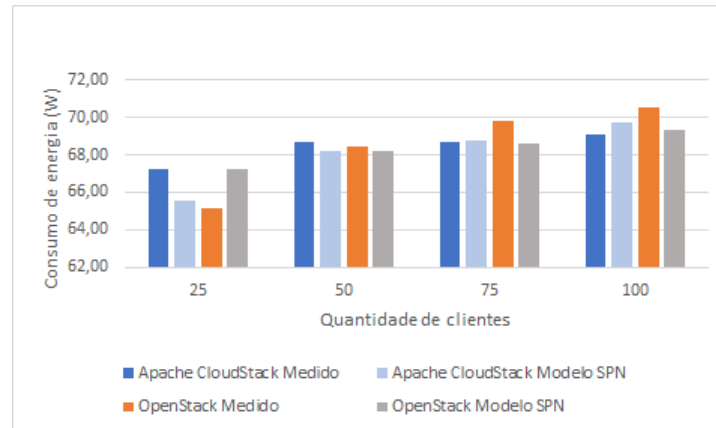


Figura 40 – Resultado da medição e simulação do consumo de energia(Watts) - OpenStack vs Apache CloudStack

Com a finalidade de, estatisticamente, analisar a diferença entre o consumo de energia entre os dois sistemas, a Tabela 34 apresenta o resultado do Teste T. Neste comparativo, o valor de  $P$  ficou em 0,006. Para que as duas amostras sejam consideradas distintas, o valor de  $P$  deverá ser  $< 0,05$ . Sendo assim, o Apache CloudStack possui o consumo de energia diferente do OpenStack. Através do teste, podemos concluir que as duas amostras não são estatisticamente similares, com 95% de confiança.

Tabela 34 – Teste-t: duas amostras em par para a medição do consumo de energia (W).

<b>Estatística</b>	<b>Resultado</b>	
Quantidade (Número)	120	
Desvio Padrão	1,7849	
Nível de Confiança	95%	
Média	0,45141	
Limite Inferior	0,12878	
Limite Superior	0,77404	
Valor de P	0,006	
Estatística	OpenStack	Apache CloudStack
Média	68,768	68,316
Desv.Pad	2,2695	0,88454

O consumo de energia dos dois sistemas é estatisticamente distinto. Posto isto, para o cenário em estudo, foi observado um maior consumo de energia do OpenStack quando comparado com o mesmo cenário para o Apache CloudStack. A Tabela 11 apresenta os resultados da medição obtidos para o tempo de resposta. Nesse exemplo, OpenStack demandou um tempo média de 49,67% superior para o tempo de resposta em comparação com o Apache CloudStack.

## 6.7 Considerações finais

Este capítulo apresentou seis estudos de caso com o objetivo de validar o modelo SPN proposto e, principalmente, avaliar e comparar as infraestruturas de nuvem computacionais Apache CloudStack e a CloudStack em relação a desempenho e consumo de energia. Inicialmente, o primeiro estudo apresentou um comparativo do desempenho (utilização da CPU e memória RAM, vazão e tempo de resposta) e do consumo de energia do OpenStack e do Apache CloudStack. Cada um desses sistemas passou por um processo de medição separadamente, e que foi repetido 30 vezes. Posteriormente, o segundo estudo de caso apresentou a validação do modelo SPN em relação à utilização da CPU e memória RAM, e por fim, com o consumo de energia. O estudo de caso III mostrou os resultados obtidos para a vazão e o tempo de resposta da simulação realizada no modelo proposto. Já o estudo de caso IV evidenciou os impactos das variações da quantidade da CPU ( $QC$ ) e da quantidade de memória RAM ( $QM$ ). O estudo de caso V teve como objetivo quantificar os impactos decorrentes da variação do tempo de chegada e da quantidade de clientes ( $QR$ ) ao sistema. Por fim, o estudo de caso VI fez o comparativo do consumo de energia entre os dois sistemas de nuvem, evidenciando que o OpenStack apresentou um maior consumo energético quando comparado à média das medições realizadas.

## 7 Conclusão

Devido ao aumento da utilização da tecnologia da informação em ambientes online para o ensino e aprendizagem, há uma necessidade crescente de que tais sistemas tenham o desempenho para atender as demandas das instituições e seus usuários. A falta de uma infraestrutura adequada para suportar acesso de múltiplos usuários simultaneamente no ambiente de aprendizagem virtual AVA, Moodle, pode ocasionar um desempenho insatisfatório. Este trabalho propôs uma metodologia para avaliação de desempenho baseada em modelo estocástico, SPN, considerando a carga de vários clientes simultâneos no sistema AVA ao realizar um conjunto de atividades, entre elas, o *upload* de um arquivo de 1MB. Este trabalho também apresentou uma análise de seis estudos de caso. No primeiro, o desempenho e o consumo de energia são comparados entre o OpenStack e o Apache CloudStack. O segundo estudo de caso apresentou a validação do modelo SPN através das métricas de utilização da CPU e memória RAM, e do consumo de energia. O estudo de caso III teve como objetivo mostrar os resultados obtidos para a vazão e o tempo de resposta através do modelo proposto. Os demais estudos de caso tiveram como objetivo realizar experimentos para mostrar o comportamento do sistema nos mais diversos cenários com o modelo proposto. Esses cenários foram realizados via modelagem, uma vez que era inviável realizar medição para essa diversidade de cenários analisados.

O modelo estocástico proposto mostrou, através dos resultados, que pode auxiliar na escolha do sistema em nuvem privada. Ficou evidenciado que o sistema Apache CloudStack obteve um melhor desempenho nas métricas estudadas nesta pesquisa. Por exemplo, comparado ao OpenStack, o Apache CloudStack demandou um tempo 49,46% inferior, no consumo de energia foi obtida uma redução de 28,96%, e na utilização da CPU houve um ganho de 1,79%; e, por fim, a vazão foi incrementada em 59,06%. Apenas na utilização da memória RAM o OpenStack apresentou um melhor desempenho, com um consumo menor em 4,45%.

### 7.1 Contribuições

A seguir, são apresentadas as principais contribuições deste trabalho:

- Uma solução integrada composta de uma metodologia e modelo de SPN, para a escolha de uma infraestruturas de nuvens privadas entre o Apache CloudStack e OpenStack, considerando aspectos de desempenho e consumo de energia;
- Avaliação de desempenho do sistema em nuvem Apache CloudStack e OpenStack em cenários diversos com o Moodle sendo hospedado em cada uma dessas infraestruturas;
- Comparativo do consumo de energia entre os dois sistemas de nuvem Apache CloudStack e OpenStack;
- Estratégia de regressão linear para previsão do comportamento do sistema analisado.

### 7.1.1 Produção Científica

- Aceitação de artigo em periódico (Qualis B2): DA SILVA, Alison Vinicius Gomes; DE LIMA, Claudemir Jeremias; DE ALMEIDA CALLOU, Gustavo Rau. Análise de desempenho do ambiente virtual de aprendizagem na nuvem privada apache cloudstack. GESTÃO. Org, v. 17, n. 8, p. 120-133, 2019.
- Prêmio de melhor artigo Científico da oitava edição do SBTI à obra "Análise de Desempenho do Ambiente Virtual de Aprendizagem na Nuvem Privada Apache CloudStack" em 2019.
- Aceitação de artigo em anais de congresso nacional (Qualis C): LIMA, Claudemir J. et al. Desenvolvimento de um Protótipo de Identificação do Nível de Escrita para Dispositivos Móveis: uma Ferramenta de Sondagem no Ensino. In: Anais do IV Congresso sobre Tecnologias na Educação. SBC, 2019. p. 589-595.
- Artigo submetido para a Revista Brasileira de Computação Aplicada (Qualis B3): DE LIMA, Claudemir Jeremias; DA SILVA, Alison Vinicius Gomes; DE ALMEIDA CALLOU, Gustavo Rau. Avaliação de Desempenho e Consumo de Energia de um Ambiente Virtual de Aprendizagem em Nuvens Privadas. RBCA-Revista Brasileira de Computação Aplicada, 2020.

## 7.2 Limitações

As limitações deste trabalho são expostas na sequência.



- O trabalho proposto disponibiliza um ambiente restrito quanto às configurações de *hardware* utilizadas para compor uma infraestrutura: CPU, memória RAM e disco de armazenamento HD são estáticos.
- Este trabalho teve como foco o estudo de métricas relacionadas ao desempenho dos dois sistemas em nuvem. Outras métricas relacionadas à disponibilidade não foram consideradas.
- A análise do consumo de energia no modelo SPN tratada neste trabalho, usa fórmulas lineares. Como todos sabemos, quanto mais confiáveis as informações sobre o consumo de energia, melhor será a visualização do impacto no planejamento da melhor estrutura em nuvem.
- As regressões lineares propostas neste trabalho poderiam ter sido feitas considerando um maior número de elementos. Todavia, a estratégia proposta já fornece uma estimativa aos projetistas das infraestruturas dos ambientes para AVAs, tendo em vista que cada ponto utilizado na regressão linear representa a média de uma mostra com 30 elementos.
- Quanto ao modelo e às técnicas de análise utilizadas neste trabalho, é necessário um conhecimento básico para a parametrização no modelo proposto e nas ferramentas utilizadas para coleta dos dados na medição e simulação.
- Quantidade de usuários nos estudos limitada até 100 usuários simultaneamente.

### 7.3 Trabalhos futuros

Vários direcionamentos podem ser seguidos e propostos a partir deste trabalho. A seguir, são enumeradas algumas das possíveis continuações desta pesquisa.

- Os estudos de caso propostos neste trabalho foram baseados no planejamento de infraestrutura de nuvens privadas com Apache CloudStack e com OpenStack, hospedando o Moodle como um serviço na VM. No entanto, este trabalho pode ser estendido para realizar a proposição de um estudo de caso baseado no planejamento de infraestruturas de nuvem pública e também nuvem híbrida.
- Estudos de caso mais complexos, com a utilização de milhares de usuários acessando o Moodle ao mesmo tempo, podem e devem ser realizados no futuro. Para isso, um estudo da computação em nuvem que visa somar o poder computacional de várias

máquinas juntas pode vir a ser útil.

- Na instalação do Apache CloudStack, utilizamos o *hypervisor* KVM, podendo ser utilizado em trabalhos futuros o *hypervisor* XenServer para aplicar um dimensionamento elástico, agregando mais recursos computacionais para a nuvem e, assim, se ter um novo cenário para comparação com o KVM.
- Realizar a proposição de estratégias para previsão comportamental dos sistemas analisados a partir de uma base de dados aberta. Por exemplo, pode-se utilizar séries temporais, regressão linear e métodos de inteligência artificial a fim de prever o comportamento desses sistemas baseados no passado.
- Um estudo integrado de desempenho e disponibilidade pode ser realizado para complementar a pesquisa já realizada neste trabalho.
- Realizar a proposição de estratégias para previsão comportamental dos sistemas analisados, considerando assistir *podcasts*, videoaulas ou vídeos do *Youtube*.

## Referências

- APACHE. **Jmeter**. 2019. Disponível em: <<https://jmeter.apache.org/index.html>>.
- APACHE1. **Apache CloudStack - Open Source Cloud Computing**. 2020. [Http://cloudstack.apache.org/](http://cloudstack.apache.org/).
- BARROS, J.; CALLOU, G.; GONÇALVES, G. Análise integrada de desempenho e consumo de energia em sistemas de armazenamento de dados distribuídos. In: SBC. **Anais do XV Workshop em Clouds e Aplicações**. [S.l.], 2017.
- BAUER, E.; ADAMS, R. **Reliability and availability of cloud computing**. [S.l.]: John Wiley & Sons, 2012.
- BORBA, E. R. **Modelagem de desempenho e disponibilidade para sistemas de armazenamento híbridos**. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2017.
- BRUSCHI, G. C.; SPOLON, R.; PAURO, L. L.; LOBATO, R. S.; MANACERO, A.; CAVENAGHI, M. A. Stackact: Performance evaluation in an iaas cloud multilayer. In: IEEE. **2016 15th International Symposium on Parallel and Distributed Computing (ISPDC)**. [S.l.], 2016. p. 149–156.
- CALLOU, G.; MACIEL, P.; TAVARES, E.; ANDRADE, E.; NOGUEIRA, B.; ARAUJO, C.; CUNHA, P. Energy consumption and execution time estimation of embedded system applications. **Microprocessors and Microsystems**, Elsevier, v. 35, n. 4, p. 426–440, 2011.
- CALLOU, G. Rau de A. **Energy consumption and execution time estimation of embedded system applications**. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2009.
- DANTAS, J. R. Planejamento de infraestrutura de nuvens computacionais para serviço de vod streaming considerando desempenho, disponibilidade e custo. Universidade Federal de Pernambuco, 2018.
- DESHMUKH, P. P.; AMDANI, S. Virtual memory management of private cloud using openstack api. In: IEEE. **2019 IEEE 5th International Conference for Convergence in Technology (I2CT)**. [S.l.], 2019. p. 1–4.
- FÉ, I. d. S. **Planejamento de transcodificação de vídeo em nuvem elástica**. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2017.
- FILHO, I. J. M.; CARVALHO, R. S.; GOMES, A. S.; PERRIS, P. A. da R.; RODRIGUES, R. L.; FELICIANO, F. D. de O. Análise comparativa da usabilidade dos ambientes de gestão da aprendizagem amadeus e moodle. **Revista Brasileira de Informática na Educação**, v. 22, n. 01, p. 107, 2014.
- GERMAN, R. **Performance analysis of communication systems with non-Markovian stochastic Petri nets**. [S.l.]: John Wiley & Sons, Inc., 2000.

JMETER, A. Disponível em: <http://jmeter.apache.org>. **Access in: October, 30th, 2019.**

KIM, H.; LEE, H.; KIM, W.; KIM, Y. A trust evaluation model for qos guarantee in cloud systems. **International Journal of Grid and Distributed Computing**, v. 3, n. 1, p. 1–10, 2010.

KUMAR, R.; JAIN, K.; MAHARWAL, H.; JAIN, N.; DADHICH, A. Apache cloudstack: Open source infrastructure as a service cloud computing platform. **Proceedings of the International Journal of advancement in Engineering technology, Management and Applied Science**, p. 111–116, 2014.

LILJA, D. J. **Measuring computer performance: a practitioner's guide**. [S.l.]: Cambridge university press, 2005.

MACHIDA, F.; ANDRADE, E.; KIM, D. S.; TRIVEDI, K. S. Candy: Component-based availability modeling framework for cloud service management using sysml. In: **IEEE. 2011 IEEE 30th International Symposium on Reliable Distributed Systems**. [S.l.], 2011. p. 209–218.

MANSUR, A. F. U.; GOMES, S. S.; LOPES, A. d. A.; BIAZUS, M. Novos rumos para a informática na educação pelo uso da computação em nuvem (cloud education): Um estudo de caso do google apps. In: **Foz do Iguaçu: Anais do XVI Congresso Internacional ABED de Educação a Distância**. [S.l.: s.n.], 2010. p. 35.

MARON, C. A.; GRIEBLER, D.; VOGEL, A.; SCHEPKE, C. Avaliação e comparação do desempenho das ferramentas openstack e opennebula. **12th Escola Regional de Redes de Computadores (ERRC)**, p. 1–5, 2014.

MATOS, M. A. Manual operacional para a regressão linear. **Faculdade de Engenharia da Universidade do Porto**, p. 63, 1995.

Merlin, P.; Farber, D. Recoverability of communication protocols - implications of a theoretical study. **IEEE Transactions on Communications**, v. 24, n. 9, p. 1036–1043, 1976.

MONTGOMERY, D. C.; RUNGER, G. C. **Applied Statistics and Probability for Engineers**. [S.l.]: John Wiley and Sons, 2003.

MOODLE. **moodle.org**. 2020. <https://moodle.org/>.

MORAIS, A. M. de; CALLOU, G. R. de A.; LINS, F. A. A. Simulação e avaliação de desempenho de uma rede blockchain utilizando containers docker. **Cadernos do IME-Série Informática**, v. 44, p. 73–87, 2020.

MULLERIKKAL, J. P.; SASTRI, Y. A comparative study of openstack and cloudstack. In: **IEEE. 2015 Fifth International Conference on Advances in Computing and Communications (ICACC)**. [S.l.], 2015. p. 81–84.

MURATA, T. Petri nets: Properties, analysis and applications. **Proceedings of the IEEE, IEEE**, v. 77, n. 4, p. 541–580, 1989.

NETO, J. R. d. M. **Modelagem e análise de desempenho e consumo de energia em aplicações móveis**. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2015.

NIST. **NIST Cloud Computing Program**. 2014. [Http://www.nsti.gov/itl/cloud/index.cfm](http://www.nsti.gov/itl/cloud/index.cfm).

OPENSTACK. **OpenStack**. 2020. [Https://www.openstack.org/](https://www.openstack.org/).

PAURO, L. L. Auditoria e monitoramento de eventos inconsistentes em instâncias de máquinas virtuais em iaas no orquestrador apache cloudstack. Universidade Estadual Paulista (UNESP), 2016.

PEIXOTO, M. L. M. **Oferecimento de QoS para computação em nuvens por meio de metaescalamento**. Tese (Doutorado) — Universidade de São Paulo, 2012.

PETRI, C. A. Kommunikation mit automaten. 1962.

PRODANOV, C. C.; FREITAS, E. C. de. **Metodologia do trabalho científico: métodos e técnicas da pesquisa e do trabalho acadêmico-2ª Edição**. [S.l.]: Editora Feevale, 2013.

RATNA ANAK AGUNG PUTRI E WIRIANATA, T. e. E. F. A. e. I. I. e. H. D. e. P. P. D. Projeto de rede de computação em nuvem para implementação de computação de alto desempenho na plataforma openstack. In: **Proceedings of the 3rd International Conference on Communication and Information Processing**. [S.l.: s.n.], 2017. p. 356–360.

RODRIGUES, R. L.; MEDEIROS, F. P. D.; GOMES, A. S. Modelo de regressão linear aplicado à previsão de desempenho de estudantes em ambiente de aprendizagem. In: **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)**. [S.l.: s.n.], 2013. v. 24, n. 1, p. 607.

SABHARWAL, N. **Apache CloudStack cloud computing**. [S.l.]: Packt Publishing Ltd, 2013.

SILVA, S. Acessibilidade digital em ambientes virtuais de aprendizagem. **Revista GEINTEC-Gestão, Inovação e Tecnologias**, v. 2, n. 3, p. 245–254, 2012.

SOUSA, E.; LINS, F.; TAVARES, E.; MACIEL, P. Cloud infrastructure planning considering different redundancy mechanisms. **Computing**, Springer, v. 99, n. 9, p. 841–864, 2017.

SOUSA, E. T. G. d. Modelagem de desempenho, dependabilidade e custo para o planejamento de infraestruturas de nuvens privadas. UNIVERSIDADE FEDERAL DE PERNAMBUCO, 2015.

SYSTEMS, O. **OpenNebula – The open source Cloud Management Platform**. 2020. [Https://opennebula.io/](https://opennebula.io/).

TORRES, E.; CALLOU, G.; ALVES, G.; ACCIOLY, J.; GUSTAVO, H. Storage services in private clouds: Analysis, performance and availability modeling. In: IEEE. **2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)**. [S.l.], 2016. p. 003288–003293.

TORRES, E. B.; CALLOU, G.; ALVES, G.; ACCIOLY, J.; GUSTAVO, H. Performance and availability evaluation of storage services in private cloud. In: IEEE. **2016 11th Iberian Conference on Information Systems and Technologies (CISTI)**. [S.l.], 2016. p. 1–6.

TORRES, T. P. **Planejamento de infraestruturas em nuvens privadas para sistemas de educação a distância utilizando modelagem de dependabilidade e custo**. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2017.

TRIVEDI, K. S.; HUNTER, S.; GARG, S.; FRICKS, R. Reliability analysis techniques explored through a communication network example. Citeseer, 1996.

UFRPE1. **Relatório de Gestão**. 2019. Disponível em: <<http://www.ufrpe.br/sites/www.ufrpe.br/files/RELAT%C3%93RIO%20DE%20GESTAO%202018.pdf>>.

Vogel, A.; Griebler, D.; Maron, C. A. F.; Schepke, C.; Fernandes, L. G. Private iaas clouds: A comparative analysis of opennebula, cloudstack and openstack. In: **2016 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)**. [S.l.: s.n.], 2016. p. 672–679. ISSN 2377-5750.

VOGEL, A.; GRIEBLER, D.; SCHEPKE, C.; FERNANDES, L. G. An intra-cloud networking performance evaluation on cloudstack environment. In: IEEE. **2017 25th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)**. [S.l.], 2017. p. 468–472.

VOGEL ADRIANO E GRIEBLER, D. e. M. C. A. e. S. C. e. F. L. G. nuvens iaas privadas: uma análise comparativa do opennebula, cloudstack e openstack. In: **2016 24<sup>a</sup> Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)**. [S.l.]: IEEE, 2016. p. 672–679.

WANG QIANG E XU, P. e. Z. Y. e. C. X. Eppminer: um pacote de benchmark estendido para caracterização de energia, potência e desempenho de arquitetura heterogênea. In: **Anais da Oitava Conferência Internacional sobre Sistemas de Energia do Futuro**. [S.l.: s.n.], 2017. p. 23–33.

ZHANG, Q.; CHENG, L.; BOUTABA, R. Cloud computing: state-of-the-art and research challenges. **Journal of internet services and applications**, Springer, v. 1, n. 1, p. 7–18, 2010.

ZOTA, R.; PETRE, I. A. An overview of the most important reference architectures for cloud computing. **Informatica Economica**, v. 18, n. 4, 2014.