



UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA

LICENCIATURA E ENGENHARIA INFORMÁTICA

RELATÓRIO DE ESTÁGIO PROFISSIONAL

**PROPOSTA DE IMPLANTAÇÃO DE UMA IAAS EM
CLOUD PRIVADA ATRAVÉS DE UMA FERRAMENTA
*OPEN SOURCE***

Caso de Estudo: Grupo Entrepasto Auto Maputo – (ServiSIS Moçambique, SA)

Autor:

MATAVEL, Eco Vick Vasco

Supervisor:

Dr. Alfredo Covele

Supervisor da Instituição

Dr. Duncan Mureyani

Maputo, 15 Julho de 2024



UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA

LICENCIATURA E ENGENHARIA INFORMÁTICA

RELATÓRIO DE ESTÁGIO PROFISSIONAL

**PROPOSTA DE IMPLANTAÇÃO DE UMA IAAS EM
CLOUD PRIVADA ATRAVÉS DE UMA FERRAMENTA
*OPEN SOURCE***

Caso de Estudo: Grupo Entrepasto Auto Maputo – (ServiSIS Moçambique, SA)

Autor:

MATAVEL, Eco Vick Vasco

Supervisor:

Dr. Alfredo Covele

Supervisor da Instituição

Dr. Duncan Mureyani

Maputo, 15 Julho de 2024



UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA

TERMO DE ENTREGA DO RELATÓRIO DE ESTÁGIO PROFISSIONAL

Declaro que o estudante **Eco Vick Vasco Matavel** entregou no dia ___ / ___ / **2024** as ___ cópias do relatório de **Estágio Profissional**, com a referência: 2024EIEPD___, intitulado: **Proposta De Implantação De Uma IaaS Em Cloud Privada Através De Uma Ferramenta Open Source**

Maputo, ___ de _____ de 20__

O chefe de secretaria



UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA

DECLARAÇÃO DE HONRA

Declaro sob compromisso de honra que o presente trabalho é resultado da minha investigação e que foi concebido para ser submetido apenas para a aprovação no curso de Engenharia Informática na Faculdade de Engenharia da Universidade Eduardo Mondlane.

Maputo, ____ de _____ de 20__

O Autor

(Eco Vick Vasco Matavel)

Dedicatória

Dedico esse trabalho em memória da minha amada mãe e professora Salmina carinhosamente chamada por mana, pessoa resiliente que tive a alegria de conviver e que me revelou o valor de uma vida plena e harmônica apesar das adversidades.

Onde estiver saiba que essa vitória é mais sua que minha, grato por sempre acreditar no meu potencial e ver em mim qualidades que ninguém via, você fez-me ver o verdadeiro sentido da vida, ensinando o valor do amor, compaixão, força e tudo de bom que pude colher de você. Onde estiver, saiba que o vazio que deixou é impreenchível e embora não possa contemplar com seus lindos olhos o fruto do seu esforço, o seu legado está tatuado em nós. Te amo, descanse em paz mana Salmina.

Agradecimentos

A Primeiramente agradecer e engradecer a Deus por me guardar a cada dia no decorrer do curso.

A minha família (aos meus pais e primos) pelo apoio no meu desenvolvimento pessoal, amigos e colegas de faculdade nomeadamente, eng. Marcos Sate, eng. Ednilson Chiziane, eng. Ricardo Manhice, Tomas Penicela, Joel Augusto, Vicente Magais, Marcio De Jesus, Rafael Machava, Arnaldo Manguene, Yula Guivala , Lina Cossa, Amilcar Paco, Joaquim Cumbe (Quebra), e a todos que com suas competências contribuíram muito para que eu entendesse que nas derrotas e nas vitórias há sempre motivos de júbilo. Um agradecimento especial ao meu tio Dr. Simião Matavele pelo apoio incondicional no meu percurso académico, meu muito obrigado pela confiança que depositou em mim e que Deus te abençoe abundantemente, que tudo ti vá bem e muitíssimo obrigado.

A todo corpo docente pela dedicação e entusiasmo demonstrado ao longo do curso, aos colegas pela prontidão na troca de informações, demonstração de amizade e solidariedade.

A todos colaboradores do *open space* da ServiSIS, o local de estágio, pela prontidão em auxiliar na resolução de problemas relacionados as minhas funções diárias para a realização desse trabalho. Grato pela oportunidade, o estágio contribuiu para o meu crescimento pessoal e acima de tudo com o enriquecimento técnico. Considero que a experiencia foi muito satisfatória consegui ultrapassar adversidades e pôr em prática o aprendizado adquirido.

“a minha vida tem sido um exemplo para muitos porque tu tens sido o meu fiel defensor”.

Salmos 71:7

“Quando os ventos da mudança sopram, pessoas levantam barreiras, outras constroem moinhos de vento.”

Érico Veríssimo

Resumo

A *cloud computing* é um paradigma de computação que permite que recursos de computacionais sejam oferecidos através da *internet*. Esses recursos são oferecidos sob demanda e podem ser fornecidos com o mínimo de interação do prestador de serviços em *cloud*.

A *cloud computing* tem algumas características essenciais, destacando a elasticidade que consiste em rapidamente dimensiona os recursos computacionais. Além disso, todos os elementos que podem ser virtualizados são oferecidos como serviços.

Existem três modelos de serviço: Software com serviço (*SaaS*), Plataforma como serviço (*PaaS*) e infraestrutura como serviço (*IaaS*). Outra característica notável da *cloud computing* é que ela consiste em quatro tipos de implantações, *cloud* Privada, *cloud* Pública, *cloud* Comunitária e *cloud* Híbrida.

Este trabalho tem como objetivo propor a instalação e configuração de uma *IaaS* baseada no *software* OpenStack, busca uma dar visão para profissionais de IT que desejam uma visão geral de alto nível do OpenStack e querem saber se OpenStack é a resposta certa para atender às necessidades de TI de Sua organização. Nele também ajudaremos quem quiser estabelecer um ambiente de teste baseado nesta solução em pequena escala para ganhe experiência trabalhando com OpenStack

OpenStack gere medularmente todos os serviços da plataforma *cloud computing* (computação, armazenamento e rede), permitindo disponibilidade, escalabilidade e confiabilidade da infraestrutura. Esta ferramenta permite configurar e implementar uma infraestrutura como (*IaaS*) em *cloud* com um *hardware* básico.

Palavras-chave:

OpenStack, cloud privada, *Data Center*, Ferramentas *Open, IaaS, Cloud Computing*

Abstract

Cloud computing is a computing paradigm that allows computing resources to be offered over the internet. These resources are offered on demand and can be provided with minimal interaction from the cloud service provider.

Cloud computing has some essential characteristics, highlighting the elasticity that consists of quickly scaling computational resources. Furthermore, all elements that can be virtualized are offered as services.

There are three service models: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). Another notable feature of cloud computing is that it consists of four types of deployments, Private cloud, Public cloud, Community cloud and Hybrid cloud.

This work aims to propose the installation and configuration of an IaaS based on OpenStack software, seeking to provide insight for IT professionals who want a high-level overview of OpenStack and want to know if OpenStack is the right answer to meet the needs of Your organization's IT. In it we will also help anyone who wants to establish a test environment based on this solution on a small scale to gain experience working with OpenStack

OpenStack fundamentally manages all cloud computing platform services (computing, storage and network), allowing availability, scalability and reliability of the infrastructure. This tool allows you to configure and implement an infrastructure such as (IaaS) in the cloud with basic hardware.

Key words:

OpenStack, private cloud, Data Center, Open Tools, IaaS, Cloud Computing

ÍNDICE

1	CAPITULO I: INTRODUÇÃO	1
1.1	Contextualização.....	2
1.2	Motivação.....	3
1.3	Definição do Problema.....	4
1.4	OBJECTIVOS	5
1.4.1	Objetivo Geral.....	5
1.4.2	Objetivos Específicos	5
1.5	Metodologias.....	5
2	CAPITULO II: REVISÃO DE LITERATURA	7
2.1	Definição de <i>Cloud Computing</i>	8
2.2	Modelo Conceptual <i>Cloud Computing</i>	9
2.2.1	Caraterísticas Essências da <i>Cloud Computing</i>	9
2.2.2	Modelo de Serviços <i>Cloud Computing</i>	10
2.2.3	Modelos de Implementação de Serviços <i>Cloud Computing</i>	12
2.3	Papeis na <i>Cloud Computing</i> Mediante o Modelos de Serviços.....	15
2.4	Análise de Risco de Cada Modelo de Serviços.....	15
2.4.1	Modelo de implementação.....	15
2.4.2	Benefícios e Limitações da <i>Cloud Computing</i>	16
2.5	Tecnologias Relacionadas a <i>Cloud Computing</i>	17
2.5.1	WEB 2.0	18
2.5.2	Arquitetura Orientada a Serviços (SOA).....	18
2.5.3	Computação em grade(Grid Computing).....	18
2.5.4	Computação utilitária (Utility computing)	19
2.5.5	Virtualização	19
2.6	Comparação da Arquitectura <i>Cloud</i> e Arquitectura <i>On-Premise</i>	23
2.6.1	Parâmetro de Classificação das Diferenças.....	24

2.7	Noções Básicas de <i>Data Center On-Premise</i>	25
2.7.1	Vantagens e Desvantagens da implantação On-Premises.....	25
2.8	Ferramentas <i>IaaS Open Source</i> Para <i>Cloud Computing</i>	26
2.8.1	OpenStack	26
2.8.2	Arquitetura da plataforma <i>OpenStack</i>	28
2.8.3	Arquitetura lógica do OpenStack	30
2.8.4	OpenNebula	30
2.8.5	Eucalyptus	32
2.8.6	CloudStack	33
2.9	Análise comparativa de soluções <i>IaaS open source</i>	34
2.10	Diferentes <i>Hypervisors</i> suportados pelos <i>softwares</i> de <i>IaaS</i>	35
3	CAPITULO III – CASO DE ESTUDO	36
3.1	Descrição do Caso de Estudo	37
3.1.1	Caracterização do Grupo Entrepósito Origens em Moçambique	37
3.1.2	Atividades do grupo entreposto no mundo	37
3.1.3	Organigrama da Empresa	37
3.1.4	Apresentação da ServiSIS Moçambique	39
3.1.4.1	Missão	39
3.1.4.2	Visão	39
3.1.4.3	Valores	40
3.1.4.4	Serviços Prestados	40
3.1.5	Divisão da Equipa da ServiSIS	40
3.1.6	Descrição da Situação Atual.....	41
3.1.7	Descrição dos constrangimentos.....	41
4	CAPITULO IV – MODELO DE SOLUÇÃO	43
4.1	Proposta de Solução.....	44
4.1.1	Visão esquemática da infraestrutura IaaS em Openstack.....	44

4.2	Recursos Disponíveis para Implementação da Solução	44
4.2.1	Recursos de <i>Hardware</i>	44
4.2.2	SO <i>Host</i> -Windows 10	45
4.2.3	Oracle <i>VirtualBox</i>	46
4.2.4	Etapas Criação e configuração da máquina virtual	47
4.3	Etapas para Instalação do Openstack	50
5	CAPITULO V – CONSIDERAÇÕES FINAIS E RECOMENDAÇÕES	55
5.1	Conclusões	56
5.2	Recomendações	57
5.3	Trabalhos Futuros	58
5.4	Limitações	58
6	REFERÊNCIAS BIBLIOGRÁFICAS.....	59
6.1	REFERÊNCIAS.....	60
7	ANEXOS	1
A.1.1	Guião de entrevista.....	A
A.1.2	Sequencia de passos para instalação do VirtualBox.....	C
A.1.3	Sequencia de passos para instalação SO do ambiente de teste	D
A.1.4	<i>Data Center</i> do caso de Estudo	E

INDICE DE FIGURAS

Figura 1:	Diagrama básico da <i>cloud computing</i>	8
Figura 2:	Modelo arquitetural da <i>cloud computing</i>	9
Figura 3:	Modelos de Implementação de Serviços Cloud Computing	13
Figura 4:	<i>Cloud Computing</i> compatível x Gestão dos riscos.....	16
Figura 5:	Origens da computação em nuvem	17
Figura 6:	<i>Sample Architecture VMs</i>	19
Figura 7:	Type-2 hypervisor	21
Figura 8:	<i>Full Virtualization</i>	21

Figura 9: <i>Para Virtualization</i>	22
Figura 10: Infraestrutura <i>On-premise vs Cloud Computing</i>	24
Figura 11: Membros da OpenStack Foundation.....	26
Figura 12: Projetos do <i>OpenStack</i>	29
Figura 13:Arquitectura lógica típica de uma <i>cloud OpenStack</i>	30
Figura 14: Organograma do Departamento de Recursos Humanos	37
Figura 15: Organograma do Departamento de Direção de Vendas	38
Figura 16: Organograma da Direção Administrativa e Financeira.....	38
<i>Figura 17: Organograma da Direção de Serviços de Informação</i>	38
Figura 18: Arquitetura do sistema	44
Figura 19: Icon do VirtualBox	46
Figura 20: A interface gráfica de usuário (GUI) de login do OpenStack.....	52
Figura 21:exemplo de painel OpenStack criado.....	53
Figura 22: Exemplos de máquinas virtuais criadas	53
Figura 23:Exemplo de redes criadas.....	54
Figura 24: Serviço de identidade.....	54
Figura 25: Data center do caso de estudo	E
Figura 26: proposta de IaaS em Cloud.....	F
Figura 27: Local de estagio	F

INDICE DE TABELAS

Tabela 1: Principais Diferenças entre os modelos de implementação da cloud computing.....	14
<i>Tabela 2: Divisão de responsabilidades entre cliente e fornecedor Cloud</i>	15
Tabela 3: Benefícios da <i>Cloud Computing</i>	16
Tabela 4: Limitações da <i>Cloud Computing</i>	17
Tabela 5: Diferentes formas de virtualização	19
Tabela 6: Diferença entre Virtualização completa e paravirtualização	22
Tabela 7: Infraestrutura <i>On-premise vs Cloud Computing</i>	24
Tabela 8: Vantagens e Desvantagens da implantação On-Premises	25
Tabela 9: Descrição dos Serviços/Projetos da Plataforma <i>OpenStack</i>	29
Tabela 10: Análise comparativa de soluções IaaS open source	34

Tabela 11: Diferentes hypervisores suportados pelos softwares de IaaS	35
Tabela 12: principais características do Dell Latitude E5440	45
Tabela 13: Descrição do SO do Host	45
Tabela 14: Descrição do <i>requisitos mínimos do SO Guest</i>	47

SIGLAS E ABREVIATURAS

4G	<i>Fourth Generation Wireless</i>
5G	<i>Fifth Generation Wireless</i>
AI	<i>Artificial Intelligence</i>
API	<i>Application Programming Interface</i> (interfaces de programação de aplicativos)
BSD	<i>Berkeley Software Distribution</i>
CLI	<i>Command-Line Interface</i> (Interface de Linha de Comando)
CPU	<i>Central Processing Unit</i> (Unidade central de processamento do computador)
CRM	<i>Customer Relationship Management</i> (gestão das relações com os clientes)
DAS	Arquitetura de Sistemas Distribuídos
DHCP	<i>Dynamic Host Configuration Protocol</i>
DNS	<i>Domain Name System</i>
ERP	<i>Enterprise Resource Planning</i> (planeamento de recursos empresariais)
HDD	<i>Hard Disk Drive</i>
HDMI	<i>High-Definition Multimedia Interface</i>
HOT	<i>Heat Orchestration Template</i>
HP	<i>Hewlett-Packard</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IaaS	<i>Infrastructure as a Service</i>
IoT	<i>Internet of Things</i>
IT	<i>Information Technology</i>
NASA	<i>National Aeronautics and Space Administration</i>
NAT	<i>Network Address Translation</i> (tradução de endereço de rede)
PaaS	<i>Platform as a Service</i>
QoS	<i>Quality of Services</i>
REST	<i>Representational States Transfer</i>

SaaS	<i>Software as a Service</i>
ServiSIS	Serviços e Sistemas de Informação
SLA	<i>Service Level Agreements</i>
SSD	<i>Solid State Drive</i>
TIC	Tecnologia de Informação e Comunicação
TPM	<i>Trusted Platform Module</i>
USB	<i>Universal Serial Bus</i>
VM	<i>Virtual Machine</i>
VPN	<i>Virtual Private Network</i>
WSDL	<i>Web Service Description Language</i>

GLOSSÁRIO DE TERMOS

Termo	Definição
AMD	Fabricante de processadores e hardware.
Apache	Servidor web de código aberto que desempenha um papel fundamental na hospedagem de websites na internet.
AWS (Amazon Web Services)	Plataforma de serviços de computação em nuvem da Amazon, oferecendo armazenamento, processamento e outros recursos sob demanda.
Backup	Cópia de segurança de dados para restaurar informações em caso de perda ou dano.
Bluetooth	Tecnologia de comunicação sem fio para troca de dados entre dispositivos em curtas distâncias.
C	Linguagem de programação de baixo nível, amplamente usada em sistemas operacionais e software de desempenho crítico.
C++	Linguagem de programação de médio a alto nível, derivada de C, com suporte a programação orientada a objetos.
Canonical	Empresa que desenvolve o Ubuntu Linux.
Cartão SD	Dispositivo de armazenamento removível usado em câmeras, smartphones e outros dispositivos.
Cisco	Especialista em equipamentos de rede e telecomunicações.

Citrix	Empresa que oferece soluções de virtualização e computação em nuvem.
Cloud	Serviços de computação baseados na internet.
Cluster	Conjunto de computadores que trabalham juntos como um único sistema para melhorar o desempenho e a disponibilidade.
Cluster controller	Componente que gerencia e coordena as atividades de um cluster.
Daemon	Programa ou processo que roda em segundo plano no sistema operacional, executando tarefas específicas.
Data center	Instalação física que abriga servidores e equipamentos de rede para armazenar, gerenciar e distribuir dados.
Dell	Fabricante de computadores, servidores e soluções de rede.
Docker	Plataforma que permite criar, implantar e gerenciar contêineres de software.
Dynamics NAV	Sistema de gestão empresarial (ERP) da Microsoft, agora conhecido como Dynamics 365 Business Central.
EC2	Serviço de computação em nuvem da AWS que fornece capacidade de processamento escalável.
ESX	<i>Hypervisor</i> da VMware para virtualização de servidores.
Firewall	Sistema de segurança que controla o tráfego de rede, permitindo ou bloqueando pacotes de dados com base em um conjunto de regras.
Frameworks	Estruturas de software que facilitam o desenvolvimento de aplicações.
Gestware	Software de gestão empresarial para administração e controle de processos de negócios.
GNU	Projeto de software livre que promove a criação e distribuição de software de código aberto.
Hardware	Componentes físicos de um sistema de computação.
Help Desk	Serviço de suporte técnico para resolver problemas relacionados a TI.
Host	Máquina que hospeda máquinas virtuais ou serviços de rede.
Hypercalls	Chamadas de sistema usadas para comunicação entre máquinas virtuais e o <i>hypervisor</i> .

Hypervisor	Software que cria e executa máquinas virtuais, também conhecido como monitor de máquina virtual (VMM).
IBM	Empresa de tecnologia que oferece hardware, software e serviços.
Intel	Fabricante líder de processadores e hardware.
Interface	Ponto de interação entre diferentes sistemas ou componentes de software.
Internet	Rede global de computadores.
Intranet	Rede privada usada dentro de uma organização.
Java	Linguagem de programação de alto nível usada para desenvolver aplicações independentes de plataforma.
KVM	(Kernel-based Virtual Machine) Solução de virtualização de código aberto para Linux.
Laptop	Computador portátil.
Mainframe	Computador de grande porte usado para processar grandes volumes de dados e executar aplicações críticas.
Middleware	Software que fornece serviços adicionais a aplicações além do sistema operacional.
Multi-tenancy	Arquitetura de software onde uma única instância serve múltiplos usuários (tenants).
Online	Estado de estar conectado à internet.
Open source	Software com código-fonte disponível publicamente.
Patches	Atualizações de software que corrigem bugs ou melhoram a segurança.
Plugin	Componente adicional que estende a funcionalidade de um software principal.
Pool	Conjunto de recursos alocáveis conforme necessário.
Python	Linguagem de programação de alto nível, conhecida por sua simplicidade e legibilidade.
S3	Serviço de armazenamento em nuvem da AWS, conhecido como Simple Storage Service.
Script	Programa ou sequência de instruções que são interpretadas e executadas por outro programa.

Software	Conjunto de instruções que podem ser executadas por um computador.
Smart	Termo genérico para dispositivos inteligentes com capacidade de conexão e automação.
Snapshots	Cópias instantâneas do estado de um sistema ou máquina virtual em um ponto específico no tempo.
Startup	Empresa nova e inovadora em fase inicial de desenvolvimento.
SUSE Linux	Distribuição de Linux mantida pela SUSE.
Switch	Dispositivo de rede que conecta e encaminha dados entre dispositivos em uma LAN.
Tablet	Computador portátil com tela sensível ao toque.
Thin interface	Interface simplificada para melhorar desempenho ou segurança.
Red Hat	Empresa que desenvolve o Red Hat Enterprise Linux.
Router	Dispositivo que encaminha pacotes de dados entre redes de computadores.
RUBY	Linguagem de programação dinâmica e de código aberto, focada em simplicidade e produtividade.
Users	Pessoas que utilizam sistemas de TI.
VMware	Empresa especializada em software de virtualização.
Web	Rede mundial de informações, acessível via internet.
Wi-Fi	Tecnologia de comunicação sem fio usada para conectar dispositivos a uma rede local.
Workloads	Conjuntos de tarefas executadas por um sistema.
Xen	Hypervisor de código aberto usado para virtualização.

1 CAPITULO I: INTRODUÇÃO

1.1 Contextualização

No contexto atual, a crescente demanda na busca por soluções tecnológicas eficientes e flexíveis tem impulsionado a adoção da *Cloud Computing* em diversos setores. Esta mudança de paradigma da computação está a transformar significativamente a forma como as organizações gerem e provêm recursos de *IT*, promovendo maior agilidade, escalabilidade e economia de custos (Jones & Brown, 2020). A migração para a *Cloud Computing* oferece às empresas a capacidade de se adaptarem rapidamente as mudanças na demanda de mercado, ao mesmo tempo em que reduzem significativamente os custos associados à manutenção da infraestrutura de *data center* físico (Johnson *et al.*, 2019).

Neste contexto, a implementação de uma *Infrastructure as a Service* (IaaS) privada em *Cloud Computing* através de uma ferramenta *Open Source* emerge como uma alternativa estratégica e vantajosa. Os autores Smith (2018) e Brown (2019) destacam os benefícios da adoção de soluções de *Open Source*, como o *OpenStack*, na construção de infraestrutura em *cloud* privada e pública. O *OpenStack* oferece uma plataforma para gestão de *data center* altamente flexível e ajustável a demanda, proporcionando às organizações maior controle sobre seus recursos de *IT* (Gupta & Sharma, 2020). Além disso, Chou (2010) ressalta a importância da virtualização na migração para a *Cloud Computing*, enfatizando que ferramentas *Open Source* são fundamentais para alcançar esses objetivos. Taurion (2009) observa que a *Cloud Computing* está revolucionando a forma como as organizações concebem e implementam suas infraestruturas de *IT*, destacando os benefícios em termos de flexibilidade, escalabilidade e eficiência operacional.

Esses pensamentos sustentam a ideia de que a adoção de ferramentas *Open Source* para implementação de uma IaaS privada em *cloud computing* é uma estratégia que proporciona múltiplos benefícios para as organizações. Ao adotar uma abordagem baseada em soluções *Open Source*, as empresas podem aproveitar os benefícios da *Cloud Computing*, ao mesmo tempo em que reduzem os custos e aumentam a flexibilidade e o controle sobre sua infraestrutura de *IT*.

1.2 Motivação

A escolha do tema para o presente trabalho surge da necessidade de explorar e compartilhar conhecimentos relacionados ao paradigma da *Cloud Computing*, mais especificamente no que diz respeito ao nível de provisão de serviços *IaaS*. A *Cloud Computing* tem sido objeto de estudo frequente entre acadêmicos, entidades governamentais, organizações privadas e diversas outras instituições interessadas no tema, com ênfase na sua relevância e impacto no cenário atual.

A implementação do modelo *Cloud Computing* em *data centers* oferece uma série de benefícios, incluindo a elasticidade na disponibilização de serviços, a redução de custos na aquisição e operação da infraestrutura, bem como a diminuição da emissão de poluentes associados à manutenção dos sistemas. Além disso, contribui para a preservação ambiental, um objetivo cada vez mais prioritário para organizações que buscam otimizar gastos e aumentar os lucros (Williams, 2010).

A motivação para desenvolvimento deste trabalho é fornecer um documento abrangente que compile as principais soluções *Open Source* disponíveis no mercado para empresas e instituições interessadas em implementar a *Cloud Computing (IaaS)*, com o propósito de facilitar o acesso a informações relevantes e promover a adoção de práticas sustentáveis e eficientes no ambiente empresarial.

O avanço tecnológico observado nas últimas décadas não tem precedentes, e a computação desempenha um papel fundamental nesse processo. Participar e contribuir por meio do desenvolvimento científico e tecnológico é a motivação primordial por trás deste trabalho, impulsionando a buscar soluções inovadoras e a promover o progresso no campo da *Cloud Computing*.

1.3 Definição do Problema

O problema abordado neste trabalho está centrado no estudo dos mecanismos de implementação de uma *IaaS* em *Cloud Computing*, utilizando plataformas de *Open Source* em um ambiente de *data center On-premises*. Conforme definido por Gil (2008, p. 33), o problema é entendido como uma questão não solvida que é objeto de discussão em qualquer domínio do conhecimento.

A decisão das organizações em adotar recursos *IT* na *Cloud* é motivada pelos benefícios organizacionais e técnicos que têm o potencial de aumentar a produtividade e a competitividade empresarial (Breitman & Virtebo, 2010). No entanto, esse processo de adoção da *Cloud Computing* apresenta uma série de requisitos e desafios, incluindo preocupações relacionadas à privacidade, interrupção de processos de negócios, interoperabilidade, integridade de dados, portabilidade de aplicativos, continuidade de negócios, produtividade da equipe e questões de segurança.

Alem disso, a ampla disponibilidade de serviços na *Cloud*, juntamente com a diversificação das necessidades das organizações, aumentou o nível de complexidade das decisões relacionadas à adoção de uma *IaaS*. As soluções *IT* tradicionais enfrentam limitações em termos de escalabilidade, agilidade e economia, o que ressalta a necessidade de explorar alternativas que possam superar essas restrições dentro do modelo de *data center* convencional.

Portanto, o problema central deste estudo reside na investigação dos desafios e das soluções associadas à implementação de uma infraestrutura na *Cloud* privada baseada em plataformas *Open Source*, visando superar as limitações das abordagens tradicionais e atender às demandas crescentes por flexibilidade, escalabilidade e eficiência operacional.

1.4 OBJECTIVOS

1.4.1 Objetivo Geral

Propor a implementação de uma *Infrastructure As A Service (IaaS)* em *Cloud Computing* através uma ferramenta *Open Source* para à ServiSIS Moçambique, SA.

1.4.2 Objetivos Específicos

- Definir conceitos relevantes sobre a *Cloud Computing*, as características essenciais, os serviços, modelo de implementação e as tecnologias que contribuíram para o seu surgimento.
- Descrever as principais ferramentas *Open Source* para provisão de serviços *IaaS* em *Cloud Computing* disponíveis no mercado.
- Descrever a situação atual e os constrangimentos enfrentados pela ServiSIS na administração do seu *data center on-premises*.
- Desenvolver um protótipo aplicável como proposta de solução para o problema identificado.

1.5 Metodologias

A metodologia de pesquisa aplicada neste trabalho foi de carácter exploratório, com abordagens qualitativas. Como o propósito de definir e resolver de problemas e explorar novas áreas de conhecimento do tema do trabalho.

Para alcançar os objetivos do trabalho seguimos a implementação da metodologia abaixo que é conduzida em cinco fases principais, nomeadamente:

- ✓ Revisão da literatura
- ✓ Entrevistas com o administrador do *data center*
- ✓ Encontros (presencias ou remotos) com o supervisor
- ✓ Consulta em páginas credíveis da *Internet* e outros documentos relevantes para concretizar este trabalho;
- ✓ Revisão de trabalhos que abordam a mesma temática

CAPITULO I – INTRODUÇÃO

Neste capítulo enfatizamos a contextualização do trabalho, o problema identificado, os objetivos geral e específicos, as metodologias utilizadas, e por fim as questões de pesquisa.

CAPITULO II – REVISÃO DE LITERATURA

Neste capítulo é apresentada a revisão da literatura que será utilizada como base teórica para o desenvolvimento do trabalho.

CAPITULO III – CASO DE ESTUDO

Neste capítulo é feita uma descrição do caso de estudo, uma análise da situação atual e o levantamento dos constrangimentos atuais do local de trabalho.

CAPITULO IV – DESENVOLVIMENTO DO TRABALHO

Neste capítulo são demonstradas as etapas necessárias para a implementação de um modelo básico de infraestrutura *cloud computing*, onde apresenta a configuração que pode ser realizada na plataforma *Openstack*.

CAPITULO V – DISCUSSÃO DE RESULTADOS

Neste capítulo é onde são discutidos os resultados obtidos durante o desenvolvimento do trabalho.

CAPITULO VI – CONCLUSÕES E RECOMENDAÇÕES

Neste capítulo são apresentadas as conclusões e recomendações do trabalho, de modo que atreves do referencial teórico se possa satisfazer o objetivo geral e os objetivos específicos.

CAPÍTULO VII: BIBLIOGRAFIA

Neste capítulo são mostradas todas fontes que permitiram a elaboração do trabalho.

ANEXOS

Nesta secção são apresentadas as configurações feitas na solução implementada, guião de entrevista e outros.

2 CAPITULO II: REVISÃO DE LITERATURA

2.1 Definição de *Cloud Computing*

O termo *cloud* geralmente refere-se a uma rede ou à *internet*. É uma tecnologia que usa servidores remotos na *Internet* para armazenar, gerir e aceder dados *online* sem que haja necessidade de utilizar unidades locais. Os dados podem ser quaisquer tais como arquivos ou pastas, imagens, documentos, áudio, vídeo, entre outros formatos de representação da informação (Nikhila, 2022).

A *cloud computing*, por ser um paradigma considerado novo, possui diversas definições e ainda não tem um conceito padrão. A definição mais citada é a proposta pelo *NIST*, na qual segundo (Mell & Grance, 2011)

Cloud Computing é um modelo para permitir o acesso conveniente, sob demanda, a um conjunto de recursos computacionais configuráveis (por exemplo: redes, servidores, armazenamento, aplicações e serviços) que podem ser rapidamente provisionados e liberados com um esforço mínimo de administração ou interação com o provedor de serviços.

Segundo Veras (2012), *cloud computing* significa mudar fundamentalmente a forma de operar a infraestrutura *IT*, saindo de um modelo baseado em aquisição de equipamentos para um modelo baseado em aquisição de serviços. Simplificadamente, *cloud computing*, refere-se à ideia de utilizar, em qualquer lugar e independente da plataforma, as mais variadas aplicações através da *internet* com a mesma facilidade de tê-las instaladas em nossos próprios computadores ou outros dispositivos (Baird, 2019).

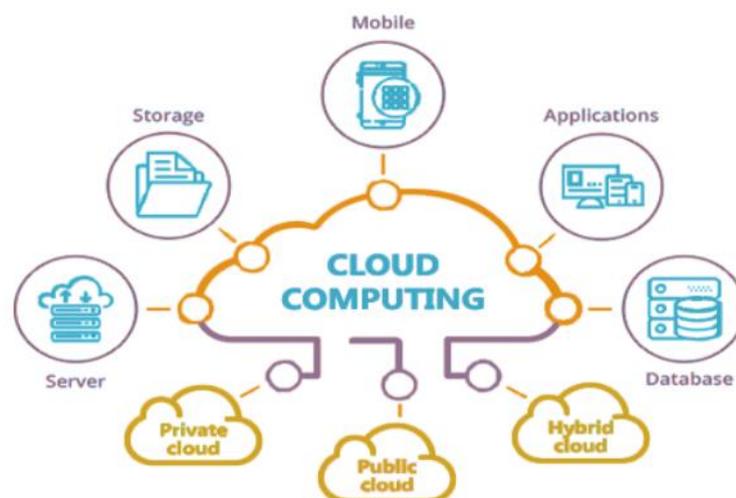


Figura 1: Diagrama básico da *cloud computing*
Fonte: (Baird, 2019)

2.2 Modelo Conceptual *Cloud Computing*

Os autores Mell e Grance (2011), descrevem a *cloud computing* como um modelo que prove flexibilidade e confiabilidade no acesso a recursos como: rede, servidores, aplicações e serviços. A figura 2 usa o modelo arquitetural para descrever o modelo conceptual de definição da *cloud computing*:

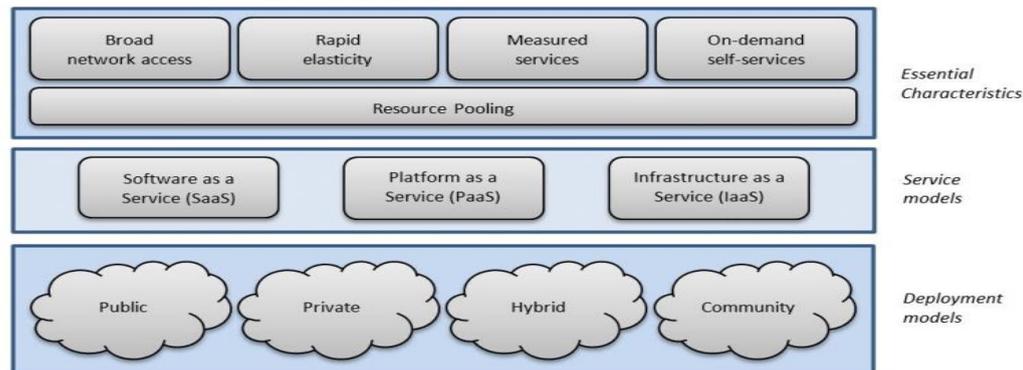


Figura 2: Modelo arquitetural da *cloud computing*
Fonte: (Mell e Grance, 2011)

2.2.1 Características Essências da *Cloud Computing*

De acordo com o modelo acima, a *cloud computing* é constituída por cinco características essenciais (Mell & Grance, 2011):

- Autoaprovisionamento de recursos sob demanda (*On-demand self-service*),
- Amplo acesso à rede (*Broad network access*),
- Recursos compartilhados (*resource pooling*)
- Rápida Elasticidade (*Rapid elasticity*),
- Serviço mensurado (*Measured service*).

2.2.1.1 Autoaprovisionamento de recursos sob demanda

Essa característica habilita o consumidor ter a facilidade de aceder recursos computacionais, como tempo do servidor e armazenamento em rede, conforme necessário automaticamente sem intervenção por parte do fornecedor dos serviços.

2.2.1.2 Amplo Acesso à rede

Nessa característica os recursos providos pela *cloud computing* devem ser acessíveis através de diferentes tipos de equipamentos de acesso a rede, como celular, *tablet*, *laptops*, computadores, plataformas *web* entre outros.

2.2.1.3 Recursos compartilhados

Nessa característica os recursos computacionais do provedor são agrupados para atender múltiplos consumidores usando diferentes recursos físicos e virtuais, dinamicamente atribuídos e reatribuídos de acordo com a demanda do consumidor. Esses recursos incluem armazenamento, processamento, memória e largura de banda.

2.2.1.4 Elasticidade rápida

Nessa característica os recursos podem ser elasticamente alocados e liberados, em alguns momentos para prover rapidamente mais ou menos recursos de forma automática ao consumidor, os recursos disponíveis para realizar o provisionamento muitas vezes parecem ser ilimitados e pode ser feito em qualquer quantidade a qualquer momento.

2.2.1.5 Serviço mensurado

Nessa característica os sistemas usados para gerir a *Cloud Computing* automaticamente controlam e otimizam o uso dos recursos, alavancando a capacidade de medição em algum nível de abstração apropriado para o tipo de serviço (por exemplo, processamento, armazenamento, largura de banda e contas de usuários ativos). O uso dos recursos pode ser monitorado, controlado e reportado, oferecendo transparência tanto para o provedor quanto para o consumidor do serviço utilizado.

2.2.2 Modelo de Serviços *Cloud Computing*

O modelo de serviços define o padrão de arquitetura aplicável para soluções *cloud computing* e também expõe uma breve especificação de serviços compatíveis com cada camada. O modelo de serviços encontrado na literatura é constituído por três camadas (Mell & Grance, 2011):

- Infraestrutura como Serviço (*Infrastructure as a Service - IaaS*)
- Plataforma como Serviço (*Platform as a Service - PaaS*)
- Software como Serviço (*Software as a Service - SaaS*)

2.2.2.1 Infraestrutura como Serviço (IaaS)

A *IaaS* é a camada fundamental para *cloud computing* na qual os demais modelos de serviços são hospedados. Este modelo de serviços relaciona-se com a capacidade que um provedor tem de oferecer uma infraestrutura de processamento e armazenamento de forma transparente (Borges et al, 2011).

Segundo Mell e Grance (2010), a *IaaS* consiste em,

Fornecer processamento, armazenamento, redes e outros recursos fundamentais recursos de computação onde o consumidor é capaz de implantar e executar *software*, que pode incluir sistemas operativos e aplicativos. O consumidor não gerência ou controla a infraestrutura *cloud* subjacente, mas tem controlo sobre sistemas operativos, armazenamento e aplicativos implantados; e possivelmente controle limitado de componentes de rede seleccionados (por exemplo, *firewalls* e *host*).

Isso permite aos utilizadores finais o acesso a vários recursos computacionais, como armazenamento, provisão, processamento ou redes. Ele permite que o usuário execute *software* arbitrário, incluindo aplicações e SO. O usuário tem total controlo de armazenamento e SO. Ele também tem um controle sobre as aplicações de um certo limite de controlo sobre as instalações de rede.

Benefícios da *IaaS*

Segundo Borges (2011), algumas das vantagens do uso da *IaaS* são:

- ✓ A redução de investimentos em *hardware*, bem como a preocupação com a depreciação dos mesmos;
- ✓ Flexibilidade para ampliar e reduzir a capacidade de processamento e armazenamento.
- ✓ Eliminação de custos com segurança e manutenção da infraestrutura local;
- ✓ Liberação de espaço físico na empresa; e
- ✓ Otimização do desempenho;

2.2.2.2 Plataforma como Serviço (*PaaS*)

A *PaaS* é a camada intermediária do modelo de serviços, sendo composta por *hardware* virtual disponibilizado como serviço na qual os serviços prestados pelo provedor da *cloud* são disponibilizados através de *Frameworks* com a finalidade de proporcionar ao consumidor um ambiente virtual de desenvolvimento, integração e implementação de teste de aplicações (Borges, 2011).

Segundo (Mell e Grance (2010), *PaaS* consiste,

A capacidade fornecida ao consumidor de implantar na infraestrutura de nuvem criada ou adquirida pelo consumidor aplicativos criados usando linguagens de programação, bibliotecas, serviços e ferramentas suportadas pelo provedor. O consumidor não gerencia ou controla a infraestrutura *cloud* subjacente, incluindo rede, servidores, sistemas operativos, ou armazenamento, mas tem controle sobre os aplicativos implantados e possivelmente definições de configuração para o ambiente de hospedagem de aplicativos.

Isso permite que o consumidor implemente na *cloud IaaS*, os aplicativos criados especificamente para as necessidades do consumidor, a linguagem de programação e suas ferramentas. No entanto, o consumidor não pode usar e gerir os recursos fundamentais infraestrutura, como servidores, redes, armazenamento ou SO.

2.2.2.3 Software como Serviço (SaaS)

O *SaaS* é a camada mais próxima do utilizador final, ela é composta por aplicativos que são executados no ambiente *cloud*. Neste modelo os principais serviços prestados pelo provedor de *cloud* são de disponibilização de *softwares* com a finalidade de proporcionar ao consumidor *softwares* de propósito específico (Borges, 2011).

Segundo (Mell e Grance (2010), *SaaS* consiste,

A capacidade oferecida ao consumidor de usar os aplicativos do provedor em execução em uma infraestrutura *cloud*. Os aplicativos são acessíveis a partir de vários dispositivos clientes por meio de um *thin interface* do cliente, como um navegador da *Web* (por exemplo, e-mail baseado na *Web*) ou uma *interface de programa*. O consumidor não gere ou controla a *cloud* subjacente infraestrutura, incluindo rede, servidores, sistemas operacionais, armazenamento ou até mesmo capacidades de aplicativos individuais, com a possível exceção de usuários limitados configurações de aplicativos específicos.

As aplicações são acessíveis a partir de vários dispositivos clientes que têm acesso a *cloud* através de um navegador *Web*. O controle e gestão da infraestrutura, que inclui SO, armazenamento e servidores, são feitos pelo provedor do serviço.

2.2.3 Modelos de Implementação de Serviços Cloud Computing

Os modelos de implementação independentemente do modelo de serviços utilizado (*SaaS*, *PaaS*, *IaaS*), dependem das necessidades das aplicações que serão

implementadas. A restrição ou abertura de acesso depende do processo de negócios, do tipo de informação e do nível de acesso desejado. A infraestrutura pode ser implementada de quatro formas, em funções das necessidades do cliente.

De acordo com Mell e Grance (2011), um ambiente de *Cloud Computing* pode ser implementado seguintes modelos de implementação:

- *Cloud Privada (Private Cloud)*
- *Cloud Publica (Public Cloud)*
- *Cloud Híbrida (Hybrid Cloud)*
- *Cloud Comunitária (Community Cloud)*

A figura 3 representa o modelo de implantação *cloud* que é usado regularmente. Embora todos os quatro sistemas sejam adaptados para a satisfação do usuário e acessibilidade de dados, eles têm recursos diferentes que diferem por causa de taxas anuais, localização de onde os dados são armazenados, privacidade e acessibilidade multiusuário (Bora & Ahmed, 2013).

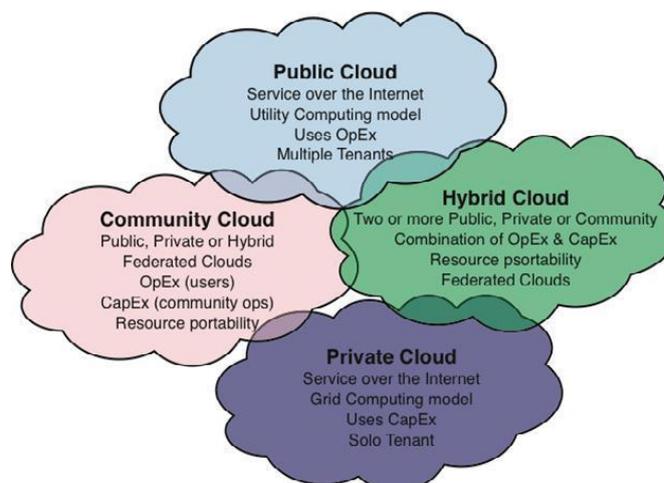


Figura 3: Modelos de Implementação de Serviços Cloud Computing
Fonte:(Bora & Ahmed, 2013)

2.2.3.1 *Cloud Privada*

A *cloud Privada* é projetada para prover uma infraestrutura de uso exclusivo de uma determinada instituição e somente os funcionários dessa instituição tem acesso. A *cloud* privada utiliza uma infraestrutura privada e fornecer serviços aos usuários internos. A gestão dos serviços *cloud computing* pode ser administrada pela própria organização ou por terceiros e pode estar dentro ou fora da organização.

2.2.3.2 Cloud pública

A *cloud* pública é projetada para prover uma infraestrutura destinada ao uso sem restrições ao público em geral. Foi a primeira infraestrutura em *cloud computing* a ser desenvolvida e oferecida aos usuários. Seu proprietário e administrador pode ser uma empresa, órgão governamental ou instituição acadêmica, sendo sediada nas instalações do provedor de serviços *cloud computing*.

2.2.3.3 Cloud Comunitária

A *cloud* comunitária é destinada ao uso de uma determinada comunidade de usuários que têm as mesmas necessidades. Todos os três tipos apresentados acima combinados ajudam a criar uma infraestrutura *cloud* comunitária, projetada para as necessidades de uma determinada indústria. A infraestrutura *cloud* é compartilhada entre poucas organizações que possuem a mesma missão e requisitos de segurança. A administração da infraestrutura pode ser de uma ou mais organizações ou terceiros, podendo estar localizada dentro ou fora das organizações da comunidade.

2.2.3.4 Cloud Híbrida

A *cloud* Híbrida é uma combinação de duas ou mais *clouds* em infraestruturas diferentes. Podem ser *clouds* privada, pública ou de uso comunitário com diferentes entidades que compartilham uma tecnologia padrão(Sousa *et al*, 2009). Como o próprio nome já diz, este modelo é mais flexível que os outros, pois pode se aproveitar da qualidade de alto controle do modelo de *cloud* Privada e também da qualidade de facilidade de expansão do modelo de *cloud* Pública, onde as atividades não críticas são realizadas pela *cloud* pública e as críticas serão realizadas pela *cloud* privada.

2.2.3.5 Principais Diferença entre Modelos de Implementação de Serviços Cloud Computing

Em função do cada modelo de implementação, a tabela 1 mostra a diferença entre os modelos de implementação de serviços *cloud computing* em função de parâmetro:

Parâmetro	Cloud Publica	Cloud Privada	Cloud Híbrida	Cloud Comunitária
Provedor	Provedor de serviços	Empresa/Terceiros	Empresa/Terceiros	Comunidade (terceiros)
Usuários	Publico em geral	Usuários selecionados	Usuários selecionados	Membros da comunidade
Acesso	Internet	Internet, VPN	Internet, VPN	Internet, VPN
Proprietário	Provedor	Organização	Organização	Comunidade

Tabela 1: Principais Diferenças entre os modelos de implementação da cloud computing

Fonte: (Javatpoint, 2024)

2.3 Papeis na *Cloud Computing* Mediante o Modelos de Serviços

Segundo Catteddu e Hogben (2009), as responsabilidades do provedor de serviços e do consumidor dos mesmos na *cloud* vária drasticamente entre os diversos modelos de serviços, com maior exigência de responsabilidade para o consumidor no modelo *IaaS*. Segundo a sua abordagem, nas tabelas abaixo, a divisão de responsabilidades do contratante e do provedor segundo o modelo do serviço de acordo com o *SLA* em função das necessidades do cliente.

<i>IaaS</i>	
Consumidor (Gestores de Sistemas)	Provedor (Hardware)
<ul style="list-style-type: none"> ➤ Gestão do serviço de autenticação e atualização. ➤ Configuração da plataforma de segurança. ➤ Manutenção da plataforma de segurança. 	<ul style="list-style-type: none"> ➤ Suporte à infraestrutura física. ➤ Segurança da infraestrutura física. ➤ Hospedagem de Sistemas (virtualização)
<i>PaaS</i>	
Consumidor (Desenvolvedores)	Provedor (Frameworks)
<ul style="list-style-type: none"> ➤ Gestão do serviço de autenticação ➤ Suporte a infraestrutura física ➤ Segurança da infraestrutura física ➤ Hospedagem de Sistemas (Virtualização) 	<ul style="list-style-type: none"> ➤ Segurança da infraestrutura física. ➤ Gestão e atualização dos SO. ➤ Configuração e Manutenção da plataforma de segurança. ➤ Monitoramento do sistema.
<i>SaaS</i>	
Consumidor (Usuário Final)	Provedor (Aplicações)
<ul style="list-style-type: none"> ➤ Conformidade com a legislação de proteção de dados de clientes. ➤ Gerir serviço de autenticação. 	<ul style="list-style-type: none"> ➤ Suporte e Segurança da infraestrutura física. ➤ Gestão e atualização dos SO. ➤ Configuração e Manutenção da plataforma de segurança.

Tabela 2: Divisão de responsabilidades entre cliente e fornecedor *Cloud*.
Fonte: (Catteddu & Hogben, 2009)

2.4 Análise de Risco de Cada Modelo de Serviços

Segundo Stokes (2013), Em função das especificações do modelo de serviços em *Cloud*, seja *IaaS*, *PaaS* ou *SaaS* constata-se que o modelo com menor exposição a risco é o *IaaS*. Em contrapartida, temos o modelo *SaaS*, que exige muito por parte do fornecedor e da equipa de segurança *IT*. Deve-se aplicar regras de segurança restritas, como encriptação nas comunicações entre o fornecedor da *cloud* e a organização.

2.4.1 Modelo de implementação

Os modelos de implementação de serviços em *cloud*, também apresentam níveis de risco diferenciados. O modelo *cloud* privado é o que menor risco apresenta de perda de dados ou ataques informáticos, e o modelo *cloud* pública é o que merece um maior cuidado na sua implementação relativamente a proteção de dados (Stokes, 2013).

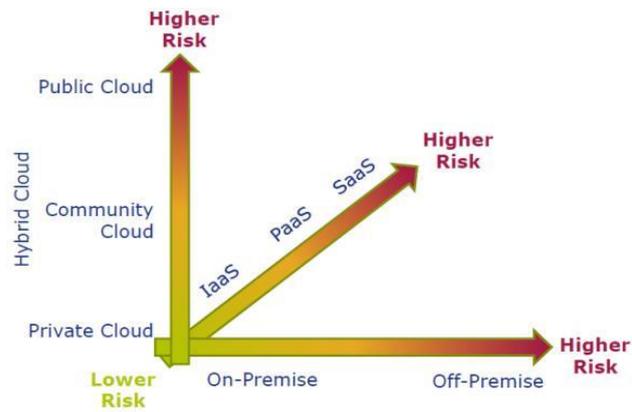


Figura 4: Cloud Computing compatível x Gestão dos riscos
Fonte: (Stokes, 2013).

2.4.2 Benefícios e Limitações da Cloud Computing

A *cloud computing* nos últimos anos está a evoluir como nunca antes, com empresas de todas as formas e tamanhos a adaptarem essa nova tecnologia. Especialistas do setor acreditam que essa tendência continuará a crescer e desenvolver mais nos próximos anos. Embora a *cloud computing* seja indubitavelmente benéfica para empresas, ela tem suas vantagens e desvantagem em função do modelo de empresa (Esis & Bent, 2021).

2.4.2.1 Principais Benefícios da Cloud Computing

A *cloud computing* se usado corretamente, pode beneficiar muito a forma de trabalhar com dados na cloud independentemente do tipos de negócios. Abaixo são mencionadas 6 vantagens desta tecnologia (Esis & Bent, 2021):

Benefícios	Discrição
Custo reduzido	Devido ao modelo de pagamento por uso ou assinatura, as organizações não precisam investir antecipadamente em infraestrutura <i>IT</i> . Para os provedores da <i>cloud</i> , os custos são mais baixos devido à economia de escala e princípio de <i>multi-tenancy</i> , nenhum 'espaço' é deixado inutilizado.
Automatização	Atualizações automáticas, <i>patches</i> de segurança e <i>backups</i> não são mais uma preocupação do cliente. O pessoal <i>IT</i> não precisa se preocupar em manter <i>software</i> atualizado.
Mais mobilidade	Dados e aplicações podem ser acedidos pela Internet a partir de qualquer tipo de dispositivo de computação <i>smart</i> , a qualquer hora e em qualquer lugar.
Recursos compartilhados	Os clientes compartilham recursos, permitindo que organizações menores tenham acesso a instalações de <i>IT</i> em escala corporativa e serviços de apoio.
Agilidade e escalabilidade	As empresas podem escalar sua infraestrutura de <i>IT</i> para cima ou para baixo em função da demanda.
Foco no negócio principal	A maioria das empresas, especialmente as empresas novas no mercado, não precisam possuir e operar a infraestrutura <i>IT</i> .

Tabela 3: Benefícios da Cloud Computing
Fonte: (ESIS & BENT, 2021)

2.4.2.2 Principais Limitações da *Cloud Computing*

Apesar dos benefícios da *cloud computing*, também existem desvantagens. Negócios, especialmente os menores, devem estar cientes das limitações antes de usar esta tecnologia (Esis & Bent, 2021).

Limitações	Discrição
Acesso à Internet	Normalmente, o acesso à Internet significa acesso à nuvem pública. Se Internet não é confiável, isso é um risco.
Segurança	O <i>data center</i> em <i>cloud</i> pode apresentar alta segurança e ser altamente gerenciado, mas também há baixa segurança dos dados mal geridos. Nem sempre é fácil diferenciar os dois.
Privacidade	No caso de ofertas de nuvem pública ou híbrida, pode ser incerto onde seus dados são armazenados fisicamente. Isso pode ser problemático em luz da variada legislação nacional e internacional sobre privacidade e Proteção de dados. É difícil saber com certeza quem pode aceder seu dados.
(SLA)	É importante garantir que seu SLA com o provedor da <i>cloud</i> permite flexibilidade e escalabilidade. Se isso não acontecer, um importante benefício da <i>cloud</i> é perdido.
Integração	Integração O SLA deve permitir a integração de dados e fluxos de trabalho entre todo o seu ambiente de IT. Isso deve ser feito de maneira segura e por um preço razoável.
Bloqueio do Provedor	A migração de serviços para <i>cloud</i> pode ser problemática e arriscada. Isso poderia significar que o negócio fica com um provedor que não atende suas necessidades, apenas para evitar a migração e um possível atendimento interrupção.

Tabela 4: Limitações da *Cloud Computing*
Fonte: (Esis & Bent, 2021)

2.5 Tecnologias Relacionadas a *Cloud Computing*

A *Cloud Computing* não é por si só uma tecnologia existem diferentes inovações trabalhando para torná-la confiável, adaptável e utilizável. Essas tecnologias possuem características e propriedades sustentam o crescimento da *Cloud Computing*.

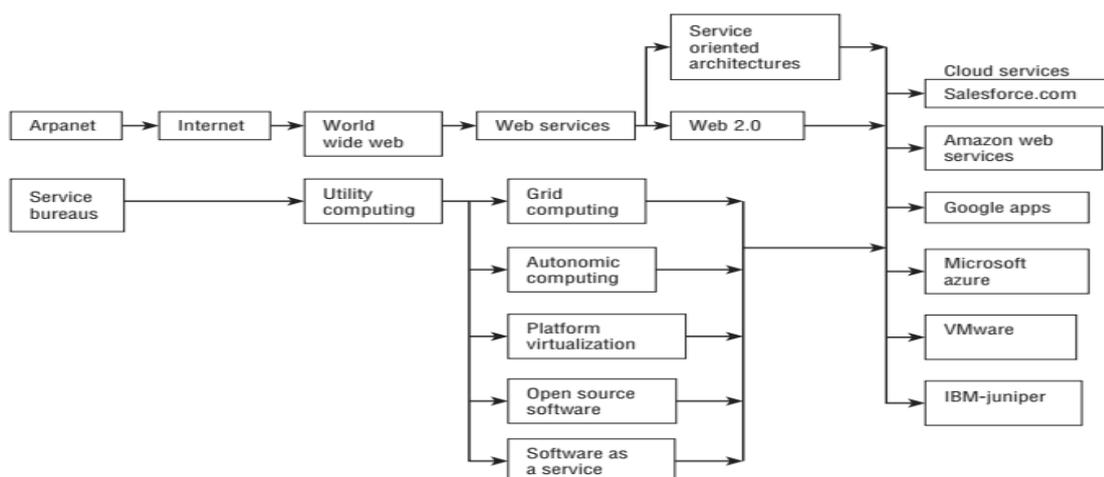


Figura 5: Origens da computação em nuvem
Fonte: (Krutz e Vines, 2010, p.5)

Existem diversas tecnologias por trás da origem da *Cloud Computing*, tornando-a flexível, confiável e utilizável (Kruz & Vines, 2010). Essas tecnologias estão listadas abaixo:

- *Web 2.0*
- Arquitetura Orientada a Serviços (SOA)
- Computação em Grade
- Computação Utilitária
- Virtualização

2.5.1 WEB 2.0

É a interface através da qual os serviços *Cloud Computing* interagem com os clientes. É por causa da Web 2.0 que temos páginas web interativas e dinâmicas. Isso também aumenta a flexibilidade entre as páginas da web. Exemplos populares de web 2.0 incluem Google Maps, Facebook, Twitter, e outros (Nikhila, 2022).

2.5.2 Arquitetura Orientada a Serviços (SOA)

A SOA ajuda a usar aplicações como um serviço para outras aplicações, independentemente do tipo de fornecedor, produto ou tecnologia. Portanto, é possível a troca de dados entre aplicações de diferentes fornecedores sem programação adicional ou fazer alterações nos serviços (Tutorialspoint, 2024).

Para JavaTpoint (2023), as vantagens de usar SOA *derivam da* facilidade de manter, independente da plataforma é altamente escalável. Aumentando a usabilidade, manutenção e o valor das aplicações.

2.5.3 Computação em grade(Grid Computing)

Grid Computing refere-se à computação distribuída na qual um grupo de computadores de vários locais são conectados entre si para alcançar um objetivo comum. Esses recursos computacionais são heterogêneos e Geograficamente disperso. *Grid Computing* divide tarefas complexas em pedaços menores. Esses pedaços menores são distribuídos para CPUs que armazenado dentro de grelhas (Tutorialspoint, 2024).

2.5.4 Computação utilitária (Utility computing)

A *Utility computing* é baseada no modelo *Pay per Use*. Oferece recursos computacionais sob demanda como um medidor serviço. A computação em nuvem, a computação em grade e os serviços gerenciados de TI são baseados no conceito de computação utilitária (Nikhila, 2022).

2.5.5 Virtualização

A virtualização é um método de divisão lógica dos recursos computacionais até o ponto em que uma máquina física pode suportar uma quantidade virtualmente ilimitada de máquinas virtuais e lógicas. Este conceito implica criar uma representação virtual de algo, e é um componente fundamental pra a *cloud computing*, permitindo a criação de VMs para abstrair as características físicas do *hardware*. Essas VMs podem emular vários SO em um único servidor, fornecendo uma camada de abstração dos recursos físicos do equipamento, com a alocação de *hardware* virtual para cada sistema (Esis & Bent, 2021).

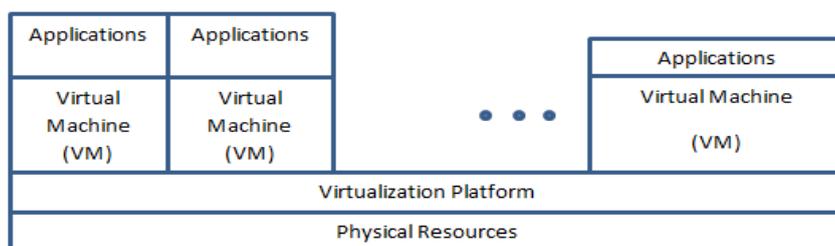


Figura 6: Sample Architecture VMs
Fonte: (Saleem & Rajouri,2017)

2.5.5.1 Tipos de Virtualização

Quanto aos tipos destacam-se a virtualização de: aplicações, procedimentos, rede, armazenamento e acesso, cuja descrição é dada na tabela 1 (Turel, 2013):

Tipos de Virtualizacao	Discrição
Aplicações	Permite que as aplicações sejam executados em diferentes SO e plataformas de <i>hardware</i> .
Processamento	Faz um sistema parecer muitos, ou muitos parecerem um
Rede	Apresenta uma visão artificial da rede que difere da realidade física
Armazenamento	Permite que muitos sistemas compartilhem os mesmos dispositivos de armazenamento, permite ocultar a localização de sistemas de armazenamento e outros.
Acesso	Permite o acesso a qualquer aplicativo através de qualquer dispositivo

Tabela 5: Diferentes formas de virtualização
Fonte: (Turel, 2013)

2.5.5.2 Tecnologias De Virtualização

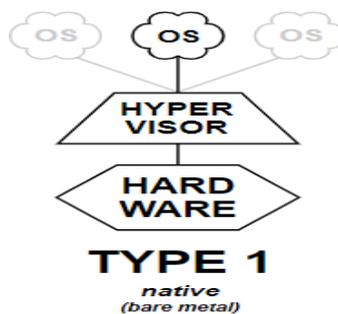
➤ Hypervisor

É um programa que permite que vários SOs compartilhem um único *host* de *hardware*. Na verdade, é um programa de baixo nível que fornece acesso aos recursos do sistema para as *VMs* (Saleem & Rajouri,2017).

➤ Hypervisor tipo 1(Type-1 hypervisor)

Este tipo de hypervisores também conhecidos como nativos ou bare-metal, operam diretamente sobre o hardware físico de um servidor, sem a necessidade de um SO host intermediário. Isso resulta em maior eficiência e desempenho na virtualização, já que os hypervisores controlam diretamente os recursos de hardware e gerem os SOs guest de forma independente. Estes tipos de *hypervisores* oferecem suporte à virtualização de hardware 3 (Saleem & Rajouri,2017). Isso é mostrado na figura 2.

Os hypervisores do Tipo 1 são instalados diretamente no *hardware* físico de um servidor.São exemplos de Hypervisores do tipo-1 o ESXi, *Oracle VM Server* for x86, Hyper-V, LXC, KVM(Kernel-based Virtual Machine) e Docker, pois são instalados e executados sobre um SO hospedeiro.



Type-1 hypervisor
Fonte: (autor).

➤ Type 2 hypervisor

É um tipo de hypervisor cliente que fica sobre o SO. Ele não pode ser executado até que o SO já esteja em execução. A virtualização de software é realizada neste hypervisor porque depende do SO. Se o SO falhar, todos os usuários finais serão afetados, conforme mostrado na figura 3 (Saleem & Rajouri, 2017).

Para os autores Saleem e Rajouri (2017), esses *hypervisores* são basicamente como aplicativos instalados no SO *guest*. O, *VMware Fusion*(para Mac), *Virtual Server 2005 R2*, *VMware Workstation*, *Oracle VirtualBox*, *Microsoft Virtual PC*, *QEMU*, *Docker Desktop*, *Windows Virtual PC* e *VMware workstation* estão na categoria deste tipo de *hypervisor*. A virtualização de *software* oferece melhor compatibilidade de *hardware* do que a virtualização *bare metal*, porque o SO é responsável pelos *drivers* de *hardware*, e não pelo *hypervisor*.

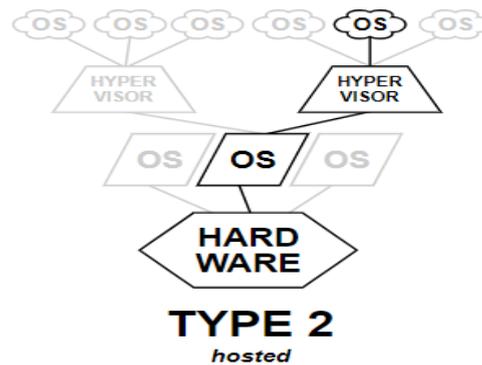


Figura 7: Type-2 hypervisor
Fonte:(autor)

2.5.5.3 Virtualização completa(*Full Virtualization*)

A *Full Virtualization* é uma técnica na qual uma *VM* é instalada e executada dentro de outra. Essa abordagem suporta a execução de diferentes SO em um único *hardware*, embora exija uma configuração específica. O *hypervisor* interage diretamente com a *CPU* e o disco do servidor físico, criando um espaço isolado para cada máquina virtual, como ilustrado na Figura 4. Nesse modelo de virtualização, cada servidor virtual opera independentemente dos outros servidores virtuais que possam estar em execução na mesma máquina física (Saleem & Rajouri, 2017).

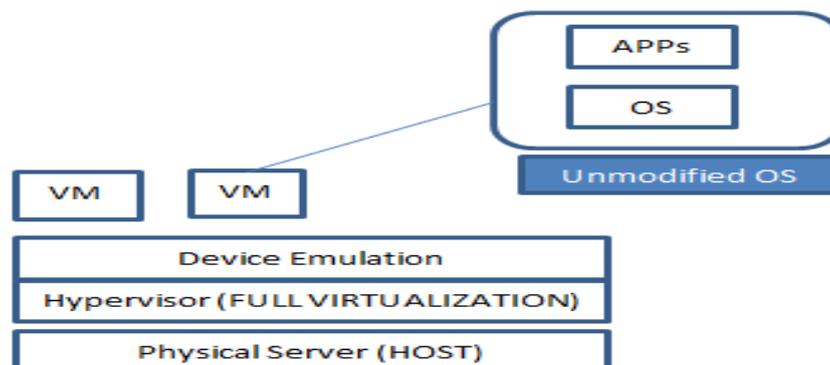


Figura 8: Full Virtualization
Fonte: (Saleem & Rajouri,2017).

2.5.5.4 Para virtualização

A Para virtualização é a categoria de virtualização de *CPU*, onde o SO convidado está ciente de que está sendo virtualizado. Devido a esta informação antecipada o SO *guest* pode causar curto-circuito em seus *drivers* para minimizar a sobrecarga de comunicação com dispositivos físicos. Essa técnica de virtualização elimina a desvantagem da virtualização completa (Saleem & Rajouri, 2017).. Isso é mostrado na figura 5.

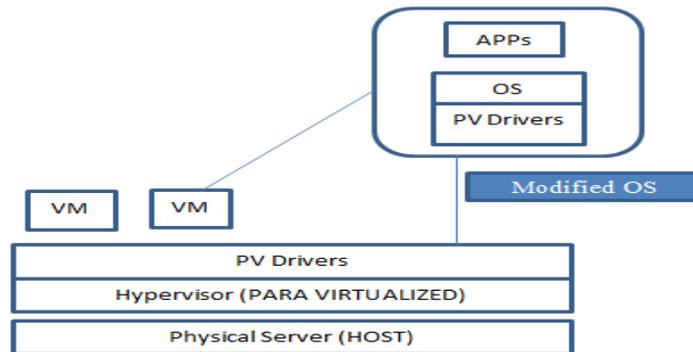


Figura 9: Para Virtualization
Fonte: (Saleem & Rajouri,2017).

Diferença entre Full Virtualization e Paravirtualization

Para Vazhakka (2022), as diferenças entre a *FullVirtualization* e *Paravirtualization* são as seguintes:

Full Virtualization	Paravirtualization
A virtualização completa é menos segura.	Enquanto a Paravirtualização é mais segura que a Virtualização Completa.
A virtualização completa é mais portátil e compatível.	A paravirtualização é menos portátil e compatível.
A virtualização completa é lenta do que a paravirtualização em operação.	A paravirtualização é mais rápida em operação em comparação com a virtualização completa.
Ele fornece o melhor isolamento.	Ele fornece menos isolamento em comparação com a virtualização completa.

Tabela 6: Diferença entre Virtualização completa e paravirtualização
Fonte:Adoptado de (Vazhakka, 2022).

2.5.5.5 Benefícios da virtualização

Segundo Turel (2013), a virtualização de servidores e máquinas é a principal vantagem da *cloud computing*. De um recurso de *hardware* podem ser criados dispositivos virtuais quando possível. Eles podem ser oferecidos aos usuários, independentemente de sua localização física.

Alguns dos benefícios da virtualização na *cloud computing* são (Turel, 2013):

- **Economiza dinheiro:** os servidores com a virtualização, podemos transformar um servidor de finalidade única em multitarefa e transformar vários servidores em um *pool* de computação que pode se adaptar com mais flexibilidade às *work loads* em constante mudança.
- **Economiza energia:** virtualização reduz o número de servidores físicos, reduzindo a energia necessária para ligá-los e resfriá-los.
- **Economiza tempo:** Com menos servidores, podemos gastar menos tempo nas tarefas necessárias para a manutenção do servidor. Ao agrupar vários dispositivos de armazenamento em um único dispositivo de armazenamento virtual, a execução de tarefas como *backup* e recuperação é feita com mais facilidade e rapidez.

2.5.5.6 Riscos da Virtualização

A virtualização em *cloud* ajuda a economizar recursos de *hardware*, mas, por outro lado, traz alguns fatores de risco definidos abaixo (Turel, 2013).

- A segurança em *cloud*, especialmente ao proteger sistemas de alta segurança, requer monitoria de segurança extensivo e controles de identidade e gestão de acesso.
- A tecnologia de virtualização ainda está a amadurecer. Ferramentas e controles adicionais são necessários e *hypervisores* devem ser tratados como SO e protegidos como tal.
- Controles de administrador são necessários. A definição de funções personalizadas, é necessária para garantir que a permissão seja restrita para tarefas que não devem ser executadas por funcionários específicos.

2.6 Comparação da Arquitectura *Cloud* e Arquitectura *On-Premise*

Segundo Silva (2014), em primeira análise podemos encontrar semelhanças entre ambas arquiteturas, a infraestrutura *cloud* e a infraestrutura *on-premises*, porque ambas dividem-se em três camadas. Na arquitetura *on-premises* a primeira camada, a infraestrutura, compreende todo o equipamento de rede, tais como servidores, *switches*, *load balance*, *routers*, equipamentos de proteção, rede física, etc, em tudo similar com à

camada *IaaS* do modelo *cloud*. A segunda camada é composta pelo *middleware*, onde se encontram os SOs, as plataformas de desenvolvimento e o diverso *software* de suporte à camada seguinte, que podemos mapear para *PaaS* no modelo *cloud*. Na camada de topo temos todo o *software* aplicativo em tudo idêntico ao *SaaS* no modelo *cloud Computing*.

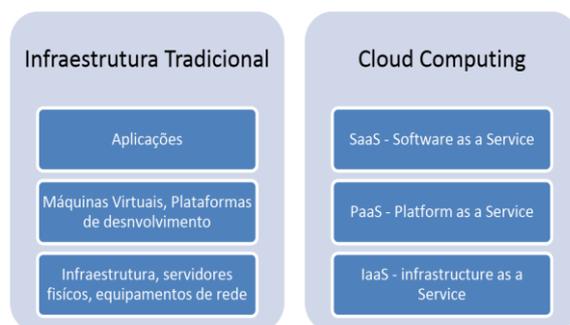


Figura 10: Infraestrutura *On-premise* vs *Cloud Computing*
Fonte:Adotado de (Silva,2014)

Segundo Silva para Ghosh e Hughes (2011), a hospedagem baseada em *cloud* possui muitas vantagens em relação à hospedagem *on-premises*, como melhor escalabilidade, facilidade de gestão, *multi-tenancy* e economia de custos.

Para Breitman e Virtebo, (2010), existem também importantes diferenças entre o modelo *cloud computing* e o *on-premises* conforme a tabela abaixo, daptada do relatório do grupo Gartner:

	Computação On-premises	Cloud Computing
Modelo De Aquisição	Hardware	aquisição de serviço
	Espaço físico	N/A
	Infraestrutura de instalação e funcionamento	N/A
Modelo De Negocio	Custo e depreciação de ativos	Pagamento conforme o uso
	Overhead administrativo(suporte, manutenção segurança do equipamento, refrigeração)	N/A
Modelo de Acesso	Rede interna	Internet, através de vários tipos de dispositivos com acesso.
	Intranet	N/A
Modelo Técnico	único	Escalável
	sem compartilhamento	Elástico
	Estático	Dinâmico
	Condomínio	N/A

Tabela 7: Infraestrutura *On-premise* vs *Cloud Computing*
Fonte: Breitman & Virtebo (2010).

2.6.1 Parâmetro de Classificação das Diferenças

As organizações que escolhem a infraestrutura de dados na *cloud* veem diferenças significativas daquelas que hospedam e gerem seus dados dentro de

instalações comerciais. Algumas das principais diferenças estão nos seguintes parâmetros Virginia, 2022:

Custo – os *data centers on-premises* exigem um investimento inicial considerável, enquanto as implantações em *cloud* são relativamente menos dispendiosas, de acordo com o *SLA* que orienta o *user* no modelo de pagamento conforme o uso.

Segurança de dados – *data centers on-premises* oferecem controle global sobre os dados; portanto, facilmente se pode implementar medidas de segurança avançadas. Entretanto, facilmente se perde o controle direto sobre os dados em implantações *cloud*, dependendo do tipo de serviço escolhido.

Implantação de recursos – No cenário *on-premises*, a maioria dos seus recursos é implantada nas instalações da empresa, enquanto os *data centers* em *cloud* têm a maior parte de seus recursos em um servidor de terceiros.

Escalabilidade e flexibilidade de operações – implantações *on-premises* são menos escaláveis e menos flexíveis do que os *data centers* na *cloud*.

2.7 Noções Básicas de *Data Center On-Premise*

Data center on-premise refere-se a infraestrutura *IT* localizada fisicamente no edifício de uma organização. A equipe *IT* é responsável por manter os servidores e realizar serviços de manutenção rotineira, como instalação e atualização de *software* de segurança. Uma *cloud* privada hospedada por um provedor de serviços em *cloud* terceirizado também pode ser considerada local. Aqui, a empresa mantém o *software* no local, enquanto o provedor de serviços supervisiona a *cloud* privada (Virginia, 2022).

2.7.1 Vantagens e Desvantagens da implantação On-Premises

Além de oferecer melhor controle sobre os dados empresariais, o armazenamento de dados no local oferece os seguintes benefícios e prejuízos (Virginia, 2022):

Vantagens	Desvantagens
<ul style="list-style-type: none">➤ Permite uma operação rápida e conveniente, mesmo sem acesso à internet.➤ Reduz os custos mensais de uso de uso dos equipamentos de IT.	<ul style="list-style-type: none">➤ Requer suporte interno de TI, aumentando os custos de manutenção.➤ A falta de segurança sofisticada e estratégia de <i>backup</i> aumenta o risco de perda de dados.➤ Menos escalabilidade significa que quaisquer avanços futuros ou realocação de negócios podem ser bastante caros.

Tabela 8: Vantagens e Desvantagens da implantação On-Premises

Fonte: (Virginia, 2022).

2.8 Ferramentas *IaaS Open Source* Para *Cloud Computing*

No trabalho vamos analisar as ferramentas *open source* mais usadas para implementação de *Cloud Computing* orientado ao modelo *IaaS* que são: *OpenStack*, *OpenNebula*, *Eucalyptus* e *CloudStack*. Com a finalidade de realizar comparações sobre as razões da escolha do *OpenStack* como a solução adequada, abaixo são descritas as principais ferramentas *open source* para *cloud computing*. Cada uma possui as suas próprias características e funcionalidades, e a escolha depende dos requisitos e necessidades específicas de cada organização.

2.8.1 OPenStack

2.8.1.1 Definição

Segundo Manzanero(2015), o *openStack* é uma coleção de tecnologias *Open Source* que fornecem um *software* para a implantação da *cloud computing*. O *OpenStack* prove uma *IaaS* especializada em controlar grandes *pools* de computação, armazenamento e rede de um *data center* de forma modular e escalável, o projeto *openStack* foi iniciado em 2010 pela empresa *Rackspace Cloud* e pela agência espacial Americana, *NASA*. Atualmente, mais de 150 empresas já aderiram ao projeto, empresas importantes no mercado como a *AMD*, *Intel*, *Canonical*, *SUSE Linux*, *Red Hat*, *IBM*, *Dell*, *HP*, *Cisco* e outras. Conforme ilustrado na figura 11.

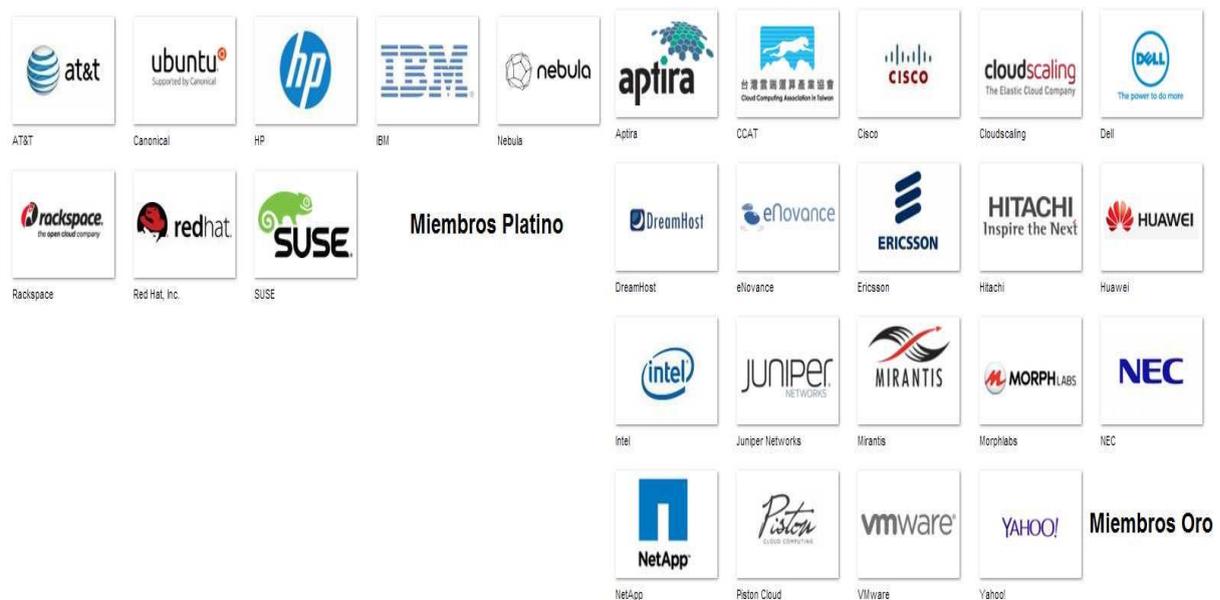


Figura 11: Membros da OpenStack Foundation
Fonte: Adaptado de Verdura(2014)

Para Melo e Azevedo(2019), O OpenStack é um sistema modular criado inicialmente como uma interface com a AWS, ele possui aspetos que o tornam capaz de criar uma *cloud* privada. Isto é devido principalmente ao modelo de *tenant* ou projeto.

Segundo Manzanero (2015), Para criação de recursos, o OpenStack conta com diversos módulos, onde cada um é responsável por gerir uma parte específica dos recursos de infraestrutura. Porém, em cada implementação, o uso destes módulos depende do propósito que se quer alcançar, exceto quando se trata dos *core* que existir obrigatoriamente.

Segundo Alselek (2016), na arquitetura *OpenStack* os principais módulos *cloud* são compostos por três nós: *contoller*, *compute* e *network*.

Nó *Controller*- Ele fornece serviços centrais de gestão de comunicação da *IaaS* na *cloud* e controla os outros nós da *cloud*, ele cria e opera a *IaaS* em *cloud* de forma eficiente e escalável.

Os serviços básicos fornecidos por este nó são:

- Autenticação,
- Autorização,
- Gestão de imagens,
- Gestão de rede e
- Provisão de recursos.

Nó *Compute*- Ele fornece serviços computacionais reais. Este nó tem a responsabilidade de executar instâncias de *VMs* no OpenStack. As instâncias são geradas neste nó. Este modo também faz a gestão de agentes da camada 2 que operam redes externa, *plug-in* de rede e implementação de grupos de segurança.

Os serviços básicos fornecidos por este nó são:

- Provisão e gestão de *VMs*;
- Gestão de imagens;
- Gestão de Tarefas; e
- Gestão de recursos computacionais.

Nó network - Ele fornece e controla os serviços de rede. Ele entrega diferentes serviços da camada 2 e camada 3, como implantação de redes virtuais e túneis, *NAT*, *DHCP* e roteamento. Este nó também tem a responsabilidade de garantir que as instancias tenham conectividade com a *Internet*.

Os serviços básicos fornecidos por este nó são:

- ✓ Gestão da Rede;

Serviços de rede para as *VMs*;

- ✓ Criação e gestão de de redes virtuais, sub-redes e *routers*;
- ✓ Gestão de endereços IP;
- ✓ Alocação de *IPs* para as *VMs* e serviços de rede;
- ✓ Implementação de serviços *DNS* e *DHCP*; e
- ✓ Estabelecimento de ligação entre as *VMs* e a *internet* e serviços de *VPN*.

2.8.2 Arquitetura da plataforma *OpenStack*

2.8.2.1 rquitetura conceptual

Segundo Verdura(2014), a plataforma *OpenStack* permite desenvolver todos os tipos de *cloud*, de uma maneira simples de implementar, massivamente escalável e rico em recursos. Para implantar essas *cloud*, o *OpenStack* fornece uma solução *IaaS*. através de uma série de serviços que se relacionam entre si através de *APIs* públicos.

Para Verdura(2014), note-se que estes serviços(também conhecidos como projetos) estão totalmente isolados. Desta forma, pode implantar apenas os serviços necessários para responder as necessidades particulares de cada projeto (sendo este um claro exemplo da flexibilidade que o *OpenStack* possui).

2.8.2.2 Serviços utilizados pelo *OpenStack*

Segundo Verdura (2014), *OpenStack* é dividido em diferentes serviços, denominados *projetos* na sua terminologia. Os projetos encontrados disponíveis no *OpenStack* são *Dashboard*, *Compute*, *Networking*, *Object Storage*, *Block Storage*, *Identity*, *Image*, *Telemetry* e *Orchestration*.

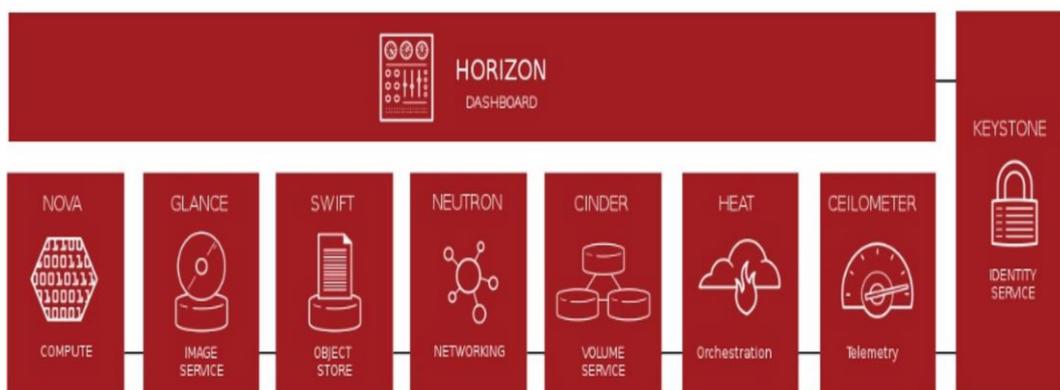


Figura 12: Projetos do OpenStack

Fonte: Verdura(2014)

2.8.2.3 Descrição de Serviços do OpenStack

Para Verdura (2014), a descrição dos projetos utilizados no OpenStack é composta por serviço, projeto e as suas características, detalhado na tabela 9.

Serviço	Projeto	Descrição
<i>Dashboard</i>	<i>Horizon</i>	Fornecer uma página inicial, baseado na <i>web</i> , para interagir com os serviços do OpenStack, para criação de uma instância, atribuir endereços <i>IP</i> ou configurações de segurança.
<i>Compute</i>	<i>Nova</i>	Gere o ciclo de vida das instâncias <i>VM</i> no OpenStack. É responsável pela criação, planejamento e desmontagem das <i>CVM</i> .
<i>Network</i>	<i>Neutron</i>	Permite a conectividade de rede como um serviço, para outros serviços do OpenStack. Fornece uma <i>API</i> para que os usuários definam as redes, permite a criação de Redes Virtuais de Inquilinos e suas possíveis conexões.
Armazenamento		
<i>Object Storage</i>	<i>Swift</i>	Armazena e recupera objetos de dados não estruturados, através de uma <i>API REST</i> baseada em <i>HTTP</i> . É altamente tolerante a falhas devido à sua replicação de dados e arquitetura escalável.
<i>Block Storage</i>	<i>Cinder</i>	Fornecer armazenamento persistente para instâncias em execução. Sua arquitetura facilita a criação e gestão de dispositivos de armazenamento em bloco.
Serviços Compartilhados		
<i>Identity</i>	<i>Keystone</i>	Fornecer serviço de autenticação e autorização para os outros serviços do OpenStack. Também fornece um catálogo com todas as interfaces dos serviços do OpenStack.
<i>Image</i>	<i>Glance</i>	Armazena e recupera imagens de disco de <i>VM</i> . O OpenStack <i>Compute</i> faz uso deste serviço ao criar uma instância. Armazena e recupera imagens de máquinas virtuais.
<i>Telemetry</i>	<i>Cielometer</i>	Monitora e mensura vários aspectos da <i>cloud</i> para faturamento, benchmarking (comparações de mercado), escalabilidade e para fins estatísticos
Serviços de alto Nível		
<i>Orchestration</i>	<i>Heat</i>	Orquestra várias composições de aplicações de <i>cloud</i> usando ambos os modelos com formato <i>HOT</i> nativo de template como ou o formato <i>AWS Cloud Formation</i> .

Tabela 9: Descrição dos Serviços/Projetos da Plataforma OpenStack

Fonte: (Verduara,2014)

2.8.3 Arquitetura lógica do OpenStack

Para Verdura (2014), uma vez conhecida a arquitetura conceitual da plataforma *OpenStack*, descreve brevemente a sua arquitetura lógica. A arquitetura lógica da plataforma é composta por vários módulos, dos quais selecionamos três tipos:

- **Daemons:** em plataformas *Linux* geralmente são instalados como serviços.
- **Script:** realiza instalações e testes no ambiente virtual de um serviço.
- **CLI:** permite que os usuários façam chamadas para a *API* dos serviços disponíveis no *OpenStack* através de comandos simples.

O diagrama a seguir mostra a mais comum, mas não a única, arquitetura lógica *cloud* em *OpenStack* descrito na Figura 9. nele pode apreciar a interação dos usuários finais por meio do painel, *CLIs* e as *APIs*(Verdura, 2014):

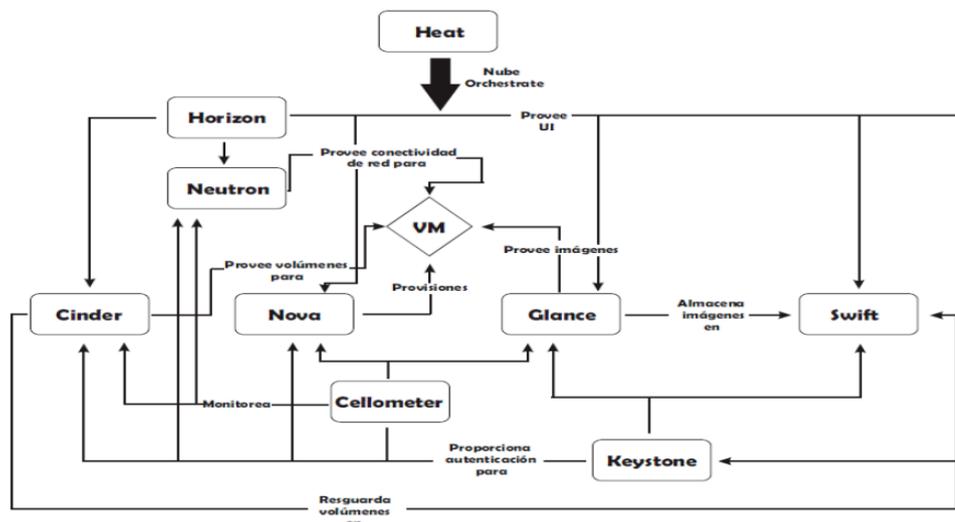


Figura 13:Arquitetura lógica típica de uma *cloud OpenStack*
Fonte: Adoptado de Manzanero(2015)

2.8.4 OpenNebula

A *OpenNebula* começou por ser um projeto de investigação em 2005 no grupo de pesquisa de Arquiteturas *DSA*. Este grupo, sediado na *Universidade Complutense de Madrid* é uma equipe de pesquisa focada na prestação de serviços em uma infraestrutura distribuída: virtualização, armazenamento e rede para plataformas *IaaS* (Verdura, 2014).

Aissaoui et al.(2012), O *OpenNebula* é uma plataforma *cloud computing* de *open source* desenvolvida pela *NASA*. Ele foi projetado para atender às necessidades exclusivas da *NASA* e oferece recursos de *IaaS* para pesquisa científica em

desenvolvimento. *OpenNebula* foi concebido para construir *cloud* privada, pública e híbrida. Este gere um conjunto de recursos como armazenamento, rede, virtualização ou políticas de segurança, combinando assim os recursos dos diferentes *data centers*. Ele permite que o usuário implemente e gerencie máquinas virtuais em recursos físicos e pode definir *data centers* ou *clusters* de usuários para infraestrutura virtual flexível que pode se adaptar automaticamente à mudanças do *workload*.

Segundo (Verdura, 2014), a plataforma *OpenNebula* gradualmente modificou sua licença até que se torne completamente gratuito e seja amplamente suportado por um grande comunidade de desenvolvedores e integradores. Ela consolidou comercialmente em 2010 com a criação da empresa C12G, que agrega o valor comercial agregado que muitas empresas demandavam e libera o projeto de financiamento público quase exclusivo.

Segundo Ferreira(2017),O *OpenNebula* têm duas funcionalidades chave a gestão virtual da infraestrutura e gestão da *Cloud*. Com a gestão virtual da infraestrutura o *OpenNebula* integra-se diretamente com os *hypervisors* (*KVM*, *Xen* ou *VMware*, *ESX*) e tem controlo completo sobre os recursos virtuais e físicos, garantindo assim uma excelente capacidade de otimização e gestão de recursos em alta disponibilidade. Com a gestão da *Cloud* é possível criar múltiplos projetos e gerir de forma simples a *Cloud* através da sua interface gráfica. Esta plataforma é recomendada para quem procura soluções pequenas/médias através da *Cloud Computing*.

Para (Verdura, 2014) O *Nebula* é um exemplo de uma plataforma de *cloud computing* específica do setor que foi adaptada para atender aos requisitos exclusivos de uma organização específica. O *Nebula* foi construído com base no *OpenStack* e em outras tecnologias *openstack* e inclui recursos como:

- Provisionamento e gerenciamento de *VMs*.
- Gerenciamento de rede e armazenamento.
- Recursos avançado de segurança, isolamento de rede e criptografia de dados.

2.8.5 Eucalyptus

A infraestrutura *Eucalyptus* iniciou o seu percurso em Maio de 2008, no *Start-up* Californiana *Eucalyptus Systems*. Este nascimento precoce tem feito com que fosse considerado um dos primeiros, mais estáveis e aplicativos completos desenvolvidos para implantar *IaaS* privado ou híbrido (Verdura, 2014).

Segundo Aissaoui et al. (2012), *Eucalyptus (Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems)* é uma plataforma de *cloud* de *open source* que segue o modelo *cloud computing* da *AWS*. Esta é uma plataforma *IaaS* que agrupa a infraestrutura virtualizada existente para criar recursos computacionais de *cloud* privada ou híbrida compatíveis com as *APIs* da *AWS*, rede e armazenamento. *Eucalyptus* usa as terminologias como imagens, instâncias e endereçamento *IP*, segurança, rede e controle de acesso.

Segundo Verdura (2014), uma das principais características pelas quais o *eucalyptus* é conhecido é por causa de sua capacidade de implantar *IaaS* híbrido graças à sua compatibilidade com *AWS*, considerado o padrão de fato para *cloud* pública. Isso significa que o *Eucalyptus* permite nativamente a interoperabilidade com serviços *AWS* por meio das *APIs EC2* e *S3* da *Amazon*.

O maior problema do *eucalyptus* é que ele se distribui em duas edições, uma *open source* e outra *enterprise*. Isso significa que se desejar funcionalidade de *cloud* completa, deve optar por opções *open source*, que embora inclua um núcleo composto por uma grande quantidade de *software* gratuitos, possui outros componentes com licenças privadas. Razão pela qual, embora inicialmente muitas empresas e organizações mostrassem um considerável interesse neste cenário, eles acabaram descartando-o em favor de outras opções totalmente gratuitas, como *OpenStack* (Verdura, 2014).

O *Eucalyptus* é frequentemente usado por organizações que desejam manter a compatibilidade com a *AWS*, ao mesmo tempo em que mantêm o controle de sua própria infraestrutura. Algumas das principais funcionalidades do *Eucalyptus* incluem (Verdura, 2014):

- Compatibilidade com a *API* da *AWS*, permitindo que aplicativos e serviços desenvolvidos para a *AWS* sejam executados no *Eucalyptus*.

- Provisionamento de máquinas virtuais, gerenciamento de armazenamento e rede, e balanceamento de carga.
- Recursos de segurança avançados, como criptografia de dados e isolamento de redes.

2.8.6 CloudStack

O *CloudStack* foi originalmente desenvolvido, como o *Eucalyptus*, por outra empresa californiana chamada *Cloud.com*. Um ano após a sua libertação quase todo o *CloudStack* como *open source* sob tutela da *Commonwealth General Public License GNU*, foi adquirida em 2011 por da multinacional *Citrix Systems*, tornando-se assim a plataforma oficial para *cloud computing* de produtos Citrix (Verdura, 2014).

Et al (2015), *CloudStack* é uma plataforma *open source*, escrita em *Java*, projetada para desenvolvimento de *cloud computing* orientada a *IaaS*. Este gere um conjunto de recursos como armazenamento, rede, virtualização ou políticas de segurança, que são utilizadas para a criação de máquinas virtuais em infraestruturas diferentes, combinando assim os recursos dos diferentes *data centers*. Ele agrega recursos *cloud computing* privadas, públicas ou híbridas. Este utiliza *hypervisors* para virtualização, como *KVM* ou *VMWare*.

A *CloudStack*, como *OpenNebula* ou *OpenStack*, é uma plataforma *cloud computing open source* usada para criar, gerir e implantar *IaaS*. Por estas razões e por ser reconhecida como uma plataforma com pouco demanda por recursos, com documentação sólida e ótima usabilidade, tem sido tornou-se uma das grandes opções no desenvolvimento *cloud* (Verduras, 2014).

Para Ferreira(2017), a infraestrutura de gestão da *CloudStack* é altamente escalável e pode servir de gestão a dezenas de milhares de *hosts* instalados em *data centers* geograficamente distribuídos. Para uma instalação mínima de produção utilizando *CloudStack* bastam duas máquinas: uma para executar o *CloudStack Management Server* e outra para ser utilizada como Infraestrutura da *cloud*, sendo que esta ultima instalação é mesmo uma infraestrutura mínima e simples.

O *CloudStack* é conhecido por sua facilidade de uso e robustez, sendo adotado por provedores de serviços *cloud* e empresas de grande porte, ele é projetado para ser escalável e oferece recursos como (Verduras, 2014):

- Gestão de rede, incluindo a configuração de redes virtuais e balanço de carga.
- Gestão de armazenamento em bloco, permitindo a criação e o gestão de volumes de armazenamento.
- Funcionalidades avançadas, como migração ao vivo de VMs, balanço de carga automático e suporte a *snapshots*.

2.9 Análise comparativa de soluções *IaaS* open source

Após identificadas as principais soluções *open source*, realizou-se uma análise comparativa entre as ferramentas *IaaS*, para assim verificar o resumo das vantagens e desvantagens de cada e refletir sobre a escolha do *OpenStack* como a melhor opção

As ferramentas escolhidas são das principais para prover *IaaS*, cada uma tem suas características e funcionalidades, a escolha varia de requisitos e necessidades da organização, as descrições são descritas na tabela (Anaka et al, 2019).

Características	<i>OpenStack</i>	<i>Eucalyptus</i>	<i>OpenNebula</i>	<i>CloudStack</i>
Ano	2010	2008	2008	2010
Tipo de <i>Cloud</i>	<i>public</i>	<i>Public,private</i>	<i>Private</i>	<i>Public, Private</i>
Compatibilidade	<i>Supports EC2, S3</i>	<i>Supports EC2, S3</i>	<i>Open, Multiplatform</i>	<i>Supports,EC2, S3</i>
Modo de Implantação	<i>Web interface /command line</i>	<i>Command line interface</i>	<i>Command line / web interface</i>	<i>Web interface/CLI</i>
Hypervisor	<i>KVM,Xen, ESX, VMWare, LXC Hyper-V</i>	<i>VMWare, Xen, KVM</i>	<i>Xen, VMWare, ESX, KVM,ESXi</i>	<i>VMWare, Oracle VM, KVM, XEN, Citrix</i>
Interface Web	<i>Native API, Amazon EC2 API</i>	<i>Web Service</i>	<i>LibVirt, EC2, OCCl, API</i>	<i>EC2, WSDL, WSRF</i>
SO Suportado	<i>Linux, Windows X86 Server</i>	<i>Linux</i>	<i>Linux</i>	<i>Mac OS x,Asianux, Cent OS, Debian, DOS</i>
Language de programação	<i>Python</i>	<i>Java, C, Python</i>	<i>Java, C++, RUBY</i>	<i>Java</i>
Segurança	<i>Protection against DOS / faulty clients. Keystone for identity mgt</i>	<i>Public and private key</i>	<i>Authen,RSA, SSH, password, LDAP</i>	<i>Cloudsstack security groups</i>
Tipo de Licença SO	<i>Apache license version 2</i>	<i>BSD License</i>	<i>Apache version 2</i>	<i>Apache license</i>

Tabela 10: Análise comparativa de soluções *IaaS* open source
Fonte:(Anaka et al, 2019)

2.10 Diferentes *Hypervisors* suportados pelos *softwares de IaaS*

Segundo Ferreira (2017), dos diferentes *hypervisors* suportados pelos softwares open source abordados anteriormente, claramente entre todos os *softwares*, o OpenStack é o que oferece maior destaque no que concerne à compatibilidade com *hypervisors*, sendo também o único compatível com o *Hyper-V*, o sistema de virtualização desenvolvido pela Microsoft, bem como é o único compatível com *Docker*, tecnologia *open source* que permite criar, executar, testar e implementar aplicações distribuídas dentro de contentores de software, sendo estes contentores um método de virtualização de um sistema operativo, conforme ilustrado na tabela abaixo:

	KVM	XEN	ESXi	Hyper-V	LXC	Docker
<i>OpenNebula</i>	Sim	Sim	Sim	Não	Não	Não
<i>Eucalyptus</i>	Sim	Não	Sim	Não	Não	Não
<i>CloudStack</i>	Sim	Sim	Sim	Não	Não	Não
<i>OpenStack</i>	Sim	Sim	Sim	Sim	Sim	Sim

Tabela 11: Diferentes *hypervisores* suportados pelos *softwares de IaaS*

Fonte: Adaptada de (Turel,2013)

3 CAPITULO III – CASO DE ESTUDO

3.1 Descrição do Caso de Estudo

3.1.1 Caracterização do Grupo Entrepasto Origens em Moçambique

O Grupo Entrepasto teve o seu início em Moçambique na década 40, com a constituição e desenvolvimento de um conjunto de empresas e um núcleo comum de acionistas, operando basicamente nos sectores agroindustriais e comercial, hoje agrupadas na Companhia de Moçambique.

Tendo iniciado a sua atividade no sector algodoeiro, o Grupo viria a investir progressivamente em outras áreas dos negócios, na representação, venda e assistência técnica de veículos e máquinas agrícolas, no fabrico de óleos alimentares e sabões no processamento da castanha de caju, no sector madeireiro, na segurança e no turismo cinegético.

3.1.2 Atividades do grupo entreposto no mundo

Vocacionado desde a sua origem para romper fronteiras, o Grupo Entrepasto, entrou no novo milénio solidamente implantado em três continentes, provando que uma das suas principais características e a expansão constante das suas atividades através da diversificação de negócios e deteção de oportunidades em país tão distintos como Portugal, Brasil, Moçambique. Da indústria agrícola, passando pelos serviços, pelo comércio ou mesmo pelo imobiliário, sector em que tem feito novos investimentos, são inúmeras as áreas de atuação onde o grupo está presente

3.1.3 Organigrama da Empresa

Departamento de Recursos Humanos: É a área da gestão técnica do pessoal e administrativa, tendo como função: triagem curricular, análise de curriculum vitae, entrevistas, criação do registo de processos e ações laborais.

5

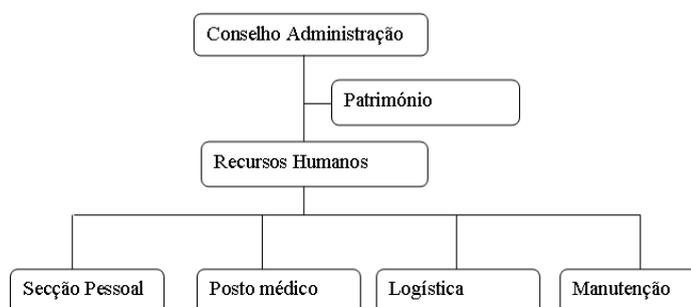


Figura 14: Organograma do Departamento de Recursos Humanos

Departamento de Direção de Vendas: É a área que determina os acordos e as condições de venda.

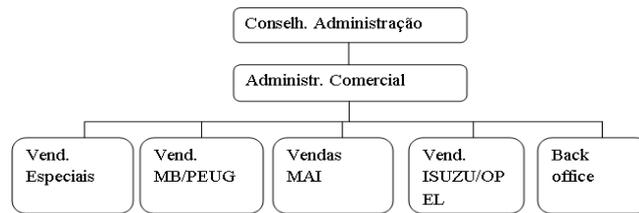


Figura 15: Organograma do Departamento de Direção de Vendas

Direção Administrativa e Financeira: É a área responsável pela administração dos recursos financeiros da empresa, no sentido de fazer com que os ganhos da empresa sejam suficientes não só para cobrir todos os gastos do dia-a-dia, mas também para gerar lucros e assim garantir o seu desenvolvimento

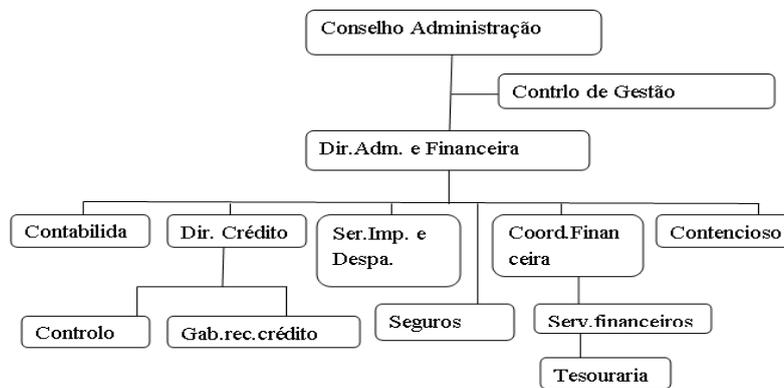


Figura 16: Organograma da Direção Administrativa e Financeira

Direção de Serviços de Informação: É a área que Concebe, desenvolve e gere sistemas baseados em computadores.

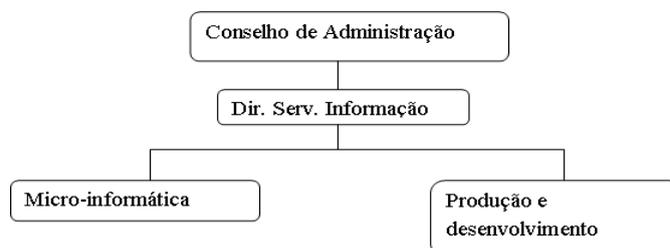


Figura 17: Organograma da Direção de Serviços de Informação

3.1.4 Apresentação da ServiSIS Moçambique

A ServiSIS Moçambique é uma das empresas do Grupo Entreposto dedicada ao fornecimento de serviços na área de Sistemas de Informação, foi fundada em 2006, ela se dedica ao fornecimento de soluções empresariais na área de Tecnologias e Sistemas Informação. A sua visão é ser o parceiro preferencial em serviços e sistemas de informação em Moçambique, oferecendo uma estratégia completa e diversificada, assente na diferença, criatividade, orientação estratégia e mais-valia dos nossos produtos, serviços e profissionais.

Constrói plataformas 100% otimizadas para a dimensão e atividade das empresas, destacando-se pela capacidade de responder à especificidade de cada projeto e sustentando o seu serviço numa lógica de flexibilidade e fiabilidade, apoiada pelo investimento constante na qualificação da equipa e no estabelecimento de parcerias com as maiores referências mundiais na área das tecnologias de informação.

A Dataserv, por seu lado, é uma empresa líder no setor das tecnologias de informação, proporcionando um vasto leque de serviços e produtos especialmente orientados para empresas e particulares para quem a segurança, a eficácia e a capacidade de resposta sejam fundamentais.

Os profissionais estão ao dispor do cliente para criar, desenvolver e fornecer soluções por medida que se adequem às suas reais necessidades com 100% foco no cliente.

3.1.4.1 Missão

A serviSIS assume como sua missão abortar ao mercado soluções de gestão estratégica e operacional aos seus clientes, ajudando-os no caminho do sucesso. Para tal, tem uma abordagem diferenciada em que o ROI – *Return on Investment*, começa no momento do contacto do cliente.

3.1.4.2 Visão

A serviSIS tem como principal objetivo ser uma empresa de destaque em Moçambique em soluções empresariais na área de Tecnologias e Sistemas Informação e para isso criou uma abordagem ao mercado apoiada em metodologias próprias sempre orientadas para o cliente. Nesse sentido apostou numa equipa especialista e dedicada aos mercados alvo onde atua.

Estão absolutamente convictos de que a maioria das empresas necessita de novos sistemas de informação e que a ServiSIS com o conhecimento profundo do negócio, dos problemas e das soluções vai afirmar-se como o “seu” parceiro para as Tecnologias e Sistemas de Informação.

3.1.4.3 Valores

Privilegia a transparência no relacionamento com os clientes e estão convictos que sua competência permite implementar qualquer projeto com elevada satisfação para o cliente que acredita no seu trabalho e nas soluções que implementa.

O sucesso dos seus clientes é o nosso, e a ServiSIS atenta às necessidades dos clientes procura constantemente novas soluções para o negócio, são uma equipa empreendedora, dinâmica e proactiva com um enorme rigor e determinação na execução dos projetos.

Presente no dia-a-dia, os sei valores têm-se demonstrado fundamentais, para em conjunto com os clientes conseguirmos atingir a nossa missão com sucesso, que é a de proporcionar às empresas as soluções adequadas ao seu correto funcionamento com elevados padrões de qualidade e obter a Satisfação Total do Cliente.

3.1.4.4 Serviços Prestados

A ServiSIS Disponibiliza uma oferta alargada de serviços dentre eles:

- ✓ Consultoria em Tecnologias e Sistemas de Informação
- ✓ Análise de Processos de Negócio e Implementação de Soluções de Gestão e Estratégicas, de acordo com as best practices;
- ✓ Implementação *Dynamics NAV, CRM, Gestware e GesvenGRH*
- ✓ *HelpDesk*;
- ✓ Formação em Sistemas de Gestão, Infra-estruturas;
- ✓ *ASP*; e
- ✓ Serviços de *Outsourcing* em *IT*;

3.1.5 Divisão da Equipa da ServiSIS

A equipa de suporte da ServiSIS divide-se em 3 áreas principais:

- **Equipe de Analistas de Sistemas:** responsável pelo desenvolvimento, manutenção e suporte a clientes externos que aderem aos serviços prestados pela ServiSIS. Possui um funcionário da empresa e dois em fase de integração.
- **Equipe de Suporte ao Usuário:** dividida em duas equipas a de Microinformática a de Suporte Aplicacional. Ela é responsável pela infraestrutura de comunicação, atendimento inicial aos usuários, controle de acesso, SO, atuando na troca de periféricos, na resolução de problemas básicos de rede, na instalação e remoção de *softwares* e auxiliando os usuários do edifício sede e das delegações a nível nacional nos procedimentos cotidianos. Possui dois funcionários da empresa e dois estagiário na equipa de suporte aplicacional, e também, um funcionário da empresa e um estagiário na equipe de micro informática.

3.1.6 Descrição da Situação Atual

Segundo o cenário observado na ServiSIS, a empresa possui um *data center on-premises* privado, o mesmo atende o edifício sede do Grupo Entrepósito Auto é usada para conceder acesso as aplicações, documentos e dados da instituição entre os departamentos, os diversos departamentos podem obter informações disponível nos servidores locais em qualquer momento, de acordo com as permissões de acesso a aplicações, facilitando assim a comunicação na rede interna do edifício.

A ServiSIS é responsável pelo departamento de IT, responsável por a nível do grupo garantir o bom funcionamento da rede local e garantir a disponibilidade de aplicações aos clientes da empresa, através da plataforma *web* de suporte *helpdesk* denominada *servicedesk* onde estes expõe problemas de acessos a aplicações e outras inquietações a equipe de suporte.

3.1.7 Descrição dos constrangimentos

Segundo o cenário observado no caso de estudo verificou-se que existem vários constrangimentos em ter um *data center on-premises* que não funciona na modalidade *cloud computing*, e alguns dos principais constrangimentos incluem:

- **Custo Elevado** - A construção e manutenção de um *data center on-premises* pode ser extremamente dispendiosa. Há necessidade de investir em *hardware*, como *servers*, *router*, *switches* e unidades de armazenamento, bem como em

sistemas de alimentação elétrica redundante, resfriamento adequado, sistemas de segurança física, eletricidade e manutenção.

- **Limites de escalabilidade** - O *data center on-premises*, está restrito à capacidade física disponível no local. Caso a demanda por recursos de computação ou armazenamento aumentar, será necessário adquirir e instalar *hardware* adicional, o que pode levar tempo e requerer investimentos significativos.
- **Complexidade e tempo de implementação** - Construir e configurar um *data center on-premises* é um processo complexo e demorado. Envolve a coordenação de várias equipes, como engenheiros de rede, especialistas em segurança, eletricitas e outros. A aquisição de *hardware*, a instalação e a configuração também podem levar semanas ou até meses para serem concluídas.
- **Manutenção e atualização contínuas** - Um *data center on-premises* requer uma equipe dedicada para realizar a manutenção regular e as atualizações de *hardware* e *software*. Isso inclui substituição de componentes defeituosos, aplicação de *patches* de segurança, atualizações de *firmware* e muito mais. Essas atividades podem interromper os serviços e exigir tempo e esforço consideráveis.
- **Falta de redundância e alta disponibilidade** - A menos que seja projetado com redundância adequada, um *data center on-premises* pode ser vulnerável a interrupções de energia, falhas de *hardware* ou desastres naturais. Manter uma infraestrutura altamente redundante e com alta disponibilidade pode ser um desafio técnico e financeiro para muitas organizações.
- **Risco de segurança** - Um *data center on-premises* requer medidas de segurança física e lógica para proteger os dados e os sistemas. Isso inclui sistemas de controle de acesso, monitoramento de vídeo, sistemas de detecção de intrusão, *firewalls* e muito mais. Garantir a conformidade com as melhores práticas de segurança pode ser uma tarefa complexa e exigir recursos adicionais.

4 CAPITULO IV – MODELO DE SOLUÇÃO

4.1 Proposta de Solução

4.1.1 Visão esquemática da infraestrutura IaaS em OpenStack

Para superar as dificuldades descritas no Capítulo 3, propõe-se o desenvolvimento de uma solução *IaaS* em *cloud*, utilizando a ferramenta *OpenStack*. O objetivo é fornecer características de um *data center* na *cloud* e aproveitar seus melhores atributos para otimizar a manutenção da infraestrutura *on-premises* da ServiSIS, operando no modelo *cloud computing*.

O diagrama a seguir descreve, em cinco camadas, o protótipo arquitetural da infraestrutura proposta: *hardware*, *SO host*, *VirtualBox*, *Ubuntu Desktop* e *OpenStack*. Essas camadas serão utilizadas para criar nosso ambiente virtual na máquina host. A máquina virtual será configurada com *Ubuntu Desktop* para suportar o *OpenStack* e seus serviços. Os serviços do *OpenStack* serão acessíveis tanto internamente quanto externamente à máquina virtual.

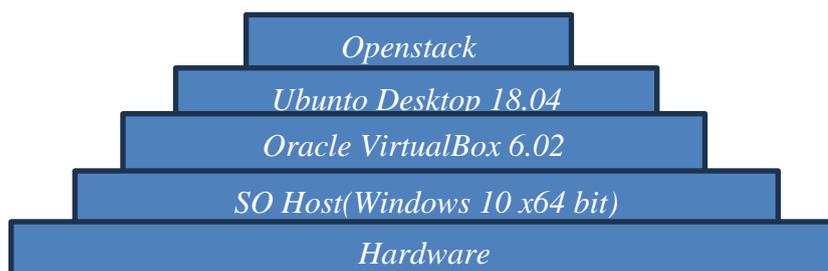


Figura 18: Arquitetura do sistema
Fonte: Do autor

4.2 Recursos Disponíveis para Implementação da Solução

Os recursos disponíveis para implementar um ambiente de teste estão descritos a seguir com uma abordagem da base para o topo:

4.2.1 Recursos de Hardware

Para a camada de *hardware*, utilizaremos um *laptop DELL Latitude E5440*, que possui uma tela de 14 polegadas com resolução *Full HD* e um processador *Intel Core i7* de 4ª geração. Este *laptop* inclui recursos de segurança, como leitor de impressões digitais e *TPM* (Dell, 2024).

O Dell Latitude E5440 oferece um equilíbrio adequado entre poder de processamento e eficiência energética, sendo capaz de lidar com tarefas corporativas

diárias, como navegação na *web*, criação de documentos, planilhas e apresentações, além de multitarefa moderada. Ele combina bom desempenho, segurança e durabilidade, atendendo às necessidades dos profissionais que valorizam produtividade e portabilidade (Dell, 2024). A Tabela 12 resume as principais características do *Dell Latitude E5440* e as especificações do laptop.

Característica	Descrição
Processador	<i>intel(R) core(TM) i7-4600U CPU @ 2.10GHz 2.70GHz</i>
Opções de armazenamento	<i>HDD ou SSD</i>
Recursos de segurança	<i>Leitor de impressões digitais, TPM</i>
Portas	<i>USB, HDMI, slot para cartão SD</i>
Conectividade	<i>Wi-Fi, Bluetooth</i>
Tamanho da tela	<i>14 polegadas</i>
Memória	<i>8 GB</i>
Armazenamento	<i>1000 GB</i>

Tabela 12: principais características do Dell Latitude E5440
Fonte: (Autor)

4.2.2 SO Host -Windows 10

O Windows 10, um sistema operacional da Microsoft que sucedeu o Windows 8.1, foi lançado em 1º de outubro de 2014. Este sistema permite a criação de uma conta para cada usuário do computador, sendo um sistema operacional multiusuário (Microsoft, 2024).

O Windows é um sistema operacional gráfico desenvolvido pela Microsoft, que utiliza imagens, ícones, menus e outros elementos visuais para facilitar o controle do computador. Ele é responsável por gerenciar todo o hardware e software, oferecendo interação através de uma interface gráfica de usuário (GUI) (Microsoft, 2024). A descrição do hardware é fornecida na tabela abaixo.

Componente	Descrição
Processador	<i>intel(R) core(TM) i7-4600U CPU @ 2.10GHz 2.70GHz</i>
Memória	<i>8 GB</i>
Armazenamento	<i>1000 GB</i>
Sistema Operativo	<i>Windows 10 Pro x64</i>

Tabela 13: Descrição do SO do Host
Fonte: (Autor, 2023)

4.2.3 Oracle *VirtualBox*

O Oracle *VirtualBox* é um poderoso *hypervisor* para arquiteturas *x86* e *AMD64/Intel64*, amplamente utilizado tanto em ambientes corporativos quanto domésticos. Reconhecido por sua riqueza de recursos e alto desempenho, o *VirtualBox* se destaca como a única solução profissional de virtualização disponível gratuitamente como *software open source*, sob os termos da *GNU General Public License* versão 3 (Oracle, 2024).

Inicialmente desenvolvido pela *Innotek GmbH*, o *VirtualBox* foi adquirido pela *Sun Microsystems* em 2008 e, posteriormente, pela *Oracle* em 2010, que continuou seu desenvolvimento e suporte. Esta ferramenta oferece uma ampla gama de funcionalidades, incluindo suporte para múltiplos SO, *snapshots*, escalabilidade e uma interface amigável, o que a torna uma escolha popular para desenvolvedores e administradores de sistemas (Oracle, 2024).

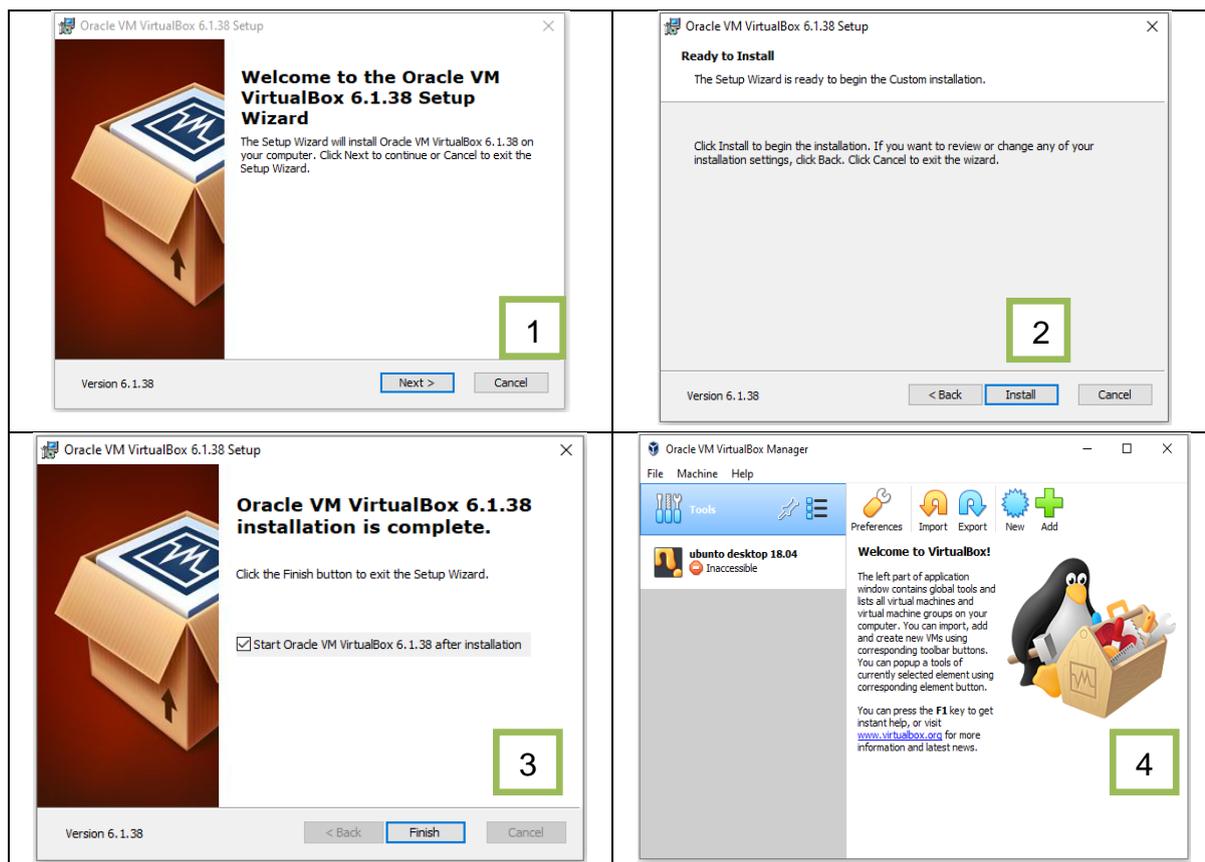


Figura 19: Icon do VirtualBox
Fonte: (oraclevirtualbox.org, 2024)

Atualmente, o *VirtualBox* é executado em *hosts Windows, Linux, macOS, Solaris* e oferece suporte a um grande número de SO *guest*, incluindo, nomeadamente o *Windows (NT 4.0, 2000, XP, Server 2003, Vista, Windows 7, Windows 8, Windows 10)*, *DOS/Windows 3.x, Linux(2.4, 2.6, 3.xe 4.x)*, *Solaris e Open Solaris, OS/2 e OpenBSD* (Oracle, 2024).

4.2.3.1 Etapas de Instalação do *VirtualBox*

Esta seção descreve o processo de instalação do *VirtualBox* em um ambiente com *Windows 10*. Primeiro baixar o *setup* para instalação do *VirtualBox* no *site* oficial. Em seguida, executamos o arquivo de *setup "VirtualBox-6.1.38-Win.exe"*, cujas instruções de instalação está descrita na sequencia de imagens abaixo.



4.2.4 Etapas Criação e configuração da máquina virtual

Nesta seção, descreve-se o processo de criação e configuração de uma máquina virtual para a instalação do SO *guest Ubuntu Desktop 18.04.3*. Ao executar o aplicativo *VirtualBox*, é ativado o "Gestor de VM no VirtualBox".

4.2.4.1 Descrição do SO *Guest - Ubuntu Desktop 18.04*

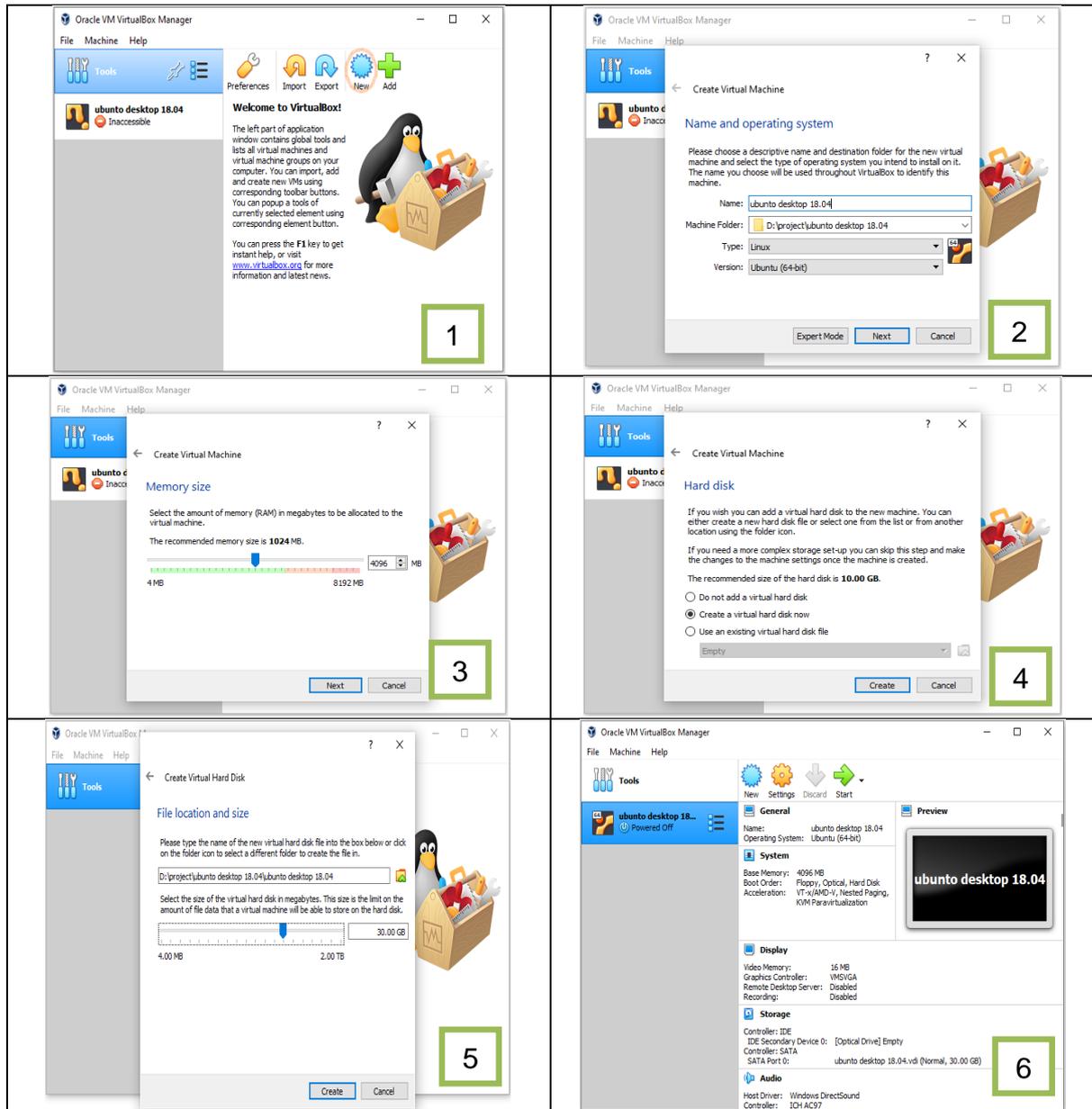
O *Ubuntu Desktop 18.04* é uma versão das versões de SO *Ubuntu*, desenvolvido para uso em computadores *desktops*. Ele oferece uma interface gráfica amigável e recursos avançados para atender às necessidades de usuários. Abaixo são descritos os recursos mínimos para a instalação do *Ubuntu Desktop 18.04* em relação a recursos como Processador, Memória e Espaço em disco.

Componente	Descrição
Processador	2 GHz dual-core ou superior.
Memória	2 GB ou mais
Armazenamento	10 GB ou mais
Sistema Operativo	<i>Ubuntu Desktop 18.04 x64</i>

Tabela 14: Descrição do requisitos mínimos do SO Guest
Fonte: (Autor)

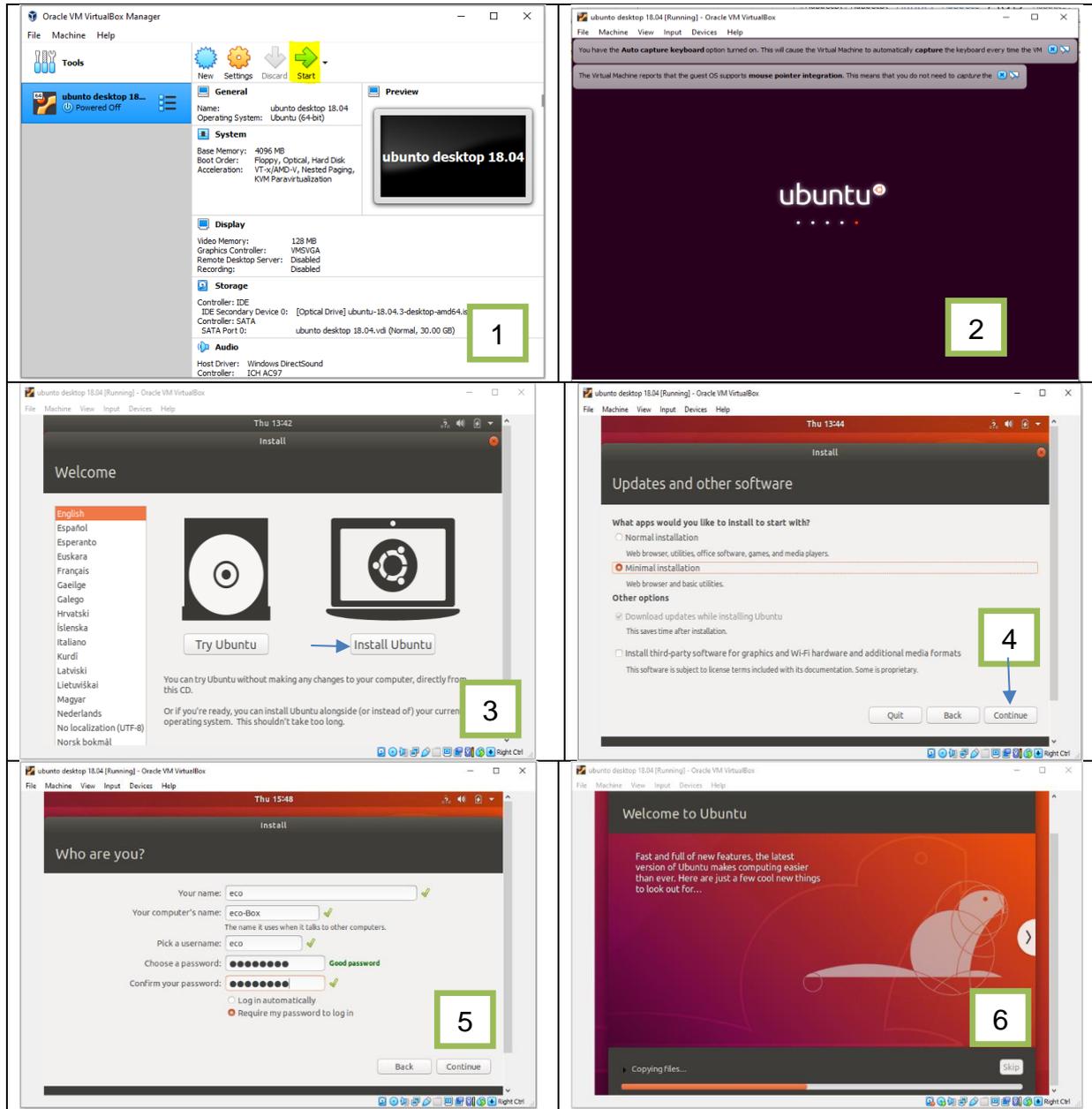
4.2.4.2 Sequencia de passos para configuração do ambiente de teste

Feita a instalação do VirtualBox, passamos para a fase de criação da VM e preparação dos requisitos de Processador, Memória e Espaço em disco para suportar a imagem ISO do *Ubuntu Desktop 18.04.3*, cujos procedimentos e instruções estão descritos na sequencia de imagens abaixo:



4.2.4.3 Sequencia de passos para instalação SO do ambiente de teste

Feita a identificação dos requisitos dos SO *guest*, passamos para a fase de instalação da imagem ISO do Ubuntu Desktop 18.04 , cujos procedimentos e instruções estão descritos na sequencia de imagens abaixo:




```
python3 -m venv openstack-venv
source openstack-venv/bin/activate
```

5. Atualizar *Pip* e *Setuptools*

```
pip install --upgrade pip setuptools
```

6. Instalar Dependências do *DevStack*

```
pip install -r requirements.txt
```

7. Configurar o *DevStack*

Crie um arquivo “*local.conf*” na raiz do diretório `devstack` com o seguinte conteúdo:

```
cat <<EOL > local.conf
[[local|localrc]]
ADMIN_PASSWORD=supersecret
DATABASE_PASSWORD=supersecret
RABBIT_PASSWORD=supersecret
SERVICE_PASSWORD=supersecret
HOST_IP=127.0.0.1
LOGFILE=$DEST/logs/stack.sh.log
VERBOSE=True
EOL
```

8. Resolver Conflitos de Dependências Específicos (se necessário), instalando as versões específicas de pacotes que podem causar conflitos:

```
pip install requests==2.23.0
pip install numpy==1.18.1
```

9. Executar o Script `stack.sh`, Certifique-se de estar no diretório `devstack` e com o ambiente virtual ativo:

```
./stack.sh
```

10. Verificação de Dependências e Conflitos, Use o `pipdeptree` para verificar a árvore de dependências e identificar conflitos:

```
pip install pipdeptree
pipdeptree
```

11. Corrigir Conflitos de Dependências (se necessário). Se você encontrar conflitos, ajuste as versões dos pacotes conforme necessário no arquivo `requirements.txt` ou instale diretamente usando `pip`.

12. Reativar o Ambiente Virtual (em sessões futuras) Sempre que você quiser trabalhar com o DevStack, ative o ambiente virtual:

```
cd devstack
source openstack-venv/bin/activate
```

13. Executar o DevStack

Execute novamente o script `stack.sh` se precisar reconfigurar ou reiniciar a instalação:

```
./stack.sh
```

A plataforma Cloud (OpenStack)

OpenStack permite gerir e prover VMs de várias especificações, com várias opções de SOs em diferentes locais bem como zonas. Podem ser máquinas em *cloud* pública ou privada. No caso de estudo o openstack foi instalado no Ubuntu Desktop 18.04.3 LTS. Conforme figura 20, ele é acessível pela *internet* através de um navegador *web* através do link https://ip_do_server/dashbboard. Depois de inserir seu nome de usuário e senha, você será direcionado ao painel de onde poderá girir os seus recursos.

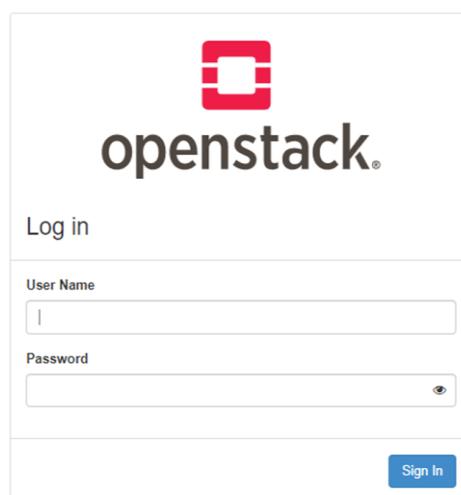


Figura 20: A interface gráfica de usuário (GUI) de login do OpenStack

Abaixo estão alguns instantâneos do módulo em funcionamento:

Horizonte (Painéis)

Conforme mostrado na figura 21 abaixo, este é o módulo relacionado à interface gráfica do usuário. É usado para criar painéis específicos.

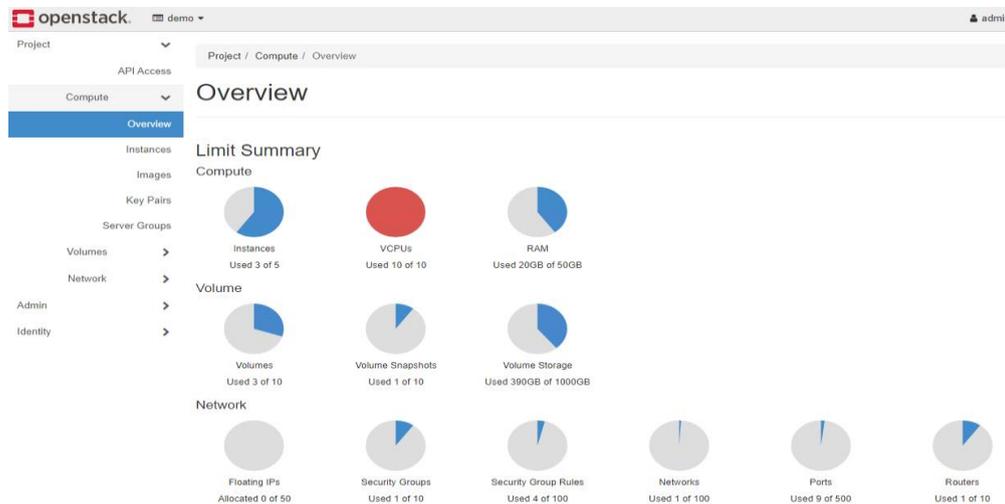


Figura 21: exemplo de painel OpenStack criado.

Nova (serviços de informática)

Este é o módulo usado para lidar com serviços de computação como máquinas virtuais, conforme visto na figura 22 abaixo. No nosso caso o usuário demo possui três instâncias (máquinas virtuais criadas).

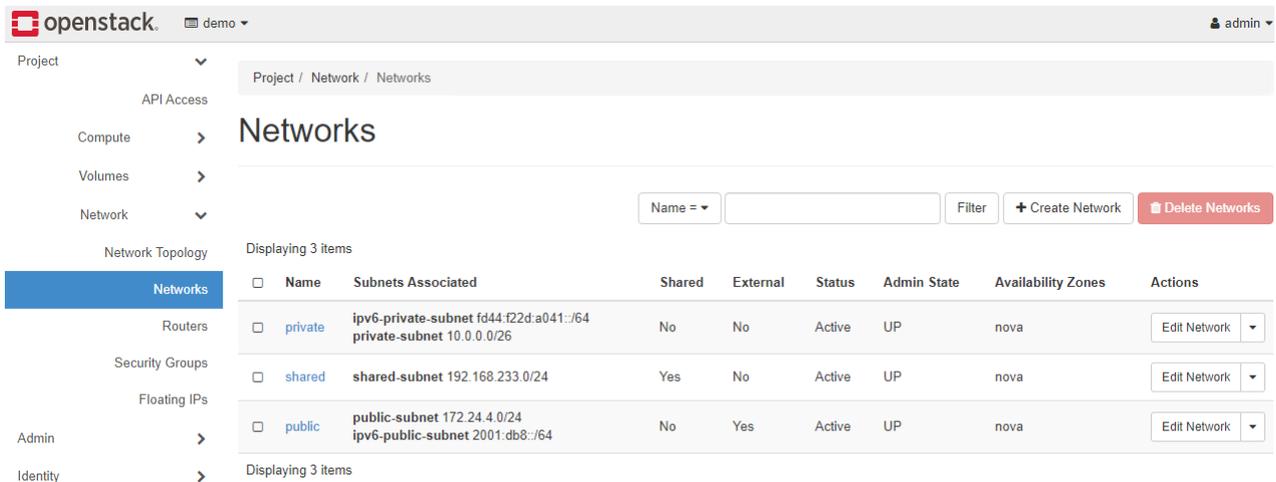
The screenshot shows the OpenStack Horizon 'Instances' page. The left sidebar contains navigation menus for Project, API Access, Compute, Overview, Instances, Images, Key Pairs, Server Groups, Volumes, Network, Admin, and Identity. The main content area is titled 'Instances' and features a table of virtual machines. The table has the following columns: Instance Name, Image Name, IP Address, Flavor, Key Pair, Status, Availability Zone, Task, Power State, Age, and Actions.

Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Age	Actions
<input type="checkbox"/> pascal-dem	pascal-ubuntu	shared 192.168.233.137 private 10.0.0.12, fd44:f22d:a041:0:f816:3eff:fe1d:6445	m1.small	-	Shutoff	nova	None	Shut Down	1 month, 2 weeks	<input type="button" value="Start Instance"/>
<input type="checkbox"/> cirros-pascal-test-prometheus	cirros-0.4.0-x86_64-disk	private 10.0.0.25, fd44:f22d:a041:0:f816:3eff:fe46:3384 shared 192.168.233.122	m1.xlarge	-	Shutoff	nova	None	Shut Down	1 month, 3 weeks	<input type="button" value="Start Instance"/>
<input type="checkbox"/> cirros-demo	cirros-0.4.0-x86_64-disk	private 10.0.0.40, fd44:f22d:a041:0:f816:3eff:fe46:3384 shared 192.168.233.160	m1.small	-	Shutoff	nova	None	Shut Down	1 month, 3 weeks	<input type="button" value="Start Instance"/>

Figura 22: Exemplos de máquinas virtuais criadas

Nêutron (Redes)

Este é o módulo usado para criar componentes de rede, como redes privadas ou públicas. De acordo com a figura, foram criadas 4,9 abaixo de três redes que consistem em sub-redes privadas, compartilhadas e públicas.



The screenshot shows the OpenStack Networks interface. The left sidebar contains navigation options: Project, API Access, Compute, Volumes, Network, Network Topology, Networks (selected), Routers, Security Groups, Floating IPs, Admin, and Identity. The main content area displays a list of networks with the following columns: Name, Subnets Associated, Shared, External, Status, Admin State, Availability Zones, and Actions. Three networks are listed:

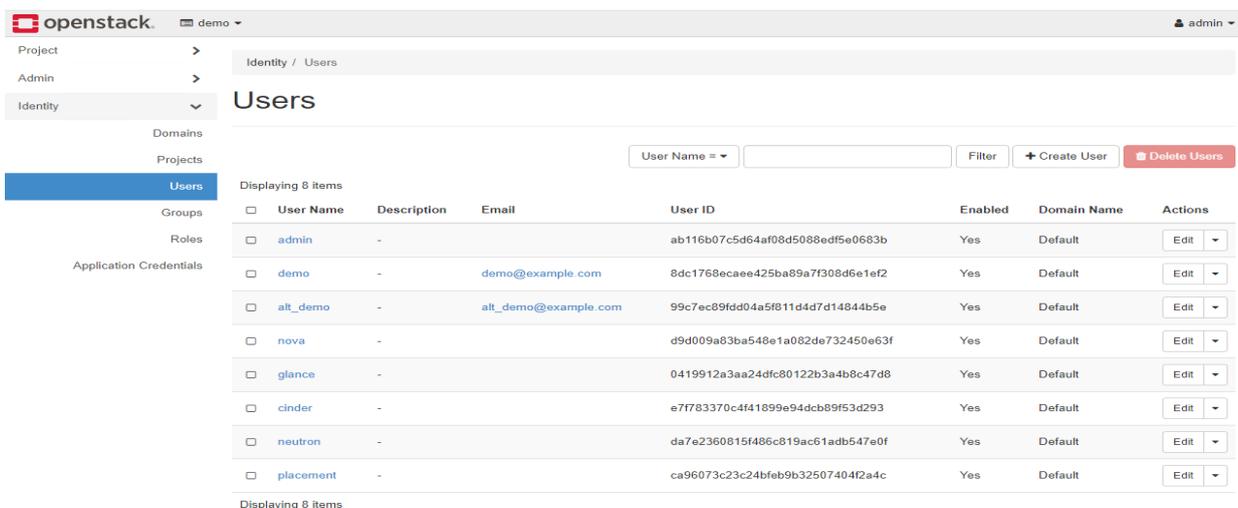
Name	Subnets Associated	Shared	External	Status	Admin State	Availability Zones	Actions
private	ipv6-private-subnet fd44:f22d:a041::/64 private-subnet 10.0.0.0/26	No	No	Active	UP	nova	Edit Network
shared	shared-subnet 192.168.233.0/24	Yes	No	Active	UP	nova	Edit Network
public	public-subnet 172.24.4.0/24 ipv6-public-subnet 2001:db8::/64	No	Yes	Active	UP	nova	Edit Network

Figura 23: Exemplo de redes criadas

Keystone (serviço de identidade)

Este é o serviço de identidade que trata da criação de usuários, bem como da autenticação e autorização.

É o que permite que um usuário seja direcionado para seu projeto ou data center virtual enquanto faça login através da GUI. Ele também controla as integrações de API. A Figura 24 mostra vários usuários criados.



The screenshot shows the OpenStack Users interface. The left sidebar contains navigation options: Project, Admin, Identity (selected), Domains, Projects, Users (selected), Groups, Roles, and Application Credentials. The main content area displays a list of users with the following columns: User Name, Description, Email, User ID, Enabled, Domain Name, and Actions. Eight users are listed:

User Name	Description	Email	User ID	Enabled	Domain Name	Actions
admin	-		ab116b07c5d64af08d5088edf5e0683b	Yes	Default	Edit
demo	-	demo@example.com	8dc1768ecaee425ba89a7f308d6e1ef2	Yes	Default	Edit
alt_demo	-	alt_demo@example.com	99c7ec89fdd04a5f811d4d7d14844b5e	Yes	Default	Edit
nova	-		d9d009a83ba548e1a082de732450e63f	Yes	Default	Edit
glance	-		0419912a3aa24dfc80122b3a4b8c47d8	Yes	Default	Edit
cinder	-		e7f783370c4f41899e94dcb89f53d293	Yes	Default	Edit
neutron	-		da7e2360815f486c819ac61adb547e0f	Yes	Default	Edit
placement	-		ca96073c23c24bfeb9b32507404f2a4c	Yes	Default	Edit

Figura 24: Serviço de identidade

5 CAPITULO V – CONSIDERAÇÕES FINAIS E RECOMENDAÇÕES

5.1 Conclusões

No final do desenvolvimento do trabalho fomos capazes de obter as seguintes conclusões e recomendações sobre os objetivos do trabalho:

A implementação de uma *IaaS* na ServiSIS, será de grande ajuda tanto para os colaboradores como administradores de infraestrutura de rede, porque através da *cloud* o acesso a serviços será melhorado e a alocação de recursos computacionais será feita em tempo real.

A implementação de uma *IaaS* na ServiSIS, facilitaria a gestão do *data center*, permitindo que tanto os usuários quanto os administradores da rede possam realizar suas atividades através de *VMs* com diversas instâncias em função das novas necessidades na regra de negocio.

De acordo com o averiguado no *data center on-premises*, da infraestrutura carece de recursos a medida em que os numero de colaboradores aumenta, o protótipo criado pode alocar dinamicamente novos usuários, o que aumentaria significativamente seu desempenho.

Treinar os administradores de rede como gerir a plataforma e os elementos necessários que devem ser criados para formar sua infraestrutura e assim a correta execução de uma instância.

Com a implantação do modelo de Infraestrutura como Serviço, os recursos *hardware* de computador, são fornecidos na forma de serviços. Com isso, a *IaaS* permite aumentar o recursos imediatamente para o respectivo crescimento do *Data Center* e usando o virtualização, você pode aumentar ou diminuir seus recursos físicos de computação durante um período de tempo muito curto.

5.2 Recomendações

Garantir a criação de uma documentação do seu *data center* de modo a facultar a realização de trabalhos futuros e auxiliar em caso de desastres a reconstruir uma infraestrutura com as mesmas configurações da anterior.

Treinar a equipa responsável por gerir a plataforma na administração de elementos necessários por ser criados para formar sua infraestrutura em *cloud* de modo a garantir a correta execução de instâncias.

Cumprimento dos pré-requisitos do Openstack o que permite ter um desempenho estável da *cloud* enquanto sua correta instalação e gestão permitirá que os usuários não tenham contratempos no uso de seus espaços virtuais conseguindo manipular recursos em tempo real.

Ter sempre material atualizado, de modo que quando for feita uma atualização da plataforma a nova versão seja compatível com os requisitos do equipamento, pois a documentação varia de acordo com a versão.

Forneça mais nós de computação se desejar executar um grande número de instâncias, já que o nó de computação é aquele que fornece os recursos de *hardware* para o instâncias.

Mantenha o número de instâncias limitado aos recursos de *hardware* do nó de computação.

5.3 Trabalhos Futuros

Como recomendações para futuros trabalhos, tem-se as seguintes sugestões:

- Expandir as práticas deste projeto de infraestrutura como serviço, adotando também a outros modelos de serviço *cloud*, para disponibilizar servidores virtuais privados direcionados para a programação de aplicações *web* em diferentes plataformas de desenvolvimento;
- Aplicar a metodologia desenvolvida na implantação deste projeto de nuvem privada com *software* livre, diminuindo os custos e viabilizando o acesso à tecnologia de computação em nuvem para empresas de pequeno e médio porte.
- esperamos que seja possível a atribuição de servidores com robustez para implementar a solução mais recente do *OpenStack* de modo a prover uma plataforma com instancias devidamente atualizadas.

5.4 Limitações

Como o trabalho tinha um período de tempo específico para ser concluído, tivemos as seguintes constrangimentos:

Limite de tempo de interação com os administradores do data center;

Pouco de acompanhamento dos administradores do *data center* da instituição;

O projeto tem apenas duas entrevistas com o cossupervisor da instituição, o tempo era limitado e os participantes para entrevista eram limitados, as entrevistas não puderam ser realizadas com todas entidades como planejado;

Falta de servidores para a realização do laboratório para demonstrações praticas das funcionalidades do projecto em *openstack*;

Complexidade de configuração do ambiente de testes devido a falta de familiaridade com comandos *Linux*;

6 REFERÊNCIAS BIBLIOGRÁFICAS

6.1 REFERÊNCIAS

AISSAOUI, M et al.(03, October 2012). **OpenStack: Toward an Open-Source Solution for Cloud Computing.** *International Journal of Computer Applications (0975 - 8887) Volume 55 – No 03.*

ALSELEK, M. (2016). **Live-Migration in Cloud Computing environment,** Instituto Superior de Engenharia do Porto Mestrado em engenharia informática Porto (outubro 2016)

ANEKE, S.(May - June 2019). **Deploying an Open Source Cloud Computing Infrastructure for Academic Research.** *IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p ISSN: 2278-8727, Volume 21, Issue 3, Ser. IV, PP 61-74*
Disponível em: www.iosrjournals.org acesso 07 de abril 2023

BAIRD, C. (2019) **A PRIMER ON CLOUD COMPUTING**, 25 de Aug 2019.

Disponível em https://medium.com/@colinbaird_51123/a-primer-on-cloudcomputing9a34e90tll303c8 Acesso em 22 Set 2023

BARKAT,A Et al.(2015). **OPEN SOURCE SOLUTIONS FOR BUILDING IAAS CLOUDS. IN: SCALABLE COMPUTING. PRACTICE AND EXPERIENCE**, vol. 16 n. 2, pp. 187-204. - ISSN 1895-1767.Desponível em <http://porto.polito.it/2654067/> acedido:10 de Fev. 2023

BORA, J., e AHMED, M. (2013). **E-LEARNING USING CLOUD COMPUTING.** *International Journal of Science and Modern Engineering (IJISME).* ISSN: 2319-6386, Vol.1 Issue-2, 2013.

BREITMAN, K; VIRTEBO, J.(2010). **COMPUTAÇÃO NA NUVEM – UMA VISÃO GERAL.** In: CONSEGI 2010 III Congresso Internacional Software Livre E Governo Eletrônico, 2010, Brasília. Amãpytuna Computação em Nuvem: Serviços Livres para a sociedade do conhecimento. Brasília: FUNAG, 2010, p. 17-45

BORGES H. P.; SOUZA J. N.; SCHULZE B.; MURY A. R. **COMPUTAÇÃO EM NUVEM**(2022)<<http://livroaberto.ibict.br/bitstream/1/861/1/COMPUTA%C3%87%C3%83O%20EM%20NUVEM.pdf/>> Acedido: 29 Set..

BROWN, A. (2019). **OPENSTACK: A FLEXIBLE SOLUTION FOR CLOUD INFRASTRUCTURE. INTERNATIONAL JOURNAL OF CLOUD COMPUTING**, 4(3), 112-125.

CATTEDDU, D.; HOGBEN, G (2009). **EUROPEAN NETWORK AND INFORMATION SECURITY AGENCY: ENISA**, 2009.

Dell. (2024). **Dell Latitude E5440 Owner's Manual**. Recuperado de [<https://www.dell.com/support/manuals/us/en/19/latitude-e5440-laptop>]

Dell. (2024). **Dell Latitude E5440 Spec Sheet**. Recuperado de <https://www.dell.com/downloads/global/products/latit/en/latitude-e5440-spec-sheet.pdf>

Disponível em <http://www.enisa.europa.eu/activities/riskmanagement/files/deliverables/cloud-computing-risk-assessment>. Acedido:25 Set. 2022

CHOU, T. (2010). **CLOUD COMPUTING: BUSINESS TRENDS AND TECHNOLOGIES**. Springer Science & Business Media.

ESIS, A; BENT, H.(2021) **ORKBOOK EXIN CLOUD COMPUTING**, Copyright © EXIN Holding B.V. Desponivel em: <https://dam.exin.com/api/englishworkbookcloudf202105.pdf>
[acesso 12 Mar. 2024](#)

FERREIRA, D.(2017). **INFRAESTRUTURA CONFIÁVEL PARA CLOUD BASEADA EM OPENSTACK**. Mestrado em Engenharia de Redes e Sistemas Informáticos, Departamento de Ciência de Computadores, Universidade do Porto, 2017

GARTNER (2013). **GARTNER IT GLOSSARY**. Disponível em <gartner.com/it-glossary> acessado a 10 nov. 2023

GIL, A. C.(2008). **COMO ELABORAR PROJETOS DE PESQUISA**. 5a. Ed. São Paulo: Atlas

GUPTA, R., & SHARMA, V. (2020). **A SYSTEMATIC LITERATURE REVIEW OF OPEN SOURCE CLOUD COMPUTING PLATFORMS**. *INTERNATIONAL JOURNAL OF INFORMATION MANAGEMENT*, 50, 206-219.

HOTA, A. Et al.(2017, Mar,07). **Comparison of Various Platforms in Cloud Computing**. *International Journal of Computer Applications (0975 – 8887)*, Volume 162.

JAVATPOINT (2021). **CLOUD COMPUTING TECHNOLOGIES**, ©Copyright 2011-2021., 2021 Disponível em: <https://www.javatpoint.com/cloud-computing-technologies> Acedido: 19 de fev. 2023

JOHNSON, L., et al. (2019). **CLOUD COMPUTING ADOPTION AND ORGANIZATIONAL PERFORMANCE: AN EMPIRICAL STUDY OF SMES IN THE UNITED STATES**. *Information & Management*, 56(3), 103167.

JONES, M., & BROWN, A. (2020). **THE STRATEGIC VALUE OF CLOUD COMPUTING: A COMPREHENSIVE ANALYSIS**. *Journal of Information Technology Management*, 21(2), 78-92.

KRUTZ, R., & VINES, R. (2010). **CLOUD SECURITY: A COMPREHENSIVE GUIDE TO SECURITY CLOUD COMPUTING**. Wiley Publishing.

[http://www.asecib.ase.ro/cc/carti/Cloud%20Computing%20Security%20\[2010\].pdf](http://www.asecib.ase.ro/cc/carti/Cloud%20Computing%20Security%20[2010].pdf)

KSENAK, A. (2016). **Analysis of different IaaS Systems within a Testing Infrastructure**, Diploma thesis, MASARYK UNIVERSITY, FACULTY OF INFORMATICS.

MANZANERO, J. S. (2015, Set). **DISEÑO E IMPLEMENTACIÓN DE UN LABORATORIO DE OPENSTACK**. Proyecto de Fin de Carrera, Universidad Carlos III De Madrid, Escuela Politécnica Superior, Ingeniería Técnica En Informática De Gestión, 121f.

MELL, P. e GRANCE, T. (2010). **THE NIST DEFINITION OF CLOUD COMPUTING**, Special Publication (NIST SP), National Institute of Standards and Technology,

Gaithersburg,MD, [online]. <https://doi.org/10.6028/NIST.SP.800-145> (Acedido em 21 Fev. de 2023)

MELO T., & AZEVEDO L.(dezembro 2019). **COMPUTAÇÃO EM NUVEM: UM ESTUDO DE CASO PARA A IMPLEMENTAÇÃO NO CCET**. Projeto de Graduação em Bacharel em Sistemas de Informação, Universidade Federal do Estado do Rio de Janeiro (UNIRIO) Rio De Janeiro, RJ – Brasil. Dezembro 2019.

Microsoft (2024). **History of Windows**. Recuperado de <https://www.microsoft.com/en-us/windows/history>

Microsoft. (2024). **Windows 10 Features**. Recuperado d (<https://www.microsoft.com/en-us/windows/features>

NIKHILA, K., RACHANA, G., BHAVANI, M. S., & AWAIZ, M. (2022). **PAPR ANALYSIS OF OFDM USING SELECTIVE MAPPING METHOD**. *Journal of Soft Computing Paradigm*, 4(3), 188-199.

Oracle. (2024). **VirtualBox**.

Recuperado de <https://www.virtualbox.org/>

PUTHAL *et al*, (2015). **Cloud Computing Features, Issues and Challenges: A Big Picture**. Disponível :<https://scholar.google.co.in/citations?user=USAID8AAAAJ&hl=en>

Saleem, M., & Rajouri, J. (2017). **CLOUD COMPUTING VIRTUALIZATION**. *International Journal of Computer Applications Technology and Research*, 6(7), 290-292.

SILVA, R. (2014). **MIGRAÇÃO E SEGURANÇA EM PLATAFORMAS CLOUD COMPUTING**. - Lisboa, 2014. 135 f. Dissertação Mestrado - Universidade Católica Portuguesa, Faculdade de Engenharia, Segurança em Sistemas de Informação, 2014.Desponovel:[https://rosario.ufma.br/jspui/bitstream/123456789/3511/1/TADEUGO N%C3%87ALVES.pdf](https://rosario.ufma.br/jspui/bitstream/123456789/3511/1/TADEUGO%20N%C3%87ALVES.pdf) acesso 10 Mar. 2023

SMITH, J. (2018). **OPEN SOURCE SOLUTIONS FOR PRIVATE CLOUD INFRASTRUCTURE**. *JOURNAL OF CLOUD COMPUTING*, 6(2), 45-57.

SOUSA, F., MOREIRA, L.& MACHADO, J. (2009). **COMPUTAÇÃO EM NUVEM: CONCEITOS, TECNOLOGIAS, APLICAÇÕES E DESAFIOS**. In: António Costa de

Oliveira; Raimundo Santos Moura; Francisco Vieira de Souza. (Org.). III Escola Regional de Computação Ceará, Maranhão e Piauí (ERCEMAPI). 1 ed. Teresina: SBC, 2009, v. 1, p. 150-175.

STOKES, D. (2013). **COMPLIANT CLOUD COMPUTING - MANAGING THE RISKS**

TAURION, C.(2009): **COMPUTAÇÃO EM NUVEM: TRANSFORMANDO O MUNDO DA DA TECNOLOGIA DA INFORMAÇÃO**. Rio de Janeiro: Brasport.

TUREL, Y.(2013). **CLOUD COMPUTING AND VIRTUALIZATION: A COMPREHENSIVE SURVEY**. Polish-Japanese Institute of Information Technology. PRZEGLĄDELEKTROTECHNICZ NY, ISSN 0033-2097, R. 89 NR 12/2013

VAZHAKKA, S. (2022). **Difference between Full Virtualization and Paravirtualization**, Last Updated : 22 Jun, 2022 Disponível em <https://www.geeksforgeeks.org/difference-between-full-virtualization-and-paravirtualization/> Acesso: 09 Marco 2023

VERAS, M. **Cloud Computing: Nova Arquitetura da TI**. Rio de Janeiro: BRASPORT, 2012.

VERDURA L. M. G. (12 Junho De 2014) **DESPLIEGUE DE UNA NUBE DE COMPUTACIÓN PRIVADA OPENSTACK EN UN ENTORNO ACADÉMICO**, Universidade Carlos III De Madrid, Escola Politécnica Superior De Engenharia De Telecomunicação De Engenharia Telemática, Engenharia De Telecomunicação,(2014).~

VIRGINIA. (2022). **ON-PREMISES VS. CLOUD DATA CENTER: WHICH IS RIGHT FOR YOUR BUSINESS**.FS Community, (2022, Mar, 21).[Online]

Disponível em: <https://community.fs.com/blog/on-premises-vs-cloud-data-center-which-is-right-for-your-business.html> acessado a 14 nov. 2023

WILLIAMS, I. (2010). **A QUICK START GUIDE TO CLOUD COMPUTING: MOVING YOUR BUSINESS**. Kogan Page, 2010.

www.dataserv-mz.com/

www.docs.openstack.org/devstack/latest/

www.javatpoint.com/cloud-computing

www.learn.microsoft.com/

www.releases.ubuntu.com/

www.servisis.co.mz/

www.virtualbox.org/

7 ANEXOS

A.1.1 Guião de entrevista



UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA

LICENCIATURA EM ENGENHARIA INFORMÁTICA

1. Já ouviu a respeito da *cloud computing*

Sim Não

2. Sua empresa utiliza a *cloud computing*?

Sim Não

3. Se sim, há quanto tempo à empresa utiliza *cloud computing*?

Menos de 1 ano De 2 a 3 anos De 4 a 5 anos Mais de 5 anos

4. Dos serviços abaixo, qual deles a sua empresa utiliza?

<input type="checkbox"/> Dropbox	<input type="checkbox"/> Microsoft 365
<input type="checkbox"/> SkyDrive	<input type="checkbox"/> Cloud Server
<input type="checkbox"/> Amazon Cloud Drive	<input type="checkbox"/> Cloud Backup
<input type="checkbox"/> Google Docs	<input type="checkbox"/> Hospedagem de sites
<input type="checkbox"/> Zoho	<input type="checkbox"/> Outlook Web App

Outros: _____.

5. Quais são as vantagens de utilizar a *cloud computing*?

<input type="checkbox"/> Acessibilidade	<input type="checkbox"/> Segurança
<input type="checkbox"/> Flexibilidade	<input type="checkbox"/> Escalabilidade
<input type="checkbox"/> Baixo custo	<input type="checkbox"/> Mobilidade
<input type="checkbox"/> Agilidade	<input type="checkbox"/> Desempenho
<input type="checkbox"/> Virtualização	<input type="checkbox"/> Disponibilidade
<input type="checkbox"/> Integração de dados e processos	<input type="checkbox"/> Outra: _____.

6. Em nível percentual até onde migraria dados da empresa para *cloud computing*?

<input type="checkbox"/> 10%	<input type="checkbox"/> 60%
<input type="checkbox"/> 20%	<input type="checkbox"/> 70%
<input type="checkbox"/> 30%	<input type="checkbox"/> 80%
<input type="checkbox"/> 40%	<input type="checkbox"/> 90%
<input type="checkbox"/> 50%	<input type="checkbox"/> 100%

7. Qual e a percentagem de serviços que a empresa migrou para a *cloud computing*?

menos de 10%

10%

20%

30%

40%

50%

60%

70%

80%

90%

100%

8. Quanto tempo foi necessário para efetuar o processo de migração?

Menos de 1 mês De 1 a 3 meses De 3 a 6 meses De 6 meses a 1 ano

Mais de 1 ano

9. Qual é o nível de satisfação da empresa com os resultados após a migração?

Insatisfação Pouca Satisfação Muita Satisfação

10. Você recomendaria a utilização da *cloud computing*?

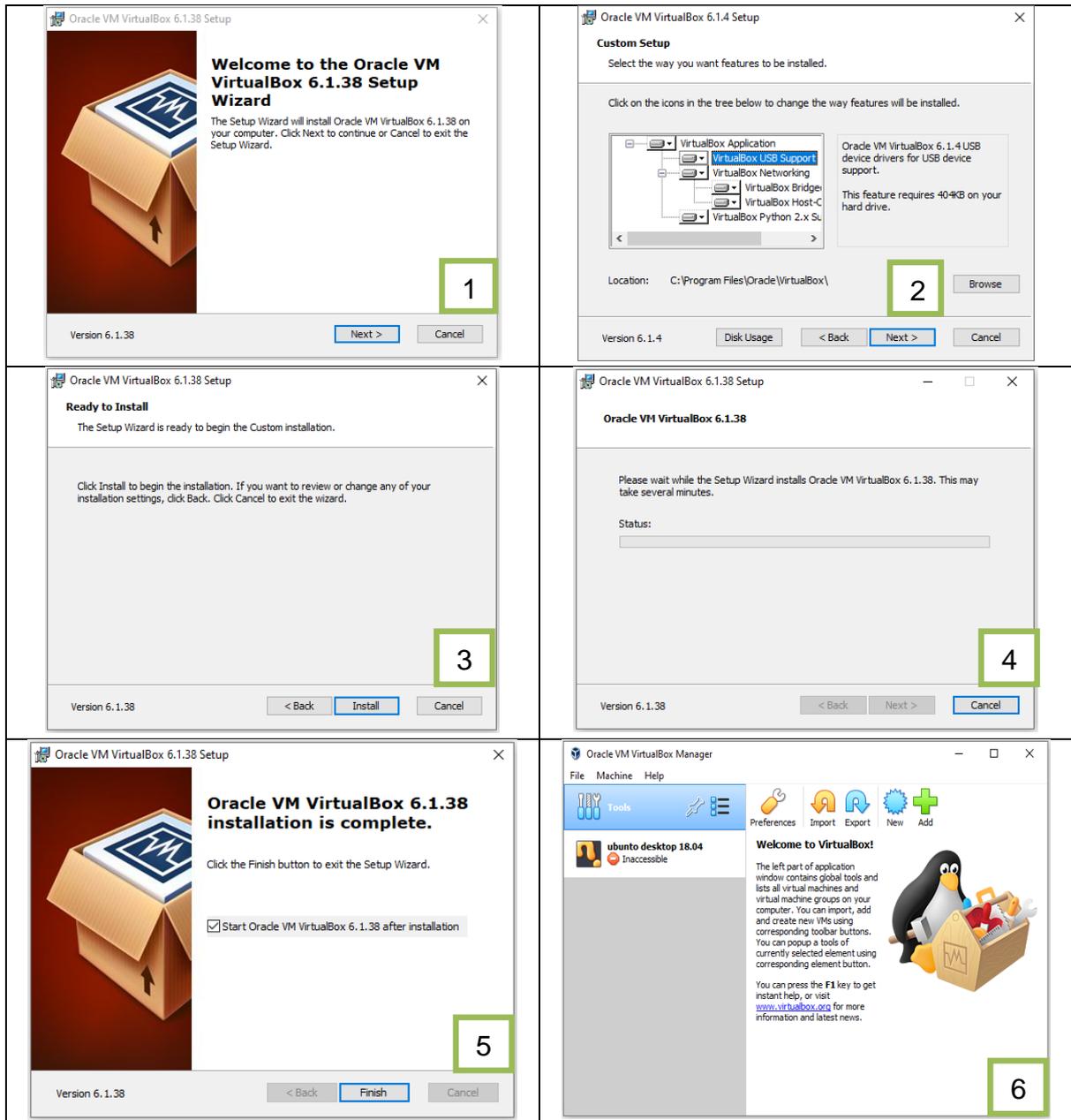
Não Recomendaria

Recomendaria com ressalvas

Recomendaria sem ressalvas

Recomendaria fortemente

A.1.2 Sequencia de passos para instalação do VirtualBox



A.1.3 Sequencia de passos para instalação SO do ambiente de teste

The sequence of steps is as follows:

- Step 1:** The main Oracle VM VirtualBox Manager window is shown. The 'Tools' menu is open, and the 'New' button is highlighted. A green box with the number '1' is in the bottom right corner.
- Step 2:** The 'Create Virtual Machine' dialog box is open, showing the 'Name and operating system' tab. The name is 'ubuntu desktop 18.04', the machine folder is 'D:\project\ubuntu desktop 18.04', the type is 'Linux', and the version is 'Ubuntu (64-bit)'. A green box with the number '2' is in the bottom right corner.
- Step 3:** The 'Memory size' dialog box is open. The recommended memory size is 1024 MB, and the slider is set to 4096 MB. A green box with the number '3' is in the bottom right corner.
- Step 4:** The 'Hard disk' dialog box is open. The 'Create a virtual hard disk now' option is selected. The recommended size is 10.00 GB. A green box with the number '4' is in the bottom right corner.
- Step 5:** The 'Create Virtual Hard Disk' dialog box is open, showing the 'File location and size' tab. The file name is 'D:\project\ubuntu desktop 18.04\ubuntu desktop 18.04' and the size is 30.00 GB. A green box with the number '5' is in the bottom right corner.
- Step 6:** The main Oracle VM VirtualBox Manager window is shown with the 'Settings' dialog box open for the 'ubuntu desktop 18.04' VM. The 'General' tab is selected, showing the VM name, operating system, system settings (4096 MB memory), display settings, storage settings, and audio settings. A green box with the number '6' is in the bottom right corner.

A.1.4 *Data Center* do caso de Estudo



Figura 25: Data center do caso de estudo

A.1.5. Exemplo de um prototipo de implementcao de IaaS em Cloud Computing

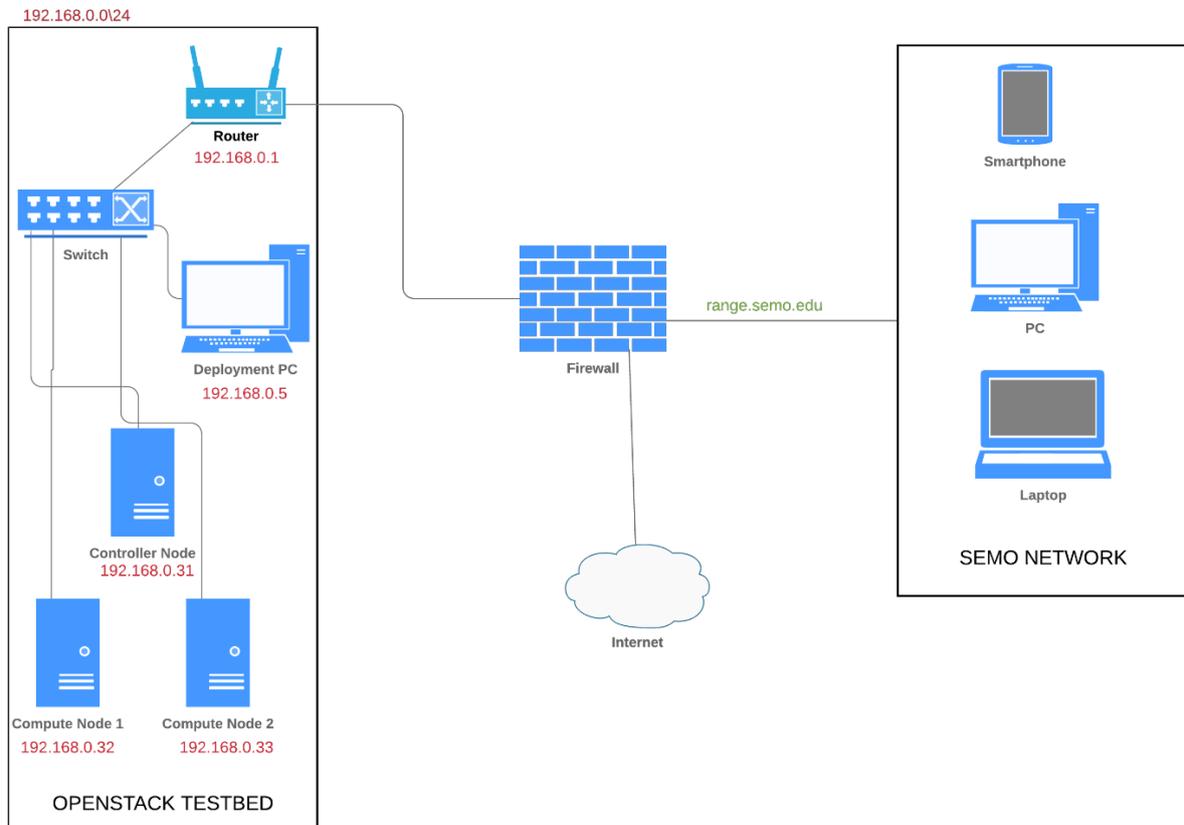


Figura 26: proposta de IaaS em Cloud

A.1.3. Foto do local de estagio



Figura 27: Local de estagio