



**UNIVERSIDADE FEDERAL DO CEARÁ  
CAMPUS QUIXADÁ  
BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

**FRANCISCO DANIEL FARIAS GRACIANO**

**GERENCIAMENTO E MONITORAMENTO DE MÁQUINAS VIRTUAIS ATRAVÉS  
DA APLICAÇÃO OPENNEBULA MOBILE MANAGER**

**QUIXADÁ**

**2016**

FRANCISCO DANIEL FARIAS GRACIANO

GERENCIAMENTO E MONITORAMENTO DE MÁQUINAS VIRTUAIS ATRAVÉS DA  
APLICAÇÃO OPENNEBULA MOBILE MANAGER

Trabalho de Conclusão de Curso Submetido à  
Coordenação do Curso de Bacharelado em  
Sistemas de Informação da Universidade  
Federal do Ceará, como requisito parcial à  
obtenção do grau de Bacharel.

Área de concentração: Computação

Orientador: Prof. Bruno Góis Mateus

QUIXADÁ

2016

Dados Internacionais de Catalogação na Publicação

Universidade Federal do Ceará

Biblioteca Universitária

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

G756g Graciano, Francisco Daniel Farias.

Gerenciamento e Monitoramento de Máquinas Virtuais Através da Aplicação OpenNebula Mobile Manager / Francisco Daniel Farias Graciano. – 2016.

46 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá, Curso de Sistemas de Informação, Quixadá, 2016.

Orientação: Prof. Me. Bruno Góis Mateus.

1.Computação em nuvem – gerenciamento. 2.Dispositivos móveis. 3. Android (Recurso eletrônico). 4. Código aberto. 5. Virtualização. I. Título.

CDD 005

---

FRANCISCO DANIEL FARIAS GRACIANO

GERENCIAMENTO E MONITORAMENTO DE MÁQUINAS VIRTUAIS ATRAVÉS DA  
APLICAÇÃO OPENNEBULA MOBILE MANAGER

Trabalho de Conclusão de Curso Submetido à  
Coordenação do Curso de Bacharelado em  
Sistemas de Informação da Universidade  
Federal do Ceará, como requisito parcial à  
obtenção do grau de Bacharel.

Área de concentração: Computação

Orientador: Prof. Bruno Góis Mateus

Aprovada em: \_\_/\_\_/\_\_\_\_.

## BANCA EXAMINADORA

---

Prof. Me. Bruno Góis Mateus (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof.<sup>a</sup> Dra. Ingrid Teixeira Monteiro  
Universidade Federal do Ceará (UFC)

---

Prof. Me. Paulo Antônio Leal Rego  
Universidade Federal do Ceará (UFC)

## **AGRADECIMENTOS**

Aos meus pais Eusali e Dijai que sempre fizeram o possível e o impossível para que eu tivesse uma boa educação, e por sempre acreditarem no meu potencial.

À minha namorada Sueleny que sempre esteve do meu lado durante todo esse período em que eu estive em Quixadá, me ajudando e me apoiando.

Aos amigos que eu fiz em Quixadá, que fizeram desse período o mais divertido e incrível possível.

Ao meu orientador Professor Bruno Góis, pela paciência e dedicação ao me orientar.

Ao professor Paulo Antônio, que me fez aprender muito durante o período em que me orientou na bolsa de iniciação acadêmica e que foi o idealizador deste projeto.

“Passei a vida inteira com medo, medo das coisas que poderiam acontecer e das que poderiam não acontecer. O que eu descobri foi que o medo é a pior parte. Esse é o verdadeiro inimigo”

(Walter White)

## RESUMO

Sabe-se que o mercado de *smartphones* anda superaquecido, por isso, é cada vez mais comum de se ver aplicações que antes só existiam para a plataforma web ou para a plataforma desktop sendo desenvolvidas para a plataforma móvel. Por conta da vida corrida que muitas pessoas levam, elas acabam optando por resolver suas atividades através de aplicações móveis sempre que possível, pois a única coisa de que elas precisam é de um *smartphone* e de uma conexão com a Internet. Outra tecnologia que está presente neste trabalho e que traz muitas vantagens para usuários e organizações, é a computação em nuvem, pois através desta tecnologia, o usuário pode ter acesso aos mais variados tipos de recursos através da internet como: recursos de armazenamento, poder de processamento, infraestrutura de rede. Tudo isso conforme sua necessidade e pagando somente pelo que ele usar. Devido à ausência de uma aplicação móvel que forneça meios para realizar o gerenciamento de nuvens OpenNebula, neste trabalho foi desenvolvida a aplicação OpenNebula Mobile Manager, que funciona como uma alternativa à ferramenta web do OpenNebula para o gerenciamento de nuvens. Este trabalho trata do desenvolvimento da aplicação citada e finaliza por validar sua usabilidade de forma satisfatória através de um método conhecido como teste de usabilidade.

**Palavras-chave:** Computação em nuvem. Android. OpenNebula.

## **ABSTRACT**

It is known that the smartphone market goes overheated, for that reason it is very common to see applications that previously only existed for the web platform or the desktop platform being developed for the mobile platform. Because of the busy life that many people take, they end up choosing to solve their activities through mobile applications whenever possible, because the only thing they need is a smartphone and an internet connection. Another technology that is present in this work and it brings many advantages for users and organizations is cloud computing, because through this technology, the user can have access to all kinds of resources over the internet as storage resources, power processing network infrastructure. All this as you need and only pay for what he uses. With this in mind, it has been developed the application of cloud management named OpenNebula Mobile Manager, which works as an alternative to web tool OpenNebula Sunstone. This work is about the aforementioned application development and ends to validate its usability satisfactorily by a method known as usability testing.

**Keywords:** Cloud Computing. Android. OpenNebula.

## LISTA DE FIGURAS

Figura 1- Representação das Camadas da Nuvem.....	17
Figura 2 - Comando onevm list .....	20
Figura 3 – Listando máquinas virtuais através do OpenNebula Sunstone .....	21
Figura 4 – Arquitetura do OpenNebula .....	22
Figura 5 - Mercado de Sistemas Operacionais para <i>Smartphones</i> .....	23
Figura 6 - Atividades do teste de usabilidade .....	26
Figura 7 - Diagrama de estados de uma máquina virtual .....	28
Figura 8 - Comunicação da aplicação com a nuvem do OpenNebula.....	29
Figura 9 - Tela de Login .....	30
Figura 10 - Tela Listar Máquinas Virtuais .....	31
Figura 11 - Tela Instanciar Máquinas Virtuais.....	32
Figura 12 - Tela Criar Novo Usuário .....	33
Figura 13 – Tela de Detalhes da Máquina Virtual .....	34
Figura 14 – Tela de Configurações de Monitoramento .....	35
Figura 15 – Gráfico relacionado à facilidade de utilização .....	38
Figura 16 - Gráfico relacionado à organização das informações .....	38
Figura 17 – Gráfico relacionado ao layout das telas .....	39
Figura 18 – Gráfico relacionado à nomenclatura utilizada nas telas .....	40
Figura 19 - Gráfico relacionado à assimilação das informações .....	40
Figura 20 - Gráfico relacionado ao nível de satisfação ao utilizar a aplicação .....	41
Figura 21 - Visão geral sobre o resultado do teste de usabilidade.....	41

## **LISTA DE TABELAS**

Tabela 1 - Perfil dos Participantes do teste de usabilidade.....	36
--	----

## SUMÁRIO

1 INTRODUÇÃO .....	14
2 FUNDAMENTAÇÃO TEÓRICA .....	15
2.1 Virtualização .....	15
2.2 Computação em Nuvem .....	16
2.3 OpenNebula .....	19
2.4 Android .....	22
3 TRABALHOS RELACIONADOS .....	23
4 PROCEDIMENTOS METODOLÓGICOS .....	24
4.1 Realizar Análise de Requisitos .....	24
4.2 Preparar o ambiente de testes do OpenNebula .....	24
4.3 Explorar a API do OpenNebula .....	25
4.4 Desenvolver a aplicação .....	25
4.5 Testar e Verificar a usabilidade da aplicação .....	25
5 A APLICAÇÃO OPENNEBULA MOBILE MANAGER .....	26
5.1 Requisitos da Aplicação.....	26
5.2 Comunicação da Aplicação com a Nuvem do OpenNebula.....	28
5.3 Telas da Aplicação .....	29
6 TESTE DE USABILIDADE .....	35
6.1 Preparação .....	35
6.2 Coleta de Dados.....	36
6.3 Interpretação dos Resultados .....	37
6.4 Relato dos Resultados.....	37
7 CONCLUSÃO E TRABALHOS FUTUROS .....	42
REFERÊNCIAS .....	44
APÊNDICE A – TERMO DE CONSENTIMENTO .....	45
APÊNDICE B – CENÁRIO DO TESTE .....	46
APÊNDICE C – INSTRUMENTO DE COLETA DE DADOS .....	47

## 1 INTRODUÇÃO

A computação em nuvem é uma tendência tecnológica que tem crescido bastante nos últimos anos. Seu conceito se baseia no fornecimento de recursos de TI sob demanda através da internet, e apresenta uma série de vantagens para usuários e organizações, por exemplo: com o uso da computação em nuvem o usuário deixa a preocupação com a parte de instalação e manutenção de equipamentos para os provedores, podendo assim direcionar esforços exclusivamente para o que é pertinente ao seu negócio (BORGES et al, 2011).

A computação em nuvem está diretamente ligada à virtualização, que consiste numa tecnologia capaz de simular vários ambientes computacionais ou máquinas virtuais em uma mesma máquina física. Esta tecnologia não é muito recente, ela é utilizada desde a década de 1960 nos *mainframes*. A virtualização de sistemas computacionais proporciona uma redução significativa de custos se comparado à aquisição de sistemas físicos, e facilita a administração, além de reduzir o consumo de energia (SOTO, 2011).

Provedores e usuários de nuvens computacionais monitoram e gerenciam o uso de recursos por meio de sistemas de monitoramento (COUTINHO et al, 2013). Na Universidade Federal do Ceará (UFC) - Campus Quixadá, é utilizado o OpenNebula para o gerenciamento da nuvem computacional do campus. O OpenNebula consiste em um conjunto de ferramentas de código aberto que são usadas para o gerenciamento de nuvens computacionais. Dentre elas, destacamos a OpenNebula Sunstone, que é uma ferramenta de gerenciamento de máquinas virtuais para a plataforma web, que simplifica o gerenciamento dos recursos da nuvem do OpenNebula através de uma interface gráfica.

Devido ao crescente uso de *smartphones* e da praticidade em se utilizar uma ferramenta móvel, surge a demanda de desenvolver uma aplicação para dispositivos móveis que realize o gerenciamento e o monitoramento da nuvem do OpenNebula permitindo que através desta aplicação, o usuário possa executar as principais funcionalidades existentes na ferramenta web. Por conta disso, foi realizada uma pesquisa na *Google Play*<sup>1</sup> em busca de alguma aplicação que dispusesse de funcionalidades semelhantes as funcionalidades da aplicação que será descrita mais a frente neste trabalho, e foi encontrada somente uma aplicação com características similares, mas que não funcionava de maneira correta, e isto motivou ainda mais o desenvolvimento da aplicação proposta neste trabalho.

---

<sup>1</sup> <https://play.google.com/store>

O objetivo deste trabalho é desenvolver uma aplicação para dispositivos móveis que permita o gerenciamento e o monitoramento das máquinas virtuais da nuvem do OpenNebula, e que possua as principais funcionalidades existentes no OpenNebula Sunstone, e apresente uma boa usabilidade.

A sequência deste trabalho está organizada da seguinte forma: Na seção 2, são discutidos alguns conceitos que ajudam a fundamentar este trabalho. Na seção 3, é discutido sobre as similaridades e diferenças de alguns trabalhos relacionados a este. Na seção 4, é explicado o passo a passo do desenvolvimento da aplicação desenvolvida neste trabalho. Na seção 5 é discutido mais a fundo sobre a aplicação OpenNebula Mobile Manager. Na seção 6 é explicado sobre o teste de usabilidade, que foi o método utilizado para validar a aplicação e finalmente na seção 7, temos a conclusão deste trabalho e o que está planejado para os trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

Esta seção define alguns conceitos que são importantes para fundamentar o desenvolvimento deste trabalho.

### 2.1 Virtualização

Virtualização é a técnica de criar uma versão virtual de algum recurso computacional. A virtualização pode ser utilizada para virtualizar computadores, sistemas operacionais, dispositivos de armazenamento, aplicativos, redes, mas o seu principal uso é para a virtualização de servidores.

A virtualização faz o uso de um software que simula a existência de um hardware, criando uma máquina virtual com todas as características de uma máquina real. O termo máquina virtual é utilizado para descrever um ambiente computacional abstrato que é criado a partir de um monitor de máquinas virtuais ou *hypervisor*. Um *hypervisor* pode criar uma ou mais máquinas virtuais sobre uma mesma máquina física (CHOINACKI, 2012). No contexto do OpenNebula, uma máquina virtual é representada por um *template*<sup>2</sup>, que segundo o OpenNebula, é um arquivo que contém um conjunto de atributos que definem uma máquina virtual.

---

<sup>2</sup> <http://docs.opennebula.org/4.12/user/references/template.html>

De acordo com a VMWARE<sup>3</sup>, com a técnica da virtualização, as empresas podem executar várias máquinas virtuais em uma mesma máquina física, sendo assim, o limite para a execução destas máquinas é o limite dos recursos que a máquina física possui, resultando em um maior aproveitamento dos recursos computacionais e em uma grande economia para as organizações. Um exemplo de software de virtualização que podemos citar é o VirtualBox<sup>4</sup>, que consiste em um software que permite a instalação e o uso de um sistema operacional dentro de outro, ou seja, dois ou mais computadores independentes, mas que compartilham o mesmo hardware. A virtualização foi a tecnologia que tornou possível fazer computação em nuvem.

## **2.2 Computação em Nuvem**

A Computação em nuvem é um paradigma da computação e seu conceito se baseia no fornecimento de recursos de TI através da internet e sob demanda, ou seja, o usuário somente irá pagar pelos recursos que ele utilizar. Com o uso da computação em nuvem o usuário deixa a preocupação com a parte de instalação e manutenção de equipamentos para os provedores, podendo assim direcionar esforços exclusivamente para o que é pertinente ao seu negócio (BORGES et al, 2011).

O objetivo da computação em nuvem é o de fornecer serviços de baixo custo e de alta disponibilidade, e garantir que usuários e organizações não precisem mais se preocupar com seus dados, pois em qualquer hora e em qualquer lugar seus dados estarão disponíveis através da internet (OLIVEIRA et al, 2013). A nuvem pode ser entendida como uma abstração para a internet e para toda a infraestrutura que a compõe.

### **2.2.1 Modelos de Serviço da Nuvem**

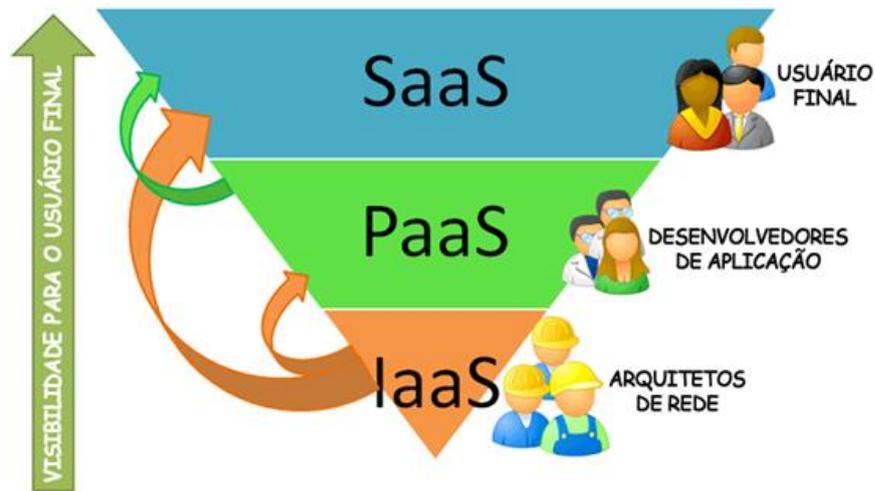
A arquitetura da computação em nuvem é dividida em três camadas, como pode ser visto na Figura 1, são elas: *Software* como Serviço (*Software as a Service* - SAAS), Plataforma como Serviço (*Platform as a Service* - PAAS) e Infraestrutura como Serviço (*Infrastructure as a Service* - IAAS) e cada camada é responsável por prestar um tipo de serviço ao usuário.

---

<sup>3</sup> <http://www.vmware.com/br/virtualization/how-it-works>

<sup>4</sup> <https://www.virtualbox.org/>

Figura 1- Representação das Camadas da Nuvem



Fonte: Vitor Meriat (2011)

#### 2.2.1.1 Software como Serviço (SAAS)

SAAS corresponde a camada mais externa da arquitetura e é responsável por disponibilizar aplicações completas através da internet para o usuário final (OLIVEIRA et al, 2013). Estas aplicações são disponibilizadas por prestadores de serviço que podem ou não cobrar uma taxa de utilização e podem ser acessadas através da internet. Logo, estas aplicações estarão disponíveis em qualquer local bastando o usuário possuir um dispositivo que faça acesso à Internet. Segundo Soto (2011, p.17): “Desta forma, dispositivos móveis podem ter acesso via *web services* a programas sofisticados e que necessitam de grande capacidade de processamento para sua execução”. Um exemplo de SAAS que pode ser citado é a ferramenta Word online<sup>5</sup>.

#### 2.2.1.2 Plataforma como Serviço (PAAS)

A camada de PAAS é a intermediária da arquitetura, ela é voltada aos desenvolvedores. Esta camada oferece todo um ambiente onde o desenvolvedor pode criar e testar aplicações diretamente na nuvem sem se preocupar com a infraestrutura por trás disso tudo.

<sup>5</sup> <https://office.live.com/start/Word.aspx>

### 2.2.1.3 Infraestrutura como Serviço (IAAS)

Finalmente chegamos à camada mais baixa da arquitetura. O IAAS é responsável por fornecer ao usuário recursos de *hardware* virtualizados sob demanda, como: poder de processamento, armazenamento de dados, infraestrutura de rede, entre outros recursos. Esta camada é quem fornece a infraestrutura responsável por manter as outras duas camadas superiores a essa.

As vantagens para usuários e organizações que trabalham com IAAS são enormes. Segundo Borges et al (2011, p.9): “Algumas vantagens de se trabalhar com IAAS são:

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

O OpenNebula é um exemplo de provedor de IAAS.

## 2.2.2 Modelos de Implantação da Nuvem

São três os modelos de implantação conhecidos: nuvem pública, nuvem privada e nuvem híbrida. A escolha entre um modelo ou outro deve ser feita baseado no tipo de aplicação que a organização pretende usar, no nível de acesso que será disponibilizado para os usuários desta organização, dentre outros fatores. A seguir será descrito um pouco sobre cada modelo de implantação mencionado anteriormente.

### 2.2.2.1 Nuvem Pública

Uma nuvem pública pertence a um provedor que vende seus serviços para usuários e organizações que compartilham desta mesma nuvem. Um usuário de uma nuvem pública

possui total controle sobre o que ele faz na nuvem, mas não possui controle sobre o que os outros usuários fazem neste ambiente. Por isso, este modelo é recomendado para organizações que buscam reduzir custos e que não precisem de tanta segurança sobre seus dados.

#### 2.2.2.2 *Nuvem Privada*

A infraestrutura do modelo de nuvem privada é ou comprada ou alugada por uma única organização, e é exclusivamente operada pela mesma. A nuvem pode ser de acesso local ou remoto e são aplicadas políticas de acesso aos serviços de acordo com as restrições da empresa (BORGES et al, 2011). Por isto, este modelo de implantação é mais adequado para organizações que desejam ter mais segurança e um maior controle sobre o acesso a seus dados.

#### 2.2.2.3 *Nuvem Híbrida*

O modelo de nuvem híbrida é a junção da nuvem pública com a nuvem privada, ou seja, incorpora o que tem de melhor em cada um dos dois modelos. Por exemplo: toda a parte de sistemas que envolve dados sigilosos de uma organização pode ficar hospedado numa nuvem privada e já os sistemas que não necessitam de tanta segurança podem ficar hospedados numa nuvem pública.

Este modelo de nuvem é ideal para organizações que desejam uma maior flexibilidade no que diz respeito aos seus dados, pois a organização pode ter o controle do que pode ser acessado dentro e fora dela.

### **2.3 OpenNebula**

Segundo Santos (2013, p.27), “O OpenNebula é uma solução de código livre para um *datacenter* de virtualização, altamente escalável e adaptável para o gerenciamento de dados virtualizados do tipo IaaS”. O OpenNebula é a parte central deste trabalho, pois será ele o responsável pela criação e o gerenciamento das máquinas virtuais através da integração da aplicação com a OpenNebula Cloud API (OCA).

A primeira versão do OpenNebula<sup>6</sup> surgiu em 2005 como um projeto de pesquisa desenvolvido por M. Llorente e Rubén S. Montero. Desde a sua primeira versão pública do *software* em março de 2008, ela vem evoluindo através de versões de código aberto e agora opera como um projeto de *software* livre.

Vale ressaltar que possuem outras soluções similares ao OpenNebula e que são de código aberto, como é o caso da nuvem da OpenStack<sup>7</sup>. Porém, o OpenNebula foi a nuvem escolhida para o desenvolvimento deste trabalho, por ser utilizada na UFC – Campus Quixadá, facilitando assim a realização dos testes por parte do autor deste trabalho e por proporcionar uma maior aproximação com os especialistas deste tipo de nuvem.

O OpenNebula dispõe de duas interfaces voltadas para administradores da nuvem, são elas, *Command Line Interface* (CLI) e *OpenNebula Sunstone*. A seguir será falado um pouco sobre estas duas interfaces.

### 2.3.1 Interfaces do OpenNebula

- **Command Line Interface (CLI)**

CLI consiste em uma interface para o OpenNebula, aonde é possível gerenciar a nuvem através de comandos de texto em um console. A figura 2 ilustra a execução do comando *onevm list* que é responsável por listar as máquinas virtuais existentes no OpenNebula.

Figura 2 - Comando *onevm list*

```

$ onevm list
ID      NAME  STAT CPU    MEM    HOSTNAME      TIME
0       VM1  prol  0      0      192.168.0.73  00 0:00:06

```

Fonte: Julio Alba Soto (2011)

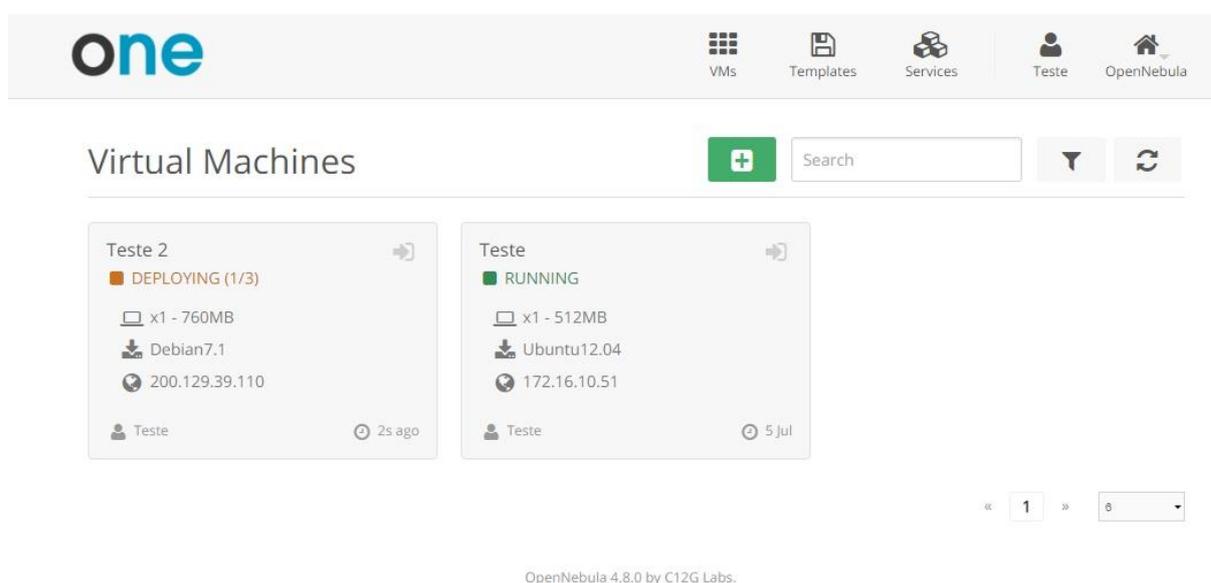
- **OpenNebula Sunstone**

<sup>6</sup> <http://opennebula.org/>

<sup>7</sup> <https://www.openstack.org/software/>

Segundo o OpenNebula<sup>8</sup>, o OpenNebula Sunstone consiste em uma interface gráfica para a plataforma web que foi desenvolvida para usuários menos experientes e para administradores da nuvem e que simplifica o gerenciamento das operações em nuvens privadas e híbridas. Tudo que é feito através do CLI pode ser feito através do OpenNebula Sunstone. A figura 3 apresenta a tela de listagem de máquinas virtuais através do OpenNebula Sunstone.

Figura 3 – Listando máquinas virtuais através do OpenNebula Sunstone



Fonte: Próprio autor

### 2.3.2 Arquitetura do OpenNebula

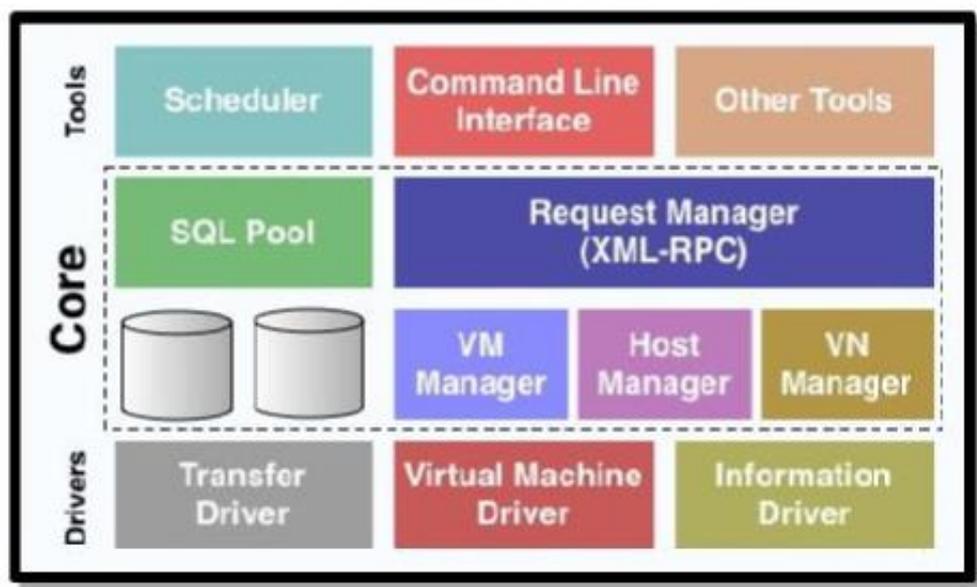
Como podemos ver na Figura 4, a arquitetura do OpenNebula é dividida em 3 camadas, são elas:

- Camada de Ferramentas: Esta camada contém ferramentas distribuídas com o OpenNebula, como é o caso da CLI (Command Line Interface) e do *Scheduler*;
- Camada de Núcleo: A camada de núcleo consiste em um conjunto de componentes para controlar e monitorar máquinas virtuais, redes virtuais, armazenamento e hosts. O gerenciamento das máquinas virtuais, redes virtuais e dos dispositivos de armazenamento é feito nesta camada através da chamada ao driver adequado.

<sup>8</sup> [http://docs.opennebula.org/4.12/administration/sunstone\\_gui/sunstone.html](http://docs.opennebula.org/4.12/administration/sunstone_gui/sunstone.html)

- Camada de Drivers: Esta camada é responsável por incorporar diferentes tecnologias de virtualização, armazenamento e monitoramento ao OpenNebula.

Figura 4 – Arquitetura do OpenNebula



Fonte: Xiaolin Li, Judy Qiu (2014)

## 2.4 Android

O Android é um sistema operacional que foi desenvolvido inicialmente para dispositivos móveis, como *smartphones* e *tablets*, mas hoje pode ser encontrado em vários outros dispositivos como relógios, TV's e até geladeiras. Possui código aberto e é baseado no kernel do Linux. Sua primeira versão comercial foi lançada em 2008 pela Google e segundo a IDC (2015), o Android já é o sistema operacional para dispositivos móveis mais utilizado no mundo, com uma fatia de 82,8 % do mercado internacional de *smartphones*, como pode ser visto na Figura 5. Por esta razão é que a plataforma móvel Android foi a escolhida para o desenvolvimento da aplicação que foi desenvolvida neste trabalho.

Figura 5 - Mercado de Sistemas Operacionais para *Smartphones*

Period	Android	iOS	Windows Phone	BlackBerry OS	Others
2015Q2	82.8%	13.9%	2.6%	0.3%	0.4%
2014Q2	84.8%	11.6%	2.5%	0.5%	0.7%
2013Q2	79.8%	12.9%	3.4%	2.8%	1.2%
2012Q2	69.3%	16.6%	3.1%	4.9%	6.1%

Fonte: IDC (2015)

### 3 TRABALHOS RELACIONADOS

Esta seção apresenta dois trabalhos relacionados a este, e discute sobre o foco principal dos trabalhos, suas similaridades e diferenças com este trabalho.

A proposta principal do trabalho de Soto (2011) é mostrar uma análise da ferramenta de gerenciamento de nuvem OpenNebula. O autor também faz uma rápida análise sobre outras soluções similares ao OpenNebula e que também são de código aberto e depois faz um comparativo entre estas ferramentas. O autor também demonstra toda a parte de instalação, configuração e gerenciamento da nuvem do OpenNebula.

O trabalho de Soto (2011) se assemelha a este trabalho pelo fato de o autor realizar o gerenciamento das máquinas virtuais do OpenNebula.

Por outro lado, o autor realiza o gerenciamento das máquinas virtuais através do CLI (*Command Line Interface*), que é uma interface utilizada para usuários da nuvem mais avançados, enquanto que este trabalho propõe que seja realizado o gerenciamento das máquinas virtuais através de um dispositivo móvel com uma interface amigável e intuitiva que possa ser utilizado por qualquer usuário do OpenNebula, independente do seu nível de entendimento sobre a ferramenta.

No trabalho desenvolvido pelos autores Nuñez et al. (2010), eles apresentam uma API REST (*Representational State Transfer*) desenvolvida por eles que facilita o gerenciamento das máquinas virtuais da nuvem do OpenNebula. Os autores fazem o uso da UML e demonstram através de diagramas de sequência como a API desenvolvida por eles interage com a arquitetura do OpenNebula e posteriormente vão demonstrando as funcionalidades que a API possui.

A semelhança que o trabalho dos autores Nuñez et al. (2010) tem com este trabalho é que eles apresentam uma alternativa simplificada de realizar o gerenciamento da nuvem do OpenNebula.

A diferença é novamente a forma como é realizado o gerenciamento da nuvem, no caso do trabalho dos autores citados anteriormente é através de uma API desenvolvida por eles, e no caso deste trabalho, o autor usa a API oficial do OpenNebula integrada com uma aplicação móvel para realizar o gerenciamento da nuvem do OpenNebula.

## **4 PROCEDIMENTOS METODOLÓGICOS**

Esta seção descreve as etapas para a realização deste trabalho:

### **4.1 Realizar Análise de Requisitos**

Nesta primeira etapa do trabalho foi realizada a análise de requisitos da aplicação, os requisitos foram definidos através de conversas com os especialistas da nuvem do OpenNebula da Universidade Federal do Ceará – Campus Quixadá. Foi-lhes perguntado sobre as funcionalidades mais utilizadas no OpenNebula Sunstone e que não poderiam faltar numa aplicação mobile, e através destas conversas chegou-se a um conjunto de requisitos que podem ser vistos na seção 5.

### **4.2 Preparar o ambiente de testes do OpenNebula**

O primeiro passo desta etapa do trabalho foi a instalação do *software* de virtualização VirtualBox, que permite que sejam instalados diversos sistemas operacionais em uma mesma máquina física. O segundo passo foi baixar do site oficial do OpenNebula<sup>9</sup> a imagem de uma máquina virtual com o sistema operacional CentOS 6.3 com o software do OpenNebula 4.14.2 instalado e pré-configurado. E finalmente, depois de baixada a imagem da máquina virtual, ela foi carregada no VirtualBox, e assim os testes puderam ser iniciados.

---

<sup>9</sup> <http://opennebula.org/tryout/sandboxvirtualbox/>>.

### 4.3 Explorar a API do OpenNebula

Nesta etapa do trabalho foi utilizada a IDE (*Integrated Development Environment*) Android Studio<sup>10</sup> juntamente com a OCA (*OpenNebula Cloud API*), a API Java do OpenNebula, para que fossem realizados testes com a nuvem instalada no passo anterior. Assim foi possível obter um primeiro contato com a API do OpenNebula e foi possível descobrir de quais funcionalidades a API dispõe.

### 4.4 Desenvolver a aplicação

Para o desenvolvimento da aplicação foi utilizada a IDE Android Studio, para o controle de versão foi utilizado o Git juntamente com o Github<sup>11</sup>, o Git é um sistema que tem a função de gerenciar as várias versões do código de um projeto de software. O uso do Git é importante porque quando surgir um problema no código do projeto, sempre será possível retornar para uma versão anterior e estável do código. E o Github é um repositório público de código aonde o código deste projeto foi armazenado<sup>12</sup>.

### 4.5 Testar e Verificar a usabilidade da aplicação

Com a aplicação finalizada, é importante verificar a usabilidade da mesma junto a alguns usuários alvo do sistema. O método escolhido para verificar a usabilidade da aplicação foi o método conhecido como “Teste de Usabilidade”. Este método tem o objetivo de avaliar a usabilidade de sistemas interativos através da observação de usuários alvo realizando tarefas típicas do sistema. (BARBOSA; SILVA, 2010)

Para realizar a avaliação, um grupo de usuários é convidado a executar um conjunto de tarefas usando o sistema a ser avaliado. No entanto, é importante que os usuários que utilizarão o sistema estejam em um ambiente que seja propício a coleta de dados, pois serão estes dados que serão utilizados para verificar a usabilidade da aplicação e assim, dependendo

---

<sup>10</sup> <https://developer.android.com/studio/index.html>

<sup>11</sup> <https://github.com/>

<sup>12</sup> <https://github.com/danielfarias09/OpenNebulaMobileManager2>

dos resultados obtidos, correções poderão ser feitas no sistema. A Figura 6 apresenta as atividades realizadas no teste de usabilidade.

Figura 6 - Atividades do teste de usabilidade

<b>Atividade</b>	<b>Tarefa</b>
Preparação	<ul style="list-style-type: none"> <li>● definir tarefas para os participantes executarem</li> <li>● definir o perfil dos participantes e recrutá-los</li> <li>● preparar material para observar e registrar o uso</li> <li>● executar um teste piloto</li> </ul>
Coleta de dados	<ul style="list-style-type: none"> <li>● observar e registrar a performance e a opinião dos participantes durante sessões de uso controladas</li> </ul>
Interpretação e consolidação dos resultados	<ul style="list-style-type: none"> <li>● reunir, contabilizar e sumarizar os dados coletados dos participantes</li> </ul>
Relato dos resultados	<ul style="list-style-type: none"> <li>● relatar a performance e a opinião dos participantes</li> </ul>

Fonte: Simone Diniz Junqueira Barbosa, Bruno Santana da Silva (2010)

## **5 A APLICAÇÃO OPENNEBULA MOBILE MANAGER**

A OpenNebula Mobile Manager é uma aplicação para a plataforma Android que auxilia no gerenciamento da nuvem do OpenNebula. Ela foi desenvolvida com o propósito de trazer as principais funcionalidades do OpenNebula Sunstone para a plataforma móvel Android.

Esta seção contém detalhes sobre os requisitos da aplicação, como ela foi desenvolvida e as tecnologias utilizadas no seu desenvolvimento.

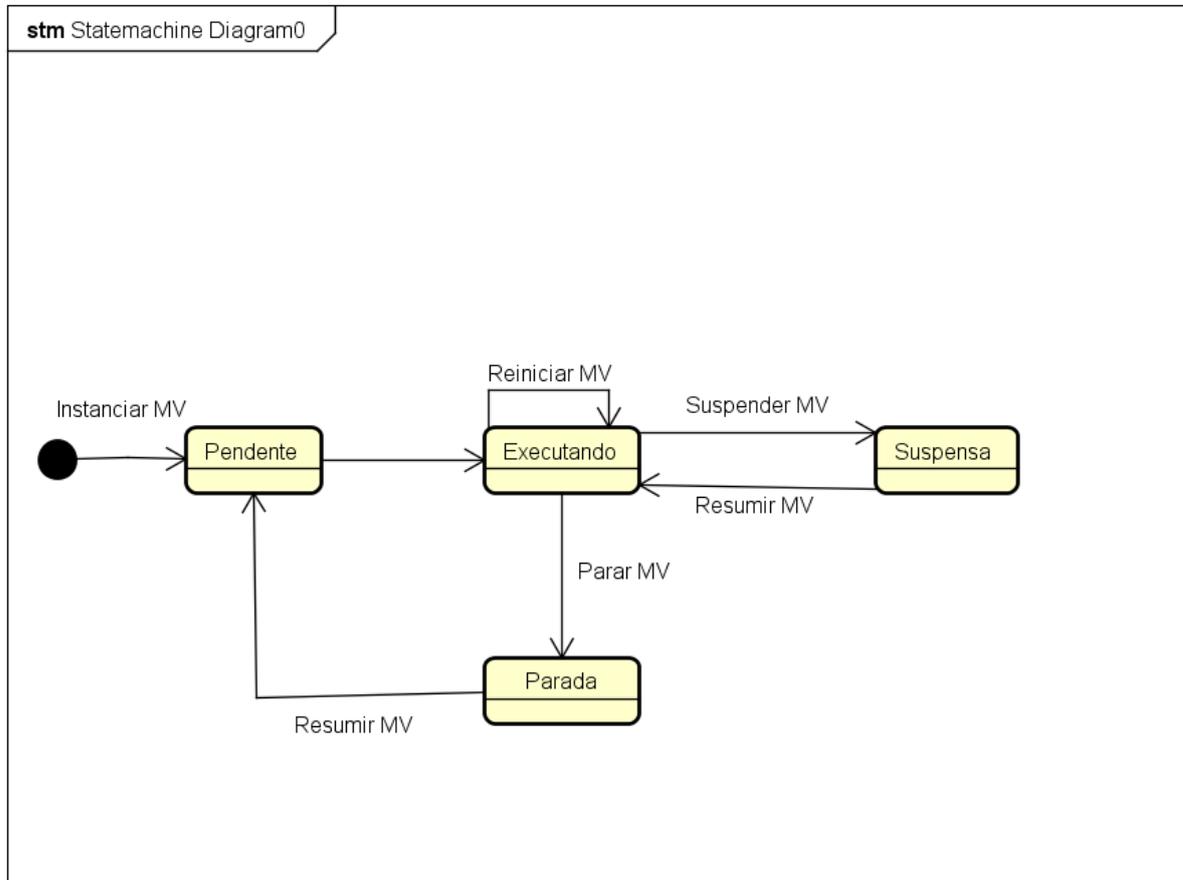
### **5.1 Requisitos da Aplicação**

Antes que seja discutido sobre as funcionalidades da aplicação, é importante saber que uma máquina virtual no contexto do OpenNebula pode assumir vários estados, mas no contexto deste trabalho, iremos tratar somente de 4 destes estados, que são eles: Pendente, Executando, Suspensa e Parada. A Figura 7 ilustra estes estados e suas transições. Sabendo

disto, agora será discutido sobre as funcionalidades da aplicação OpenNebula Mobile Manager.

- **Instanciar Máquina Virtual:** O usuário pode criar uma nova instância de uma máquina virtual escolhendo um *template* existente e um nome para a MV.
- **Criar Novo Usuário:** O usuário pode criar um novo usuário que terá acesso aos recursos do OpenNebula através da escolha de um nome de usuário e de uma senha.
- **Listar Máquinas Virtuais:** O usuário pode listar todas as máquinas virtuais criadas por ele anteriormente.
- **Ver Detalhes de uma Máquina Virtual:** O usuário tem acesso a informações como: nome, estado, identificador (ID), status, IP, quantidade de memória, VCPUs (CPUs virtuais), endereço MAC e tamanho do disco alocados para a máquina virtual.
- **Remover Máquina Virtual:** A máquina virtual é removida e desalocada.
- **Parar Máquina Virtual:** A máquina virtual para as suas funções, perde seu IP e volta para seu estado inicial. E quando resumida, a máquina virtual inicia como se tivesse acabado de ser instanciada.
- **Suspender Máquina Virtual:** A máquina virtual para as suas funções, mas seus arquivos e seu estado serão salvos. E quando resumida, a máquina virtual retoma as suas atividades de onde parou.
- **Reiniciar Máquina Virtual:** A máquina virtual é reiniciada.
- **Resumir Máquina Virtual:** A máquina virtual retoma suas atividades depois de parada ou suspensa.
- **Monitorar Máquinas virtuais:** Através desta função o usuário pode ser notificado sempre que novas máquinas virtuais forem criadas ou removidas da sua conta. E no caso de o usuário ser um usuário administrador ele pode monitorar a criação e remoção de máquinas virtuais de todos os usuários administrados por ele.

Figura 7 - Diagrama de estados de uma máquina virtual

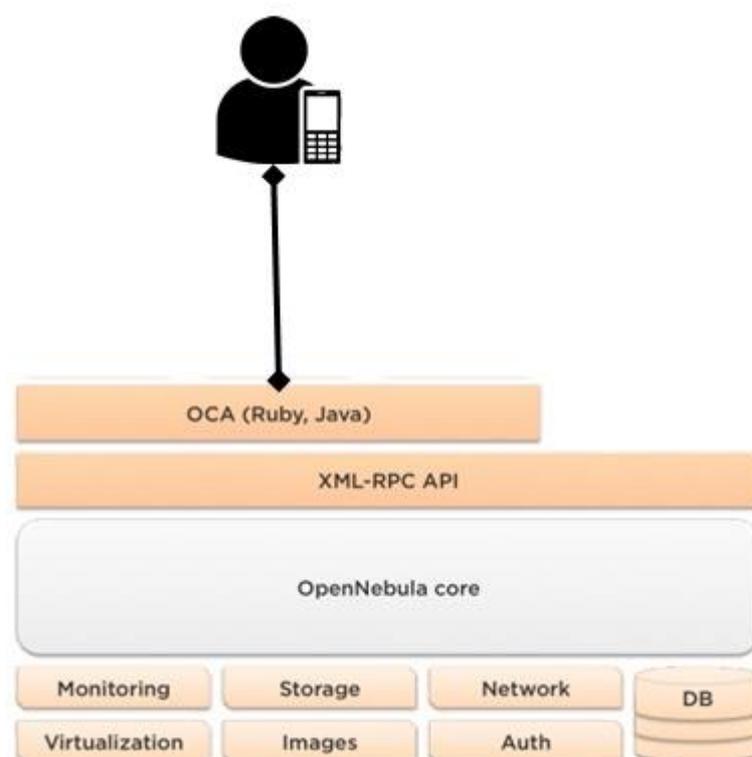


Fonte: Próprio autor

## 5.2 Comunicação da Aplicação com a Nuvem do OpenNebula

A aplicação OpenNebula Mobile Manager utiliza a OCA (OpenNebula *Cloud* API) para controlar a nuvem do OpenNebula e para ter acesso aos seus recursos. A OCA fornece uma maneira simplificada e conveniente de realizar uma interface com o núcleo do OpenNebula. A comunicação com a nuvem é feita através da API do OpenNebula, e funciona da seguinte maneira, sempre que uma funcionalidade da API é utilizada, a API acessa a camada de núcleo, que é responsável por buscar o driver adequado para tratar esta requisição. A Figura 8 demonstra como é feita a comunicação entre a aplicação e a nuvem do OpenNebula.

Figura 8 - Comunicação da aplicação com a nuvem do OpenNebula



Fonte: Próprio autor

### 5.3 Telas da Aplicação

Esta seção tem o objetivo de apresentar as telas da aplicação OpenNebula Mobile Manager e explicar o funcionamento de cada uma.

#### 5.3.1 Tela de Login

Nesta primeira tela ilustrada pela Figura 9, o usuário deverá entrar com seu nome de usuário, senha e o *end point*, sendo este último o endereço da nuvem do OpenNebula que o usuário quer ter acesso. Se todos os dados estiverem corretos, o usuário será direcionado para a tela “Listar Máquinas Virtuais” mostrada na Figura 10.

Figura 9 - Tela de Login



OpenNebula Mobile Manager

Usuário  
oneadmin

Senha  
\*\*\*\*\*

End Point  
http://192.168.2.107:2633/  
RPC2

Salvar informações

LOGAR

Fonte: Próprio autor

### 5.3.2 Tela Listar Máquinas Virtuais

A Figura 10 apresenta a segunda tela do fluxo da aplicação, é onde são listadas as máquinas virtuais criadas anteriormente pelo usuário. Nesta tela, encontra-se um botão no canto inferior direito e quando o usuário clicar sobre ele serão mostradas duas opções, que são: instanciar máquina virtual e criar novo usuário. Cada uma das quais redirecionará o usuário para uma tela diferente. E se o usuário clicar sobre uma das máquinas virtuais, ele será redirecionado para uma tela onde serão mostradas algumas informações sobre esta máquina virtual.

Figura 10 - Tela Listar Máquinas Virtuais



Fonte: Próprio autor

### 5.3.3 Tela Instanciar Máquina Virtual

Ao clicar na opção “Instanciar Máquina Virtual”, o usuário será redirecionado para esta tela que é ilustrada pela Figura 11. Nesta tela, o usuário contará com dois campos, no primeiro ele deverá digitar um nome para a MV e no segundo, ele deverá escolher um dos *templates* existentes e assim poderá criar uma nova instância de uma máquina virtual com as configurações do *template* escolhido.

Figura 11 - Tela Instanciar Máquinas Virtuais



Lista MVs

Criar MV

Nome da MV

Escolha um Template  
ttylinux ▾

CRIAR

Fonte: Próprio autor

#### 5.3.4 Tela Criar Novo Usuário

Quando o usuário clicar na opção “Criar Novo Usuário”, ele será redirecionado para esta tela que é ilustrada pela Figura 12. Nesta tela, ele terá três campos para preencher, o primeiro será o nome de usuário, o segundo será a senha e por último um campo onde ele poderá confirmar sua senha. Depois de preencher todos os campos e clicar no botão “criar”, um novo usuário será criado com os dados digitados. Um ponto importante para salientar é que um usuário somente poderá criar um novo usuário se ele tiver permissão de administrador da nuvem.

Figura 12 - Tela Criar Novo Usuário

A imagem mostra uma interface de usuário para a criação de um novo usuário. No topo, há um cabeçalho com o texto "Criar Novo Usuário". Abaixo, há três campos de entrada de texto: "Usuario", "Senha" e "Confirme a Senha". Cada campo é precedido por uma linha decorativa horizontal. No final do formulário, há um botão cinza com o texto "CRIAR".

Fonte: Próprio autor

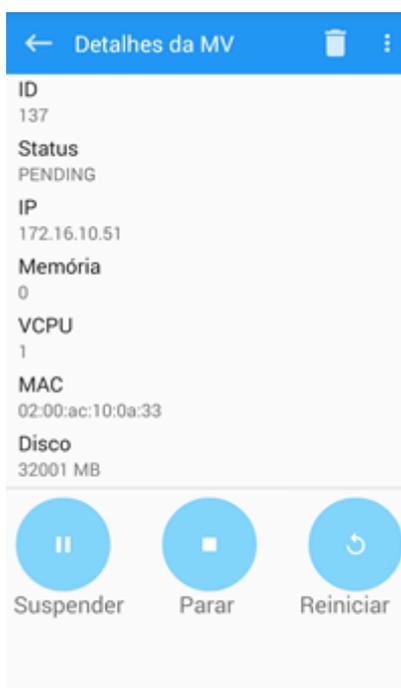
### 5.3.5 Tela de Detalhes da Máquina Virtual

Quando o usuário clicar sobre uma das máquinas virtuais ilustradas na Figura 10, ele será redirecionado para esta tela que está sendo representada pela Figura 13. A seguir, estão listadas as informações que podem ser vistas nesta tela: ID, Status, IP, Memória, VCPU (CPUs Virtuais), Endereço MAC, Tamanho do disco.

Esta tela dispõe de três botões na parte inferior, e um na parte superior com as opções que estão logo a abaixo:

- Parar Máquina Virtual
- Reiniciar Máquina Virtual
- Resumir Máquina Virtual (O botão de resumir uma máquina virtual estará presente sempre que o usuário parar ou suspender uma máquina virtual)
- Suspende Máquina Virtual.
- Remover Máquina Virtual

Figura 13 – Tela de Detalhes da Máquina Virtual



Fonte: Próprio autor

### 5.3.6 Tela de Configurações de Monitoramento

A Figura 14 apresenta a tela de configuração de monitoramento. Esta tela possui dois campos: o primeiro campo, o usuário deverá marcar caso queira receber notificações sempre que novas máquinas virtuais forem criadas ou removidas. O segundo campo é responsável por definir o intervalo de tempo em que a aplicação deverá sincronizar com o servidor do OpenNebula a fim de receber estas informações. Por padrão, este tempo é de 10 minutos.

Figura 14 – Tela de Configurações de Monitoramento



Fonte: Próprio autor

## 6 TESTE DE USABILIDADE

Nesta etapa foi utilizado o teste de usabilidade para avaliar a experiência do usuário para com a aplicação que foi desenvolvida neste trabalho. Nesta seção, é detalhado o passo a passo do teste de usabilidade aonde o último passo termina por concluir se o objetivo de desenvolver uma aplicação com boa usabilidade foi ou não atingido.

### 6.1 Preparação

Neste primeiro passo, foi definido o perfil dos participantes dos testes, que são professores, servidores ou alunos da Universidade Federal do Ceará - Campus Quixadá que já possuem alguma experiência na utilização do OpenNebula ou de qualquer outro gerenciador de nuvem. No total cinco participantes foram selecionados. O número pequeno de participantes é consequência do fato da aplicação OpenNebula Mobile Manager ser de uso muito específico, e por envolver a tecnologia de computação na nuvem. No entanto, o teste conseguiu atingir todos os perfis de usuários-alvo presentes no campus. Na Tabela 1 pode ser visto o perfil de cada um dos participantes do teste, seguido de uma breve descrição de cada

um. Para garantir a privacidade dos participantes, eles foram nomeados de: P1, P2, P3, P4 e P5.

O *smartphone* utilizado para a realização dos testes foi um Redmi 2 Pro da marca Xiaomi, com o sistema operacional Android Kit Kat na versão 4.4.4. Antes da realização dos testes, cada participante assinou um termo de consentimento que pode ser visto no Apêndice A. No Apêndice C, está apresentado o cenário de teste que foi utilizado pelos participantes na realização das tarefas. Os participantes também tiveram liberdade para explorar a aplicação e dar sugestões de melhorias.

Tabela 1 - Perfil dos Participantes do teste de usabilidade

Participante	Descrição
P1	O participante P1 era professor do curso de Redes de Computadores da UFC – Campus Quixadá, com experiência no uso do OpenNebula.
P2	O participante P2 era professor do curso de Redes de Computadores da UFC – Campus Quixadá, com experiência no uso do OpenNebula.
P3	O participante P3 era aluno do curso de Redes de Computadores da UFC – Campus Quixadá que fazia o uso do OpenNebula no seu trabalho de conclusão de curso
P4	O participante P4 era aluno do curso de Redes de Computadores da UFC – Campus Quixadá que já tinha feito o uso de outros sistemas de gerenciamento de nuvem diferentes do OpenNebula
P5	O Participante P5 era servidor técnico-administrativo da UFC – Campus Quixadá que fazia o uso do OpenNebula constantemente

Fonte: Próprio autor

## 6.2 Coleta de Dados

Os testes foram realizados na Universidade Federal do Ceará (UFC) – Campus Quixadá entre os dias 28/06/2016 e 01/07/2016. No primeiro dia de testes foram obtidas as credenciais da nuvem da UFC Campus Quixadá para que os testes fossem realizados em um ambiente real. No entanto, as credencias concedidas não tinham permissão de administrador da nuvem, por isso, a funcionalidade de criar um novo usuário só pôde ser testada uma vez, que foi na presença do administrador da nuvem do campus.

Com as credencias adquiridas, foi realizado o teste piloto, aonde foram encontrados alguns problemas na aplicação que até então não tinham sido notados, já que até aquele momento a aplicação só tinha sido testada em um ambiente de testes e nunca em um ambiente real. Com os erros devidamente corrigidos, foram iniciados os outros dias de testes.

A coleta de dados foi realizada da seguinte forma: primeiro, cada participante realizou as tarefas propostas no cenário apresentado anteriormente. Em seguida, foi aplicado um questionário ao final do teste, em que os participantes puderam avaliar alguns aspectos do sistema e sugerir modificações. O questionário pode ser visto no Apêndice C. É importante salientar que todos os participantes conseguiram realizar as tarefas com sucesso.

### **6.3 Interpretação dos Resultados**

A seguir estão listados os critérios que foram avaliados nos testes:

- Facilidade de utilização
- Organização das informações
- Layout das telas
- Nomenclatura utilizada nas telas
- Assimilação das informações
- Nível de satisfação ao utilizar a aplicação

Cada critério foi avaliado com uma nota que varia de um a cinco. O valor um corresponde a pior avaliação possível e conseqüentemente o valor cinco corresponde a melhor avaliação possível. Ao final do teste de usabilidade a pontuação que cada critério recebeu foi contabilizada, e com base nestas informações foram gerados os gráficos que estão apresentados nas Figuras 15,16 e 17.

### **6.4 Relato dos Resultados**

Durante o relato dos resultados iremos seguir a ordem dos critérios presentes no formulário de avaliação utilizado. O primeiro critério avaliado é a facilidade de utilização, conforme podemos ver na Figura 15, dois participantes do teste consideraram a facilidade de utilização do aplicativo normal, outros dois consideraram fácil de se usar, e um deles considerou muito fácil.

Figura 15 – Gráfico relacionado à facilidade de utilização



Fonte: Próprio autor

A Figura 16 apresenta o gráfico referente à organização das informações, nela podemos ver que três dos participantes considerou boa a organização das informações, e os outros dois consideraram normal.

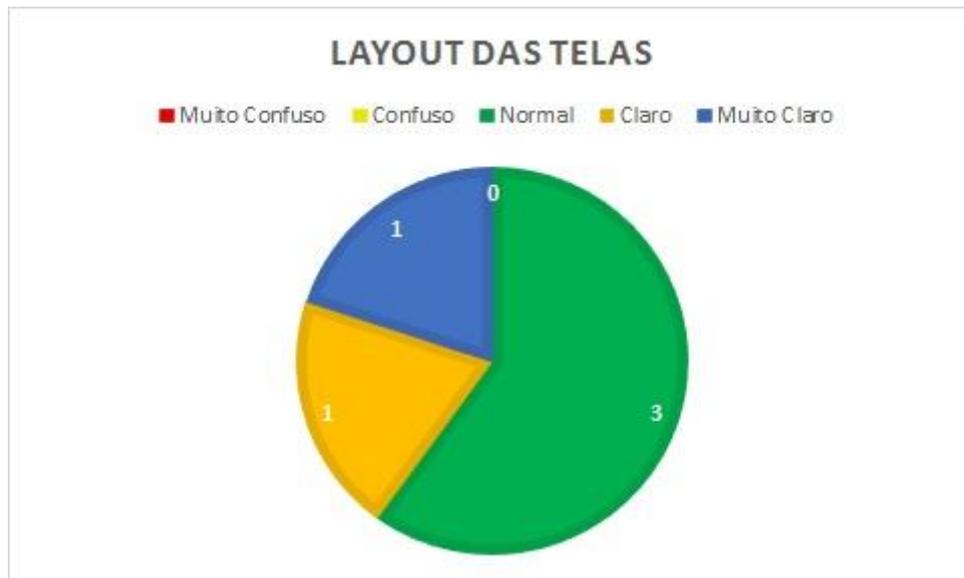
Figura 16 - Gráfico relacionado à organização das informações



Fonte: Próprio autor

A Figura 17 apresenta o resultado referente ao layout das telas, em relação a esse critério, três dos participantes consideraram normal, e os outros dois participantes ficaram divididos entre claro e muito claro.

Figura 17 – Gráfico relacionado ao layout das telas



Fonte: Próprio autor

A Figura 18 apresenta o gráfico referente à nomenclatura utilizada nas telas. Em relação a esse critério, quatro dos participantes consideraram que a nomenclatura estava clara, e um considerou muito clara.

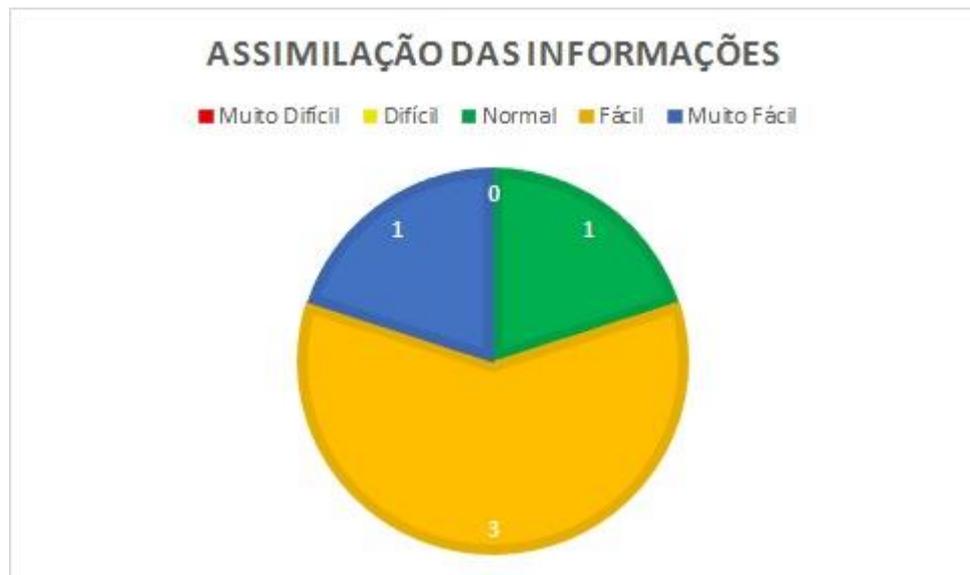
Figura 18 – Gráfico relacionado à nomenclatura utilizada nas telas



Fonte: Próprio autor

A Figura 19 apresenta o gráfico referente à assimilação das informações, podemos ver que três dos participantes consideraram que as informações são de fácil assimilação, e os outros dois ficaram divididos entre normal e muito fácil.

Figura 19 - Gráfico relacionado à assimilação das informações



Fonte: Próprio autor

Na Figura 20, temos o gráfico referente ao nível de satisfação que o participante do teste sentiu ao utilizar a aplicação, podemos ver que três dos participantes consideraram normal e os outros dois sentiram um alto nível de satisfação ao utilizar a aplicação.

Figura 20 - Gráfico relacionado ao nível de satisfação ao utilizar a aplicação



Fonte: Próprio autor

Como foi apresentado nas Figuras 15,16,17,18,19 e 20, a aplicação OpenNebula Mobile Manager obteve um bom desempenho em cada um dos critérios avaliados no teste de usabilidade.

A Figura 21 apresenta uma visão geral sobre todos os critérios analisados no teste, facilitando a visualização dos critérios que tiveram uma maior aceitação por parte dos participantes do teste. Para chegar neste resultado, foi calculada a média da pontuação que cada um dos critérios obteve. Pode ser visto que o critério cinco obteve um melhor desempenho sobre os outros critérios avaliados

Figura 21 - Visão geral sobre o resultado do teste de usabilidade



Fonte: Próprio autor

Como foi explicado na subseção 6.1, o questionário do teste de usabilidade continha um campo onde o participante era livre para dar sugestões de melhorias no sistema, logo abaixo estão algumas sugestões que foram aceitas e implementadas na aplicação:

- Implementar a opção de deslogar da aplicação;
- Consertar a opção de fazer login que estava com erro;
- Remover botão de voltar da tela de Listar Máquinas Virtuais, pois a aplicação já contava com a opção de deslogar da aplicação;
- Sinalizar os status das máquinas virtuais através de cores, para torna-los mais intuitivos;
- Não deixar enviar requisições para uma máquina virtual quando seu estado não tiver transição.

## 7 CONCLUSÃO E TRABALHOS FUTUROS

A ideia deste trabalho surgiu da ausência de aplicativos para dispositivos Android que permitissem o gerenciamento de nuvens OpenNebula. A aplicação OpenNebula Mobile Manager surge como uma alternativa à ferramenta web OpenNebula Sunstone. A aplicação desenvolvida neste trabalho fornece as principais funcionalidades presentes na ferramenta

web OpenNebula Sunstone, e ainda possui uma funcionalidade adicional que notifica o usuário sempre que novas máquinas virtuais são criadas ou removidas.

A aplicação OpenNebula Mobile Manager passou pelo teste de usabilidade como foi explicado na seção 6, e através da análise gráfica, ficou claro que a aplicação obteve um bom desempenho no teste de usabilidade e se mostrou uma boa alternativa ao uso do OpenNebula Sunstone. Entretanto, durante o teste de usabilidade alguns usuários sugeriram melhorias para a aplicação. Dentre essas sugestões, algumas não foram aceitas por fugirem do escopo deste trabalho. No entanto, as seguintes sugestões serão desenvolvidas futuramente como continuidade deste trabalho:

- Atualizar o estado das máquinas virtuais automaticamente, sem a necessidade de apertar o botão de atualizar;
- Criar a opção de customizar os *templates*;
- Permitir a criação de redes virtuais para as máquinas virtuais;
- Verificar quantos hosts estão ativos na nuvem;

Como trabalho futuro, também se espera que a aplicação desenvolvida neste trabalho seja capaz de realizar o gerenciamento de outras nuvens além da nuvem do OpenNebula. Como contribuição deste trabalho, o código da aplicação está disponível no Github.

## REFERÊNCIAS

- BARBOSA, S. D. J.; SILVA, B. S. **Interação Humano-Computador**. Rio de Janeiro: Elsevier, 2010. 384p.
- BORGES, H. P.; SOUZA, J. N.; SCHULZE, B.; MURY, A. R. **Computação em nuvem**. Brasil, 2011. Disponível em: <<http://livroaberto.ibict.br/handle/1/861>>. Acesso em: 13 nov. 2015.
- CHOINACKI, H. “**Virtualização de Servidores**”, Monografia, Departamento Acadêmico De Eletrônica - Universidade Tecnológica Federal Do Paraná – Curitiba, 2012.
- COUTINHO, E. F.; SOUSA, F. R. C.; GOMES, D. G.; SOUZA, J. N. (2013). **Elasticidade em computação na nuvem: Uma abordagem sistemática**. In XXXI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC 2013) - Minicursos.
- IDC. **Smartphone OS Market Share, 2015 Q2**. 2015. Disponível em:<<http://www.idc.com/prodserv/smartphone-os-market-share.jsp>>. Acesso em: 02 dez. 2015.
- NÚÑEZ, J. G.; SÁNCHEZ, J.H.; ARANDA, D.M. **An implementation of the Sun Cloud API for the OpenNebula Toolkit**. 2010. Disponível em: <<http://eprints.ucm.es/11284/>>. Acesso em: 03 dez. 2015.
- OLIVEIRA, A. L.; CHAPETTA, M. G. A.; ANDRADE, A. M.; PIMENA, L. B. S.; THIAGO, C. **Computação em Nuvens**. 2013. Disponível em: <<http://www.revista.universo.edu.br/index.php?journal=1reta2&page=article&op=viewArticle&path%5B%5D=1188>>. Acesso em: 13 nov. 2015.
- RAMOS, M. **Teste de Usabilidade**. 2013. Disponível em: <<http://www.marceloramos.com.br/publicacao/68/teste-de-usabilidade>>. Acesso em: 18 jan. 2016.
- SANTOS, F. A. G. **Fundamentos da computação em nuvem com abordagem das plataformas Eucalyptus e OpenNebula**. 44 f. Trabalho de Conclusão de Curso (Graduação) – Curso Superior de Tecnologia em Sistemas para Internet. Universidade Tecnológica Federal do Paraná. Campo Mourão, 2013.
- SOTO, J. A. “**Opennebula: Implantação de uma nuvem privada e orquestração das máquinas virtuais no paradigma da computação em nuvem**”, Monografia, Departamento de Engenharia de Teleinformática - Universidade Federal do Ceará - Fortaleza, 2011.
- X. Li and J. Qiu, **Cloud Computing for Data-Intensive Applications**. Reading, MA: Springer-Verlag, 2014.

## APÊNDICE A – TERMO DE CONSENTIMENTO

### Termo de Consentimento

Este termo de consentimento é referente ao trabalho de conclusão de curso do aluno Francisco Daniel Farias Graciano, com o título: Gerenciamento e Monitoramento de Máquinas Virtuais Através da Aplicação OpenNebula Mobile Manager. Os dados coletados durante os testes serão estritamente utilizados para análise e melhorias no sistema.

Posteriormente, os resultados dos testes serão divulgados no trabalho de conclusão de curso do aluno citado anteriormente, aonde o mesmo garante preservar o anonimato dos participantes.

- (  ) Aceito participar dos testes.
- (  ) Não aceito participar dos testes.

---

Assinatura do Participante

---

Assinatura do aplicador do teste

## APÊNDICE B – CENÁRIO DO TESTE

CENÁRIO: Você no papel de Aluno obteve seu acesso ao OpenNebula e agora poderá começar a realização dos testes.

- Tarefa 1: Você cria uma instância de uma máquina virtual.
- Tarefa 2: Você verifica as informações sobre a máquina virtual que você acabou de criar para ver se atende as suas necessidades.
- Tarefa 3: Você checou as informações e viu que a máquina virtual não atende as suas necessidades, você então deleta esta máquina virtual e cria uma instância escolhendo outro *template* que melhor lhe atenda.
- Tarefa 4: Chegou o final de semana e durante esse período você não vai utilizar os recursos da máquina virtual que você criou, você então para a máquina virtual. Você vai liga-la de novo somente quando precisar.

## APÊNDICE C – INSTRUMENTO DE COLETA DE DADOS

### QUESTIONÁRIO

O questionário a seguir funcionará como instrumento de coleta de dados para o trabalho de conclusão de curso do aluno Francisco Daniel Farias Graciano, estudante da universidade Federal do Ceará – Campus Quixadá. O título do trabalho é: Gerenciamento E Monitoramento de Máquinas Virtuais Através da Aplicação OpenNebula Mobile Manager.

1. Marcar o número correspondente ao grau que você mais concorda.

1.	Facilidade de utilização	Muito difícil 1	difícil 2	Normal 3	fácil 4	muito fácil 5
2.	Organização das informações	Muito ruim 1	ruim 2	Normal 3	Bom 4	muito bom 5
3.	Layout das telas	Muito confuso 1	confuso 2	Normal 3	Claro 4	muito claro 5
4.	Nomenclatura utilizada nas telas (nome de comandos, títulos, campos, etc.)	Muito confuso 1	confuso 2	Normal 3	Claro 4	muito claro 5
5.	Assimilação das informações	Muito difícil 1	difícil 2	Normal 3	fácil 4	muito fácil 5
6.	Nível de satisfação ao utilizar a aplicação	Muito baixo 1	Baixo 2	Normal 3	alto 4	muito alto 5

2. Aponte situações em que você sentiu dificuldades ao utilizar a aplicação (Opcional).

---



---



---



---

3. O espaço abaixo é reservado para que você exponha sua opinião e sugira melhorias na aplicação (Opcional).

---



---



---



---



---