



<http://ensaios.usf.edu.br/>

**REAPROVEITAMENTO DE DISPOSITIVOS COMPUTACIONAIS UTILIZANDO  
COMPUTAÇÃO EM NUVEM COM VISTAS À SUSTENTABILIDADE NA ÁREA DE  
TECNOLOGIA DA INFORMAÇÃO**  
*REUSE OF COMPUTATIONAL DEVICES USING CLOUD COMPUTING LOOKING FOR  
SUSTAINABILITY IN THE INFORMATION TECHNOLOGY AREA*

ZORZI, Lucas<sup>1</sup> BARDI, Marcelo Augusto Gonçalves<sup>1</sup>

<sup>1</sup> Grupo de Pesquisas em Meio Ambiente e Sustentabilidade (GPMAS), Universidade São Francisco

[lucas.zorzi@hotmail.com](mailto:lucas.zorzi@hotmail.com)

**RESUMO.** Com um crescimento na evolução tecnológica dos mais diversos dispositivos eletrônicos e sua difusão, além do baixo tempo de vida associado a esses equipamentos, a geração de resíduo eletrônico é uma preocupação mundial devido ao grande impacto ambiental causado pelo seu descarte inadequado. Comumente, o descarte desses resíduos pode ser feito corretamente por meios de reciclagem ou reuso, reaproveitando elementos que possuam algum valor associado. Entretanto, a falta de recicladoras, pontos de coletas e incentivos, faz com que o descarte correto desses resíduos não seja tão efetivo quanto o ideal. Dessa forma, esse projeto de pesquisa vem estudar a proposta de reinserção de equipamentos classificados como lixo eletrônico em aplicações cotidianas. Pretende-se, como alternativa, criação de um repositório *online* de lixo eletrônico para que a população local possa doar equipamentos e dá-los um destino correto. Com isso, pretendemos criar uma infraestrutura com os equipamentos ainda funcionais e por meio da aplicação da computação em nuvem usando o *software OpenStack*, dar vida nova a esses equipamentos. Com essas propostas, é possível agregar valor ao produto tecnológico pós-consumo recuperado e reaproveitado, gerando lucro a partir de algo que já se encontrava obsoleto.

**Palavras-chave:** lixo eletrônico, computação em nuvem, *OpenStack*.

**ABSTRACT.** With growth in the technological evolution of most diverse electronic devices and their diffusion, besides the low life time of these equipment, the generation of electronic waste is a global concern due the huge impact on the environment caused by its inadequate disposal. Usually, the disposal of these wastes can be done correctly by means of recycling or reuse, reusing elements that have some associated value. However, the lack of recyclers, points of collection and incentives, makes that the correct disposal of these wastes is not as effective as the ideal. Thus, this research project is studying the proposal of reinsertion of equipment classified as electronic waste in everyday applications. It is intended, as an alternative, the creation of an online e-waste repository so that the local population can donate equipment and give them a correct destination. With this, we intend to create an infrastructure with the equipment still functional and through the application of cloud computing using the software *OpenStack*, give new life to such equipment. With these proposals, it is possible to add value to the post-consumer technological product recovered and reused, generating profit from something that was already obsolete.

**Keywords:** e-waste, cloud computing, *OpenStack*.

## INTRODUÇÃO

A tecnologia é muito importante atualmente e a evolução nesta área progrediu rapidamente, sofrendo várias mudanças ao longo de sua existência com o surgimento de novas tecnologias e o aprimoramento das existentes, o que dá a característica de uma das áreas mais inovadoras.

A disponibilidade dessas novas tecnologias ofertadas tanto ao público doméstico quanto ao empresarial, mostrou-se um utensílio extremamente eficiente e despojado, tornando-se indispensável para diversas tarefas complexas pois as torna muito mais simples e rápidas, no qual representa ganhos imediatos de tempo, dinheiro, obtenção de resultados, além de aumentar a produtividade e melhorar a qualidade de produtos desenvolvidos a partir delas. Apesar dos problemas que os equipamentos eletrônicos podem causar por conta de seu descarte inadequado, graças ao progresso da tecnologia podemos suavizar os problemas causados por ela usando-a como aliada.

A computação em nuvem é uma dessas tecnologias que podem ser usadas para suavizar os problemas de resíduos eletrônicos. Ela está revolucionando a maneira como os dados são armazenados, como os *softwares* e serviços são entregues, como o consumidor acessa todos esses serviços. Assim, torna-se o uso dos recursos computacionais mais inteligente. Todos esses aspectos estão contribuindo para a redução de impactos no meio ambiente. A computação em nuvem não é um conceito novo, baseia-se em várias tecnologias e conceitos já existentes como virtualização, *Grid Computing*, *Utility computing*, computação autônoma, *Web 2.0*, *Service Oriented Architecture* e o modelo de *Software como Serviço* (TAURION, 2009).

Nuvens são grandes conjuntos de recursos virtualizados de fácil acesso e uso, tais como *hardware*, plataformas de desenvolvimento ou serviços. Esses recursos podem ser ajustados dinamicamente de forma a otimizar seu uso. Normalmente, esses conjuntos de recursos são explorados utilizando o modelo de pague pelo uso no qual paga-se de acordo com a quantidade de recursos utilizada (VAQUERO *et al.*, 2012). O usuário pode usar recursos computacionais, *softwares* e serviços em qualquer lugar por meio de várias plataformas, necessitando apenas de uma conexão de rede local ou *internet* dependendo de como o recurso é provido (VERAS, 2016).

Portanto, a computação em nuvem é um conjunto de recursos, como *hardware*, *software*, plataformas de desenvolvimento e serviços disponíveis virtualmente para que possam se ajustar dinamicamente de acordo com uma carga de trabalho variável, permitindo otimização em sua utilização. Em outras palavras, são mantidos ativos apenas os recursos necessários para atender a demanda. Isso causa uma alta economia para a empresa, reduzindo custos com energia elétrica e manutenção de equipamentos.

A nuvem pode ser acessada por diversos tipos de dispositivos como computador, *tablet*, *smartphone* e *smart TVs*. Aplicando-se a proposta da computação em nuvem, a instalação das aplicações não é mais necessária, há a possibilidade de acessar arquivos e usar serviços através da rede. Os dados não estão disponíveis apenas em um único dispositivo, mas em um servidor. Isso significa que o cliente precisa apenas de um dispositivo de acesso básico com uma conexão de rede ativa para aproveitar todo o conteúdo, salvar o trabalho e acessar novamente de qualquer lugar (TAURION, 2009).



**Figura 1** - Diagrama da computação em nuvem como serviço

O modelo de computação em nuvem foi desenvolvido para fornecer vantagens que tornam uma ótima opção manter uma infraestrutura. Para isso, ela fornece benefícios como: o custo de aquisição de toda a infraestrutura para atender às necessidades é reduzido, pode ser construído por demanda e acarreta em custos mais baixos com recursos heterogêneos. A flexibilidade é outro ponto, o fornecedor pode adicionar ou substituir os recursos sem interromper os serviços, oferecendo disponibilidade, escalabilidade e segurança aos usuários. Além disso, proporciona facilidade aos usuários acessar os serviços, eles não precisam saber aspectos ou como a nuvem funciona, criando um ambiente fácil para atrair os usuários finais (SOUZA *et al.*, 2010).

De acordo com MELL e GRANCE (2011), existem cinco características essenciais da computação em nuvem que englobam tanto infraestrutura quanto serviços:

- Autosserviço sob demanda: o usuário pode adquirir recursos computacionais conforme necessidade automaticamente sem requerer interação com humanos responsáveis pelo provedor de serviços, fornecendo os serviços de forma transparente para o usuário.

- Amplo acesso de rede: os recursos estão disponíveis sobre uma rede e acessível através de mecanismos padrão que permitem o uso por meio de diferentes plataformas, como telefones celulares, *tablets*, *laptops* e estações de trabalho, oferecendo flexibilidade de acesso aos usuários.

- *Pooling* de recursos: os recursos de computação do provedor são organizados em conjuntos para atender múltiplos consumidores usando um modelo *multi-tenant*, em que uma aplicação serve vários usuários. Assim, diferentes recursos físicos e virtuais são redistribuídos dinamicamente de acordo com a demanda do consumidor. Os usuários não precisam saber a localização exata dos recursos, podendo especificar localização em um nível mais alto de abstração, como estado, país ou *data center*.

- Elasticidade rápida: os recursos podem ser provisionados e acrescentados ou decrementados rapidamente de acordo com a demanda, fazendo aparecer ao consumidor que os recursos são ilimitados e podem ser empregados qualquer quantidade a qualquer momento.

- Serviço de medição: os sistemas de computação em nuvem são capazes de medirem o uso dos recursos e, assim, controlá-los e otimizá-los, proporcionando transparência tanto para o consumidor e provedor (MELL e GRANCE, 2011).

Existem alguns modelos para entregar a computação em nuvem dos fornecedores para consumidores e são chamados de modelos de serviços. A seguir, são mostrados os três principais modelos:

- *Software* como serviço (SaaS): é um modelo que fornece ao consumir a possibilidade de usar os aplicativos do provedor que estão sendo executados em uma infraestrutura de nuvem. As aplicações podem ser acessadas a partir de uma interface *thin client*, como um navegador *web* ou a interface de um programa. Os consumidores que utilizam este modelo eliminam os aplicativos instalados em um dispositivo local, manutenção do aplicativo e armazenamento dos dados, eles podem acessar em qualquer lugar e momento. Porém, não é possível gerenciar ou controlar a infraestrutura de nuvem, rede, servidores, sistemas operacionais, armazenamento e o aplicativo, exceto algumas definições específicas de configuração das aplicações.

- Plataforma como Serviço (PaaS): neste modelo, é fornecido o *software*, o *hardware* e o recursos para que o consumidor possa implantar aplicativos dentro infraestrutura da nuvem. Ou seja, por meio do modelo de PaaS, não é necessário comprar e manter *hardwares*, instalar e gerenciar sistemas operacionais e bancos de dados. Além disso, o os recursos podem aumentar ou diminuir de acordo com a demanda e é pago apenas o consumido. No entanto, o consumidor não gerencia ou controla a infraestrutura de nuvem, rede, servidores, sistemas operacionais e armazenamento, apenas as aplicações implantadas.

- Infraestrutura como serviço (IaaS): é um modelo que o provedor disponibiliza todos os recursos computacionais de *hardware* para o consumidor, como processamento, armazenamento, redes e outros, onde o consumidor precisa instalar e gerenciar os sistemas, pode implementar e executar qualquer *software* e tem controle sobre todos estes recursos, até mesmo alguns componentes de rede como *firewalls*, mas não a infraestrutura física da nuvem (JAMSA, 2013; MELL e GRANCE, 2011).

Sobre o acesso e a disponibilidade dos dados em uma nuvem, existem quatro modelos que podem ser implementados em uma infraestrutura de nuvem. Dependendo da necessidade do usuário, é possível gerenciar para que a nuvem seja mais restrita, compartilhada com outros consumidores, acesso restrito e muito mais. Esses modelos são:

- nuvem privada onde a infraestrutura é provida para uso exclusivo de um único consumidor, tornando um ambiente mais seguro, pois aplica controle de acesso limitando a disponibilidade dos dados apenas ao proprietário;
- na nuvem comunitária, a nuvem é compartilhada por uma comunidade específica de consumidores que tem os mesmos interesses;
- a nuvem pública é uma infraestrutura de nuvem aberta ao público em geral, eles podem usar todos os recursos oferecidos pelo provedor;
- a nuvem híbrida é uma composição de duas ou mais infraestruturas de nuvem distintas (privadas, comunitárias ou públicas) que compõem uma nuvem única que contém as definições desses modelos, mesclando os interesses (MELL e GRANCE, 2011).

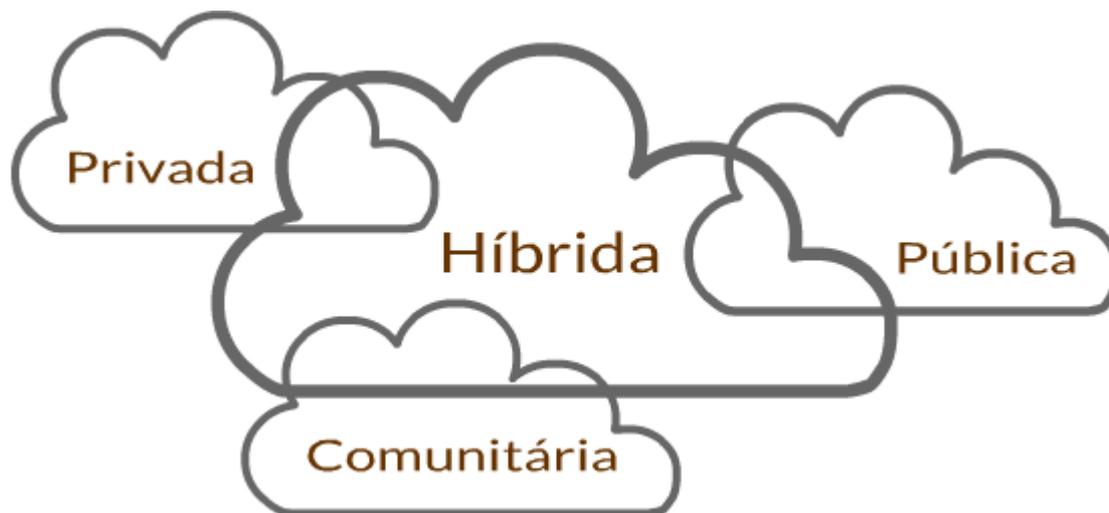


Figura 2 - Modelos de implementação de nuvem

Para saber qual é o melhor modelo de implantação da nuvem para um determinado caso, existem vários fatores a serem considerados. É preciso examinar os recursos financeiros disponíveis, o modo preferencial de gastos, de onde os usuários acessarão a nuvem e uma previsão da demanda. Além disso, é necessário levar em consideração os objetivos da organização e como encaixá-los em um modelo de nuvem, escolhendo a melhor opção (BARRET *et al.*, 2017).

O uso da computação em nuvem deve ser feito com muito cuidado, a adoção desse modelo reflete muitas vezes na alocação de dados particulares em servidores distantes e com dados trafegados via *internet*: um meio democrático, porém potencialmente inseguro. É possível manter o armazenamento em um local e o processamento em outro, o que a torna mais suscetível a invasões que englobam ataques em nível de máquina virtual, como vulnerabilidades no próprio provedor, práticas de *phishing* que coletam informações confidenciais, ataques na infraestrutura de comunicação, e diversos outros. Para saber se o uso da nuvem é uma boa opção ou não, deve-se avaliar os riscos e benefícios no tratamento dos dados e processos olhando para eles e definindo sua importância, de forma a responder questões sobre vazamento de conteúdo, acesso indivíduo, falhas no sistema, indisponibilidades, entre outras. Assim, consegue-se definir se esse modelo consegue entregar a confidencialidade, integridade e disponibilidade desejada (ZANUTTO, 2017).

A capacidade de interoperabilidade é a base de uma nuvem, fazendo uso de diversos protocolos e mecanismos para permitir aos usuários o acesso em locais distintos. Eles devem ser flexíveis de forma a aceitarem uma numerosa variedade de sistemas e implementadas em camadas, como as que compõem a *Internet* (TAURION, 2009).

### *OpenStack*

Atualmente existem diversas soluções disponíveis para a construção de uma nuvem, nos diversos modelos de implantação. De todas as opções, a que mais se destaca é o projeto OpenStack.

O *OpenStack* é um *software* de gerenciamento de nuvem que permite a criação de uma nuvem IaaS e colocá-la em execução em servidores próprios. É composto por vários projetos inter-relacionados que oferecem diversos componentes para uma solução de infraestrutura de nuvem. Esses serviços controlam grandes conjuntos de recursos de computação, armazenamento e rede em um servidor. Cada serviço fornece uma *API* aberta que pode ser gerenciada através de um painel de controle disponível em uma interface *web*, linha de

comando ou um *kit* de desenvolvimento de *software*, dando ferramentas administrativas aos usuários para gerenciarem os serviços que compõem a nuvem *OpenStack*. O *OpenStack* é um projeto de código aberto disponível à comunidade e qualquer pessoa com interesse em participar do desenvolvimento, produzindo uma plataforma de computação em nuvem tanto para nuvens públicas quanto privadas com foco na entrega de uma solução de fácil uso e confiável para todos os tipos de organizações que desejam ter uma nuvem (OPENSTACK, 2017a).

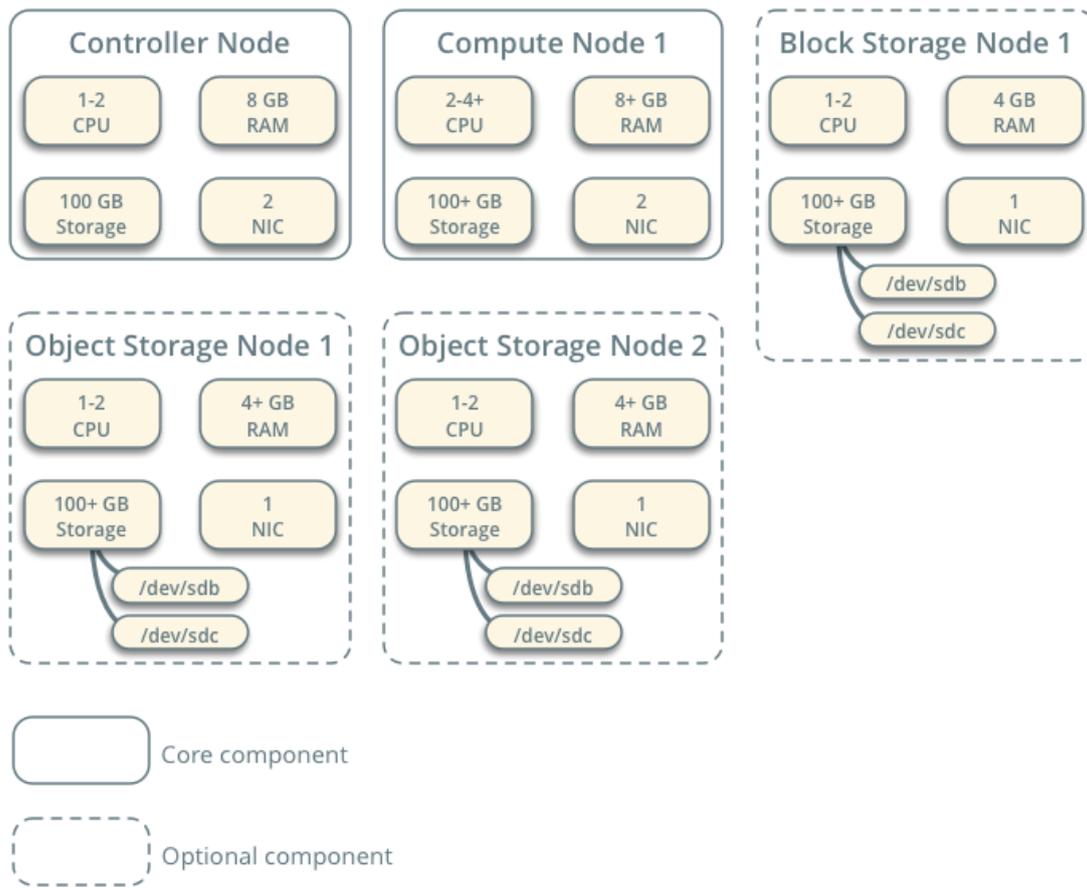
Os componentes do *OpenStack* estão divididos em três grupos: computação, rede e controle. A computação é o *hypervisor* de virtualização, responsável por manipular as máquinas virtuais da nuvem; o grupo de rede controla toda a rede e o grupo de serviços de controle executa os serviços *API*, interface *web*, banco de dados e o barramento de mensagens (RADEZ, 2015). Atualmente, existem nove grupos de serviços, totalizando 46 projetos principais do *OpenStack*, como os doze serviços principais que o *OpenStack* precisa executar (*Keystone*, *Nova*, *Horizon*, *Neutron*, *Ceilometer*, *Glance*, *Swift*, *Cinder*, *AQMP*, *Heat*, *Trove* e *Sahara*), e ainda há muito mais serviços opcionais, chamados serviços *Big Tent* que incrementam ainda mais as funcionalidades do *OpenStack*, além de projetos em desenvolvimento esperando por maturidade para entrar nesta lista (OPENSTACK, 2017a; SHRIVASTWA e SARAT, 2015).

*Nova* é o serviço principal do *OpenStack*, ele é quem gerencia toda a infraestrutura computacional da nuvem, controlando as instâncias, recursos computacionais, rede, autorizações e escalabilidade. O *Keystone* é o serviço de identificação, gerenciando tudo o que é relacionado a autenticações e autorizações de nível superior. *Horizon* é um painel de controle que agrupa os serviços do *OpenStack* numa interface gráfica *web* simples para facilitar o uso da ferramenta. *Neutron* é um projeto que provê rede como serviço entre dispositivos, sendo escalável e sob demanda. *Glance* que permite aos usuários descobrir, registrar e recuperar imagens de máquinas virtuais. *Swift* é armazenador de objetos. *Cinder* faz o armazenamento persistente de blocos para que as instâncias possam acessá-los. *Heat* é quem faz a orquestração gerenciando todos os ciclos de vida da infraestrutura e aplicações. *Trove* provê funcionalidade para o funcionamento de motores de banco de dados.

Com um conjunto de diversos componentes, o *OpenStack* possui uma arquitetura modular e, juntos, eles fazem um incrível orquestrador, isto é, os serviços estão separados e podem ser executados de forma independente, mas trabalhando juntos em harmonia para compor o *OpenStack*, dependem uns dos outros para gerenciar todo o sistema e entregar um produto, como uma máquina virtual. (SHRIVASTWA e SARAT, 2015).

A arquitetura mínima para executar o *OpenStack* são dois nós: um nó controlador (1-2 CPU; 8 GB de memória RAM; 100 GB de armazenamento; 2 controladores de interface de rede) e um nó de computação (2-4+ CPU; 8+ GB de memória RAM; 100+ GB de armazenamento; 2 controladores de interface de rede). O nó controlador contém o serviço de identidade (*Keystone*); o Serviço de imagem (*Glance*); Gerenciamento de recursos computacionais (*Nova*), Gerenciamento de rede (*Neutron*), vários agentes de rede e o Painel de controle (*Horizon*). Já o nó de computação, executa a parte do *hypervisor* que opera as máquinas virtuais (instâncias). Ele também executa um agente de serviço de rede que conecta instâncias a redes virtuais e fornece serviços de *firewall* para eles.

## Hardware Requirements



**Figura 3** - Requerimentos de *hardware* do *OpenStack*  
 (Fonte: <https://docs.openstack.org/install-guide/overview.html>)

Opcionalmente, podem existir mais nós. Por exemplo, o nó de armazenamento em bloco (1-2 CPU, 4 GB de RAM, 100+ GB de armazenamento, 1 controlador de interface de rede) que contém os discos provisionados para instâncias. Ainda, há o nó de armazenamento de objetos que requer dois *hosts* (1-2 CPU, 4+ GB de RAM, 100+ GB de armazenamento, 2 controladores de interface de rede) e contém os discos usados para armazenar contas, contêineres e objetos, no entanto, é possível implantar mais nós de ambos conforme a necessidade de armazenamento do projeto (OPENSTACK, 2017b).

Existem alguns casos de aplicação do *OpenStack* que valem a pena ser conhecidos. No estudo de caso da Canarie, foi idealizado o DAIR, um ambiente virtual integrado para desenvolver e testar novas tecnologias da informação e comunicação (TIC) e outras tecnologias digitais. São combinadas uma infraestrutura digital, redes avançadas, computação em nuvem e armazenamento. Existem dois *data centers* para hospedá-lo e o *OpenStack* controla essas infraestruturas, integrando ambos no mesmo sistema (OPENSTACK, 2017c).

O grupo Volkswagen implantou uma nuvem privada *OpenStack* em meados de 2015 com o objetivo de melhorar os fluxos de trabalho, obter mais níveis de automatização e padronizar a infraestrutura de TI e aplicações, além da redução de custos. Foi construído um *data center* apenas para hospedar a nuvem *OpenStack* com 8 736 núcleos, 258 440 GB de memória RAM e mais de 2 500 TB de armazenamento (MÜLLER, 2017).

No caso da organização europeia para pesquisas nucleares (CERN), os pesquisadores utilizam uma enorme infraestrutura para realizar pesquisas sobre física de alta energia. O

ambiente, além de fazer uso dos serviços do *OpenStack*, a nuvem está integrada com alguns serviços externos para atender às necessidades (OPENSTACK, 2017c).

Há diversos trabalhos científicos que também fazem uso do *OpenStack*, um deles utiliza o *OpenStack* para gerenciar dispositivos *IoT* utilizando uma abordagem orientada para a nuvem dedicada às cidades inteligentes. Criando uma solução capaz de lidar com enormes quantidades de dados e dispositivos que são gerenciados de forma inteligente pela nuvem (MERLINO *et al.*, 2014).

Outro trabalho feito por MATTOS e SENGER (2015), apresentou uma ferramenta para instalação de um cluster com o *framework Hadoop* que trabalha com *BigData* dentro de uma nuvem *OpenStack*. A solução facilita a instalação de ferramentas para análise e armazenamento de uma quantidade gigante de dados, manipulando tudo e mantendo a flexibilidade que só a nuvem proporciona.

KURESHI *et al.* (2013) implantaram uma nuvem privada com *OpenStack* para melhorar a infraestrutura de pesquisa de alto desempenho computacional (HPC), levando um recurso flexível e escalável para pesquisa, ensino e avaliação que maximiza a eficiência de utilização de recursos da nuvem.

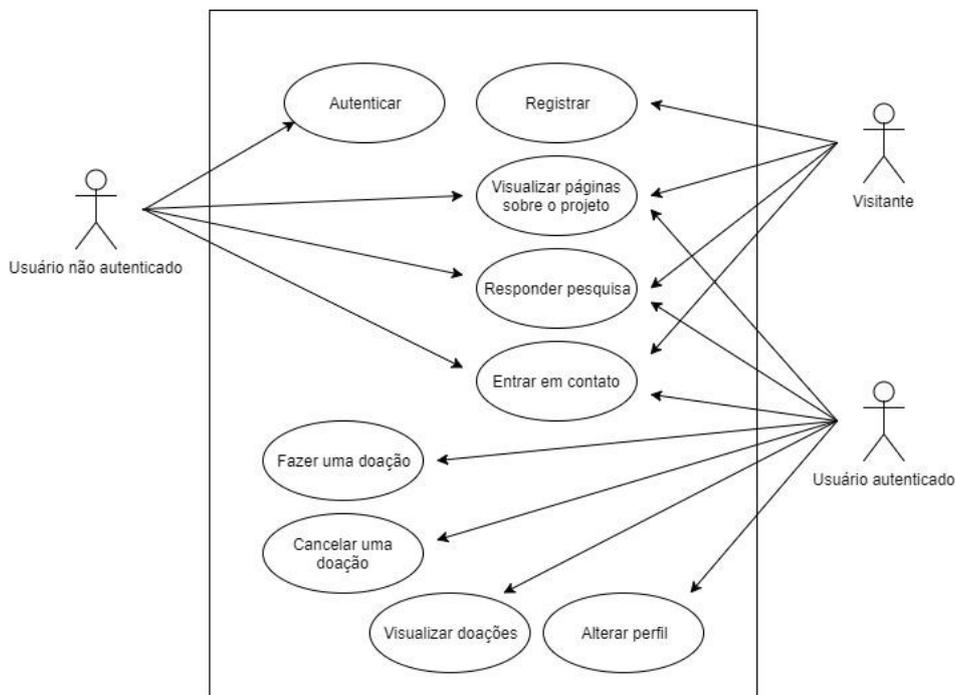
O *OpenStack* tem se mostrado uma das melhores e populares ferramentas para tornar real a construção, implantação e gerenciamento de uma nuvem em uma infraestrutura grande ou pequena. Muitas empresas decidiram usar este *software* devido ao seu poder, flexibilidade, projeto de código aberto, possui uma grande comunidade que o suporta, alta compatibilidade com serviços de terceiros e diversas outras qualidades que tendem a aparecer rapidamente.

Desta forma, este projeto de pesquisa propõe uma alternativa para acrescer a reutilização de resíduos eletrônicos e como gerenciá-los adequadamente usando o conceito de computação em nuvem por meio da aplicação do *software OpenStack* em uma infraestrutura composta por esses resíduos eletrônicos, podendo dar-lhes vida nova por meio da virtualização. Além de disponibilizar um sistema de repositório *online* para que a população da região de Itatiba – SP possa doar seus resíduos de eletrônicos, sendo recuperados e reinseridos ao uso em aplicações cotidianas por meio da virtualização, ou destinados a descarte em locais apropriados.

## **METODOLOGIA**

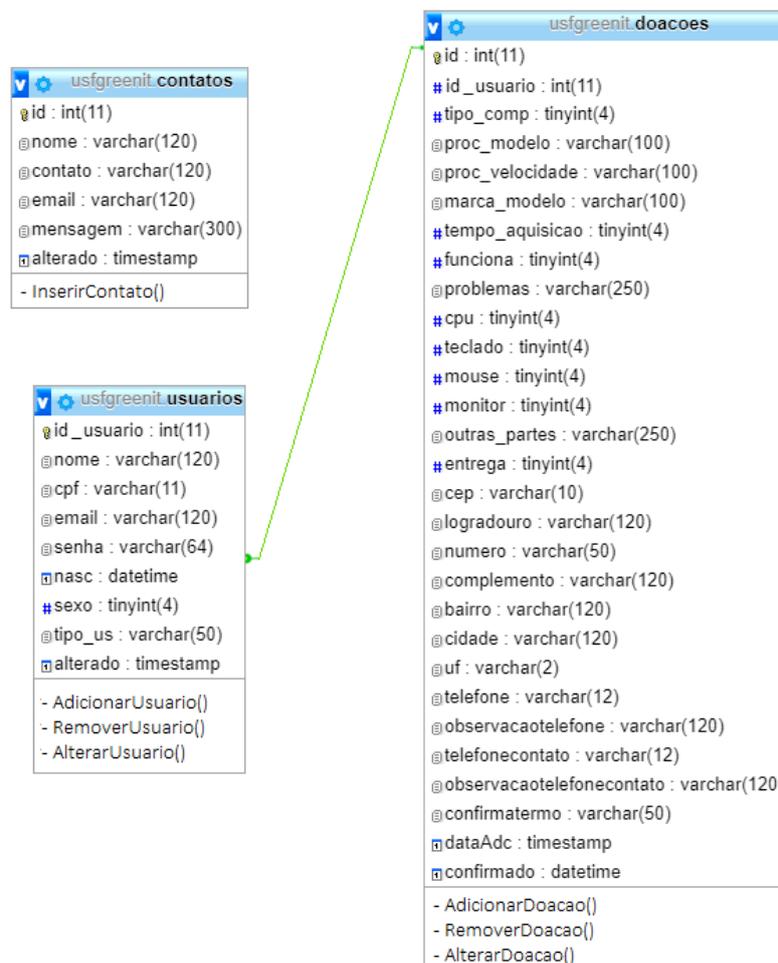
De modo a se avaliar a significância, pertinência e aceitabilidade do projeto pela sociedade, foi construído um sistema *online* para oferecer aos cidadãos uma alternativa para descarte de componentes eletrônicos funcionais por meio de doações através de nosso sistema e ainda disponibilizar dicas sobre reciclagem, reutilização e destinação correta de equipamentos eletrônicos. Este *site* está disponível sob a forma de um repositório *online*, onde a população local poderá preencher um formulário com dados sobre um equipamento ou componente e doá-los.

A Figura 4 mostra o caso de uso do *site* descrevendo suas principais funcionalidades do ponto de vista do usuário, em que um usuário não autenticado é o usuário que possui uma conta cadastrada no *site*, no entanto não está autenticado no momento, visitante é o usuário que não possui uma conta cadastrada e precisa registrar-se para ter acesso a outras partes do *site* referentes às doações e usuário autenticado é o usuário que possui uma conta cadastrada e está autenticado com ela no *site*, podendo acessar todas as páginas disponíveis.



**Figura 4** - Diagrama de caso de uso

O sistema de doações é estruturado por três classes que formam o diagrama de classe do sistema *web*, como mostrado na Figura 5.



**Figura 5** - Diagrama de classes

O processo para se efetuar uma doação está demonstrado na Figura 6.

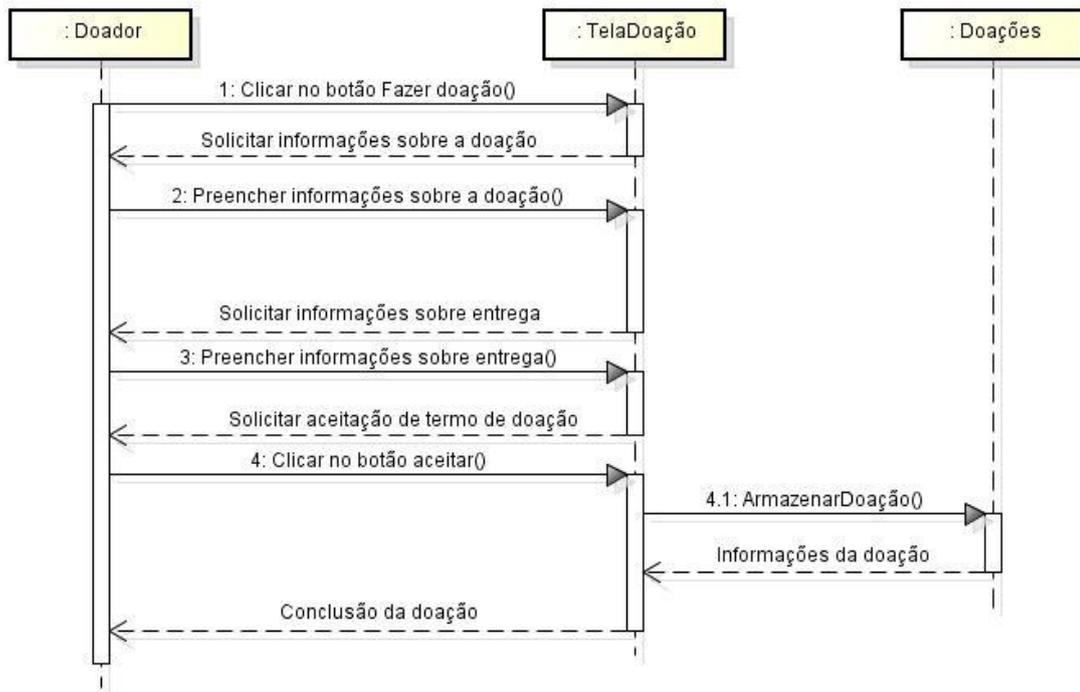


Figura 6 - Diagrama de sequência – Efetuar doação

O *site* foi construído em forma de repositório *online* em que foi utilizada as linguagens de programação *HyperText Markup Language 5* (HTML5), *Hypertext Preprocessor* (PHP) e *Cascading Style Sheets* (CSS) para a construção do *site* e *Structured Query Language* (SQL) para o banco de dados, todos concebidos por meio da ferramenta Notepad++ 7.5.1 e testes realizados utilizando os pacotes de ferramentas do XAMPP 7.1.10: Apache para hospedagem local e MySQL para gerenciar o banco de dados.

Após a conclusão da codificação e testes, o *site* foi hospedado em um servidor *online* (*web server*), bem como seu banco de dados. Para isso, utilizamos o programa FilleZila 3.28.0 para enviar os arquivos do *site* ao servidor por meio de *File Transfer Protocol* (FTP). Em seguida, compramos o domínio “usfgreenit.com.br” e apontamos os endereços dos servidores onde está o *site*, para nosso domínio. Assim, sempre que alguém acessar nosso domínio, será automaticamente redirecionado aos servidores que contem os arquivos do *site*.

Após a finalização do sistema *web*, foram iniciados estudos para criação de um ambiente em nuvem. Foram realizados, primariamente, testes utilizando um projeto chamado *DevStack*, ele requer menos poder computacional do que o projeto original *OpenStack* e possui *scripts* para criar rapidamente um ambiente de desenvolvimento completo e funcional para fazer testes e posteriormente com recursos mais poderosos, faremos uso da versão oficial do *OpenStack*.

Seguindo o guia de instalação da documentação oficial do *DevStack*, o primeiro passo é instalar o sistema operacional *Linux*, executar comandos para adicionar um usuário específico para executar o *DevStack* e, em seguida, baixar o repositório do *DevStack* que contém um *script* para instalá-lo e os modelos para a configuração dos arquivos. Após, é preciso configurar alguns arquivos para começar e, finalmente, executar o *script* que instala o *DevStack*.

Para instalar o *DevStack* no servidor foi usado o sistema operacional Ubuntu Server 14.10 LTS e um computador com as seguintes configurações de *hardware*:

CPU: AMD FX-4300 3,8 GHz Quad-core com 4 MB de cachê L3  
Memória RAM: 4 GB DDR3 670 MHz  
Disco rígido: 500 GB SATA 7200 RPM

Com o *OpenStack* alocado em um servidor, cria-se máquinas virtuais chamadas instâncias contendo um sistema operacional inserido por meio de uma imagem do sistema. Assim, dentro do servidor com *OpenStack*, é possível executar vários outros sistemas ao mesmo tempo e o *OpenStack* controla o uso do processador, a rede, o armazenamento e tudo sobre essas instâncias. Com todo esse acesso disponível apenas por meio do próprio nó do servidor, não é uma ótima vantagem. No entanto, é possível acessar individualmente essas instâncias por meio de uma rede ligada ao servidor. Assim, neste ponto, é encontrado o poder real desse projeto e do *OpenStack*.

Assim, com os componentes doados reaproveitados para compor computadores ainda funcionais, propomos compor uma infraestrutura para reutilizar esses componentes utilizando um servidor com o *OpenStack* implantado para virtualizar os computadores formados.

## RESULTADOS E DISCUSSÃO

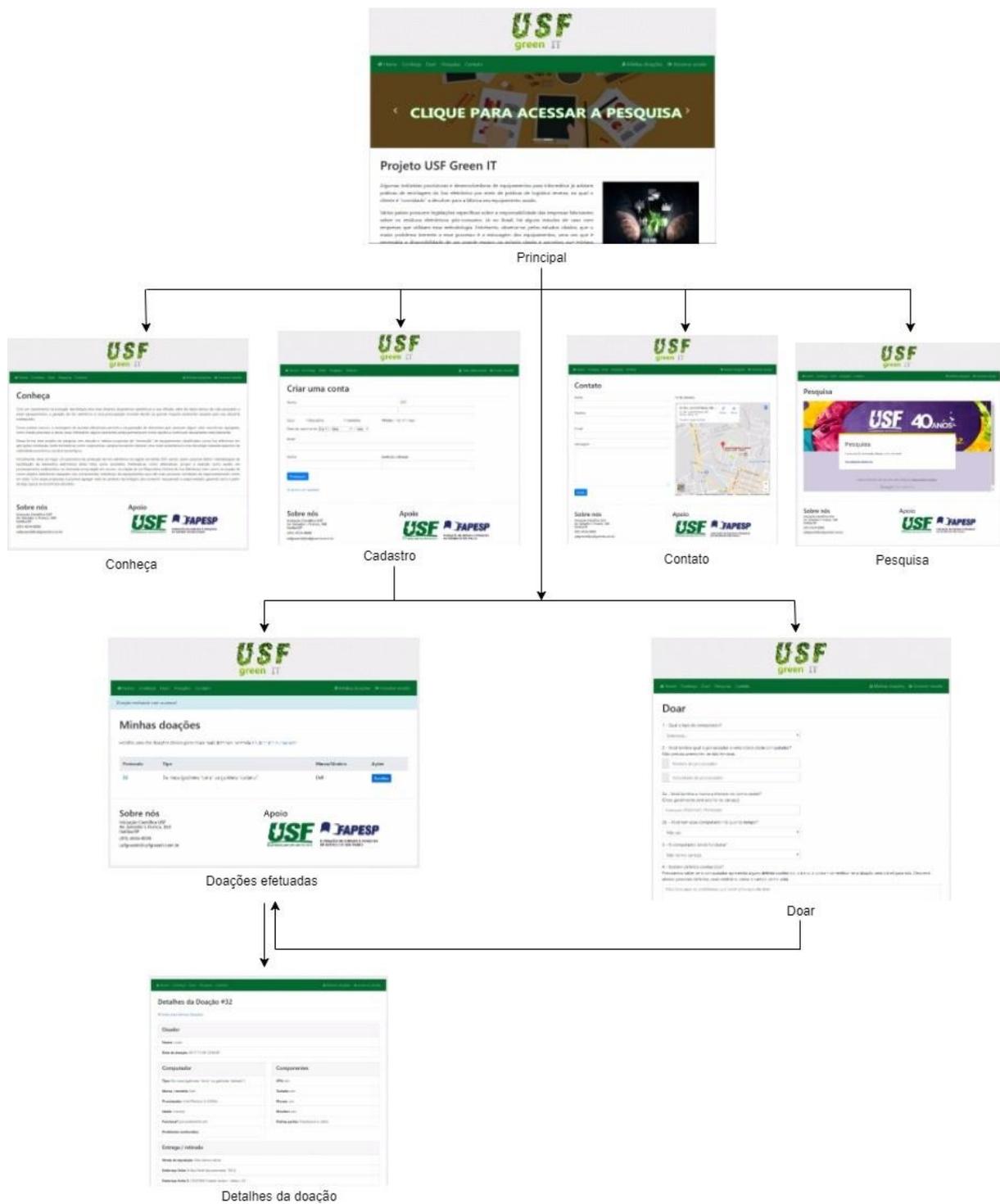
O *site* foi idealizado em um formato que nele seria necessário conter páginas para informar aos visitantes detalhes sobre o projeto, uma página para alocar uma pesquisa e um formulário para que os visitantes conseguissem entrar em contato conosco via *e-mail*. Além disso, o sistema deveria conter uma área específica para registro de usuários para manter maior controle sobre os dados dos doadores e que eles visualisassem facilmente as doações já feitas.

O sistema de doações foi estruturado de forma que um visitante possa se registrar para fazer as doações e gerenciá-las, bem como montar um perfil para que se possa reunir informações de cada responsável pelas doações.

A figura 7 mostra os detalhes da navegação pelo sistema em um diagrama de navegação.

Com um parque computacional construído com base em componentes obsoletos oriundos de doações, propomos reutiliza-los por meio da computação em nuvem, implantando um ambiente que une todo o poder de um servidor atual, com esses computadores defasados, podemos torna-los úteis novamente.

Para iniciar a aplicação do ambiente de computação em nuvem, utilizamos o *DevStack* para implantar a nossa própria nuvem no servidor disponível. Nós seguimos os passos da documentação oficial do *DevStack* e, em pouco tempo, o ambiente estava totalmente disponível para fazer testes.



**Figura 7 - Diagrama de navegação**

Sua instalação forneceu uma boa visualização sobre o comportamento do *OpenStack* e como ele funciona, mostrando-se uma ferramenta realmente poderosa. No entanto, o *DevStack* não é um produto final, para aplicar uma nuvem em uma infraestrutura real, é indicado utilizar uma distribuição oficial do *OpenStack*. Dessa forma, fizemos o uso do *DevStack* apenas por conta da infraestrutura disponível, em que para a aplicação do *OpenStack* é necessário um servidor mais potente em relação ao que tínhamos disponível.

## CONCLUSÃO

Usando a computação em nuvem, ou seja, o *software OpenStack* instalado em um servidor interligado a os computadores obsoletos, é possível fazer a abstração do processamento local dessas máquinas e transferi-los ao servidor, que faz o trabalho por elas.

Foi iniciada a instalação com sucesso do *DevStack* e foi possível fazê-lo funcionar primitivamente. Em próximos trabalhos, sugerimos a aplicação do *OpenStack* em um servidor mais potente que interligado a dezenas de computadores obsoletos, torne possível a construção de um laboratório totalmente funcional com poder de processamento equivalente ou superior caso fosse composto apenas por máquinas atuais. Assim, será viável aplicar a computação em nuvem e ainda reaproveitar muitos computadores.

## AGRADECIMENTOS

Este trabalho foi apoiado pela Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), bolsa n. 2016/16016-1.

## REFERÊNCIAS

BARRET, C.; BRITTEN, T.; CACCIATORE, K.; CHADWICK, P.; PHIPPS, P.; PRÜBMANN, G.; ROSSETTI, M.; SUN, Y.; TAHIR, S.; TRETHERWAY, H.; WU, S. *OpenStack: The Path to Cloud*. Disponível em: <<https://www.openstack.org/assets/marketing/OpenStack-the-Path-to-Cloud.pdf>>. Acesso em Júlio 2017.

JAMSA, K. **Cloud Computing**. 1. ed. Burlington: Jones & Barlett Learning, LLC. 2013.

KURESHI, I.; PULLEY, C.; BRENNAN, J.; HOLMES, V.; BONNER, S.; YVONNE, J. **Using OpenStack to improve student experience in an H.E. environment**. *International Journal of Advanced Computer Science and Applications*, v. 3, n. 4, pp. 64-70, 2013.

MATTOS, R. C. de; SENGER, H. **Serviço Open Source de Bigdata para OpenStack**. *T.I.S. São Carlos*, v. 4, n. 2, p. 128-137, mai-ago 2015.

MELL, P.; GRANCE, T. **The NIST Definition of Cloud Computing**. NIST. 2011.

MERLINO, G.; BRUNEO, D.; DISTEFANO, S. **Stack4Things: integrating IoT with OpenStack in a Smart City context**. Disponível em: <[http://www.academia.edu/13758146/Stack4Things\\_Integrating\\_IoT\\_with\\_OpenStack\\_in\\_a\\_Smart\\_City\\_context](http://www.academia.edu/13758146/Stack4Things_Integrating_IoT_with_OpenStack_in_a_Smart_City_context)>. Acesso em: Júlio 2017.

MÜLLER, M. **Driving the Future of IT Infrastructure at Volkswagen Group**. Disponível em: <[https://www.youtube.com/watch?v=HL\\_pzkDnal4](https://www.youtube.com/watch?v=HL_pzkDnal4)>. Acesso em: Júlio 2017.

OPENSTACK. **Software**. Disponível em: <<https://www.openstack.org/software/project-navigator/>>. Acesso em: Júlio 2017a.

OPENSTACK. **OpenStack Docs: Overview**. Disponível em: <<https://docs.openstack.org/ocata/install-guide-ubuntu/overview.html>>. Acesso em: Júlio 2017b.

OPENSTACK. **OpenStack Docs: Use cases.** Disponível em: <<https://docs.openstack.org/opsguide/app-usecases.html>>. Acesso em: Júlio 2017c.

RADEZ, D. **OpenStack Essentials.** 1. ed. Birmingham: Packt Publishing, 2015.

SHRIVASTWA, A.; SARAT, S. **Learning OpenStack.** 1. ed. Birmingham: Packt Publishing, 2015.

SOUZA, F. R. C.; MOREIRA, L. O.; MACHADO, J. C., **Computação em nuvem: Conceitos, Tecnologias, Aplicações e Desafios.** ERCEMAPI 2009, 2010.

TAURION, C. **Cloud computing – Computação em nuvem: transformando o mundo da tecnologia da informação.** Rio de Janeiro: Brasport, 2009.

VAQUERO, L. M.; CÁCERES, J.; HIERRO, J. J., **Open Source Cloud Computing Systems: Practices and Paradigms.** Hershey: IGI Global, 2012.

VERAS, M. **Virtualização: tecnologia central do datacenter.** 2. ed. Rio de Janeiro: Brasport, 2016.

ZANUTTO, B. G. **Segurança em Cloud Computing.** Disponível em: <<http://dcomp.sor.ufscar.br/verdi/topicosCloud/Artigo-Seguranca-Cloud.pdf>>. Acesso em: Júlio 2017.