



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

FERNANDO HENRIQUE BOECHI

**SOFTWARE COMO SERVIÇO E SUA UTILIZAÇÃO NA
COMPUTAÇÃO EM NUVEM.**

Assis

2011

Av. Getúlio Vargas, 1200 – Vila Nova Santana – Assis – SP – 19807-634
Fone/Fax: (0XX18) 3302 1055 homepage: www.fema.edu.br



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

FERNANDO HENRIQUE BOECHI

SOFTWARE COMO SERVIÇO E SUA UTILIZAÇÃO NA COMPUTAÇÃO EM NUVEM.

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação.

Orientado: Fernando Henrique Boechi

Orientadora: Rita de Cássia Cassiano Lopes

Área de Concentração: Informática

Assis

2011

FICHA CATALOGRÁFICA

BOECHI, Fernando Henrique

Software como Serviço e sua utilização na Computação em Nuvem / Fernando Henrique Boechi . Fundação Educacional do Município de Assis - FEMA -- Assis, 2011.

60p.

Orientadora: Rita de Cássia Cassiano Lopes.

Trabalho de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis – IMESA.

1. *Software* como Serviço . 2. Computação em Nuvem . 3. Segurança da Informação.

CDD:001.6

Biblioteca da FEMA

SOFTWARE COMO SERVIÇO E SUA UTILIZAÇÃO NA COMPUTAÇÃO EM NUVEM

FERNANDO HENRIQUE BOECHI

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, analisado pela seguinte comissão examinadora:

Orientadora: _____

Rita de Cássia Cassiano Lopes

Analisador: _____

Osmar Aparecido Machado

Assis
2011

DEDICATÓRIA

Dedico este trabalho aos meus pais, Aláide e Pedro, pelo grande incentivo, dedicação e confiança que me deram em todos os momentos.

AGRADECIMENTOS

À professora Rita de Cássia Cassiano Lopes, pela orientação e pelo constante estímulo transmitido durante o trabalho.

À minha família, pelo grande incentivo em minha vida para superar os obstáculos, com muito amor e carinho.

Aos professores que me auxiliaram no meu desenvolvimento acadêmico e profissional.

Aos meus amigos em especial Lucas, Marcel, Valter, Gustavo e Marcos Paulo. Aos meus amigos de curso e a todos que colaboraram diretamente ou indiretamente na execução deste trabalho.

EPÍGRAFE

“Tudo o que um sonho precisa para ser realizado é alguém que acredite que ele possa ser realizado.”

(Roberto Shinyashiki)

RESUMO

A Computação em Nuvem surge como um novo paradigma da Tecnologia da Informação e traz novos conceitos ao armazenamento, tratamento e recuperação da informação. Ainda no bojo das inovações, a nova tecnologia traz diferentes conceitos na exploração dos recursos computacionais e descreve um conjunto de ações, que juntas, proporcionam uma melhoria na utilização destes recursos.

A camada de aplicação ou *Software-as-a-Service* (SaaS) é a camada principal do conceito de computação em nuvem, pois é ela que apresenta os maiores benefícios para os usuários. SaaS vem sendo adotado cada vez mais por empresas, e ao passar dos anos este modelo vem ganhando maturidade. Embora este modelo tenha surgido bem antes do conceito de Computação em Nuvem, seus benefícios são mais bem aproveitados com o surgimento da Computação em Nuvem.

A proposta desse trabalho é apresentar os conceitos e benefícios da utilização de *Softwares* como Serviço, vinculados ao conceito de Computação em Nuvem. Por meio da implantação de um *software* de Gestão de Relacionamento com o Cliente, conhecido como CRM, será feito um estudo de caso, que proverá serviços dentro de uma nuvem privada já implantada.

Palavras-chave: Computação em Nuvem, *Software* como Serviço, Segurança da Informação.

ABSTRACT

Cloud computing emerges as a new paradigm of information technology and brings new concepts to storage, treatment and recovery information. Even in the midst of the innovations, new technology brings different concepts in the exploitation of computer resources and describes a set of actions, which together provide an improvement in the use of these resources.

The application layer or Software-as-a-Service (SaaS) is the main layer of the concept of cloud computing, because it is what provides the greatest benefits to users. SaaS is being adopted increasingly by companies, and over the years this model has gained maturity. Although this model has emerged before the concept of cloud computing, its benefits are best enjoyed with the appearance of cloud computing.

The purpose of this paper is to present the concepts and benefits of using Software as a Service, linked to the concept of cloud computing. Through the implementation of a Customer Relationship Management Software, known as CRM, will be a case study that will provide services within a private cloud already implemented.

Keywords: Cloud Computing, Software as a Service, Information Security.

LISTA DE ILUSTRAÇÕES

Figura 1 – Visão geral da computação em nuvem.....	21
Figura 2 – Modelo de serviços na computação em nuvem.....	23
Figura 3 – Papéis na computação em nuvem.....	24
Figura 4 – Tipos de Nuvem.....	25
Figura 5 – Desenvolvimento de uma nuvem confiável.....	35
Figura 6 – Gerenciamento das Máquinas Virtuais.....	38
Figura 7 – Serviços disponibilizados pelo Windows Azure.....	41
Figura 8 – Google Doc`s.....	43
Figura 9 – vTiger CRM.....	49
Figura 9 – Resultados do tempo de resta do clique por URL.....	54
Figura 10 – Gráfico de requisições abertas e transferência de dados.....	55

LISTA DE TABELAS

Tabela 1 – Descrição das URL`s para os testes.....	53
Tabela 2 – Erros de acesso a URL.....	54

LISTA DE ABREVIATURAS

CRM	Gestão de Relacionamento com o Cliente
ERP	Sistemas Integrados de Gestão Empresarial
EUA	Estados Unidos da América
IaaS	<i>Infrastructure-as-a-service</i>
IBM	International Business Machines
PaaS	Plataforma como serviço
SaaS	<i>Software</i> como serviço
SOA	Arquitetura Orientada a Serviços
TI	Tecnologia da Informação
URL	Localizador-Padrão de Recursos
VM	Máquina Virtual

SUMÁRIO

INTRODUÇÃO.....	15
OBJETIVO GERAL.....	16
OBJETIVO ESPECÍFICO.....	17
JUSTIFICATIVA.....	17
ORGANIZAÇÃO DO TRABALHO.....	18
REVISÃO DA LITERATURA.....	18
1. COMPUTAÇÃO EM NUVEM.....	20
1.1 DEFINIÇÃO.....	20
1.2 MODELOS DE NUVEM.....	23
1.3 TIPOS DE NUVEM.....	25
2. SOFTWARE COMO SERVIÇO.....	26
2.1 DESENVOLVIMENTO DE SAAS.....	28
2.1.1 Custos.....	31
2.1.2 Geração de receitas.....	32
3. SEGURANÇA EM COMPUTAÇÃO EM NUVEM.....	33
4. FERRAMENTAS QUE OPERAM NA NUVEM.....	37
4.1 EXEMPLOS DE IAAS.....	37
4.1.1 Amazon Web Services.....	37
4.1.2 Eucalyptus.....	39
4.2 EXEMPLOS DE PAAS.....	40
4.2.1 Google App Engine.....	40
4.2.2 Windows Azure.....	41
4.3 EXEMPLOS DE SAAS.....	42
4.3.1 Google Doc`s.....	43
4.3.2 Social Enterprise.....	44
5. ESTUDO DE CASO.....	45
5.1 CRM.....	45

5.2 A EMPRESA.....	47
5.3 SOLUÇÃO.....	48
5.3.1 vTiger CRM.....	49
5.4 IMPLANTAÇÃO NA EMPRESA.....	49
5.4.1 Instalação da Solução.....	51
5.5 TESTES.....	52
5.6 AVALIAÇÃO DOS RESULTADOS.....	55
CONSIDERAÇÕES FINAIS	57
REFERÊNCIAS.....	58

INTRODUÇÃO

A maneira como adquirimos *software* hoje em dia está ultrapassada; existe a necessidade de instalá-lo e atualizá-lo para cada usuário final, desperdiçando um tempo que poderia ser utilizado para outras funções, como a criação de novas soluções para o foco de negócios da empresa, por exemplo.

“*Cloud computing* pode ser definido como um modelo no qual a computação (processamento, armazenamento e *softwares*) está em algum lugar da rede e é acessada remotamente via Internet.” (GARCIA, 2010).

Dentro deste contexto, “A Computação em Nuvem aparece como uma alternativa, pois aloca recursos computacionais à medida que eles sejam demandados.” (TAURION, 2009, p. 9). Dependendo do número de transações, os recursos podem ser aumentados ou diminuídos rapidamente.

Na computação em nuvem, existem diversos tipos de serviços, não sendo apenas a infraestrutura. Esses serviços podem ser divididos em 4 níveis: a camada de infraestrutura; a camada de desenvolvimento e gerenciamento da nuvem; a camada de aplicações; e por último a camada de negócios.

A camada de aplicação ou *Software-as-a-Service* (SaaS) é a camada principal do conceito de computação em nuvem, pois é ela que apresenta os maiores benefícios da computação em nuvem para os usuários.

SaaS vem sendo adotado cada vez mais empresas, e com passar do tempo este modelo vem ganhando maturidade. Da necessidade de utilização de diversas plataformas, surge um novo modelo de negócios, no qual empresas oferecem seus *softwares* via internet, que são executados por meio do navegador do cliente final, independente da plataforma e recursos físicos utilizados por ele, evitando problemas com instalação, atualização, *hardware* e suporte.

Neste modelo, o fornecimento do *software* é responsabilidade do fornecedor, bem como toda a manutenção, operação técnica e suporte fica a cargo do fabricante. A empresa que contratou o *software* economiza seus recursos financeiros e computacionais.

Um ponto importante é que o mesmo *software* pode ser utilizado por outras empresas, através de uma arquitetura multi-inquilino, que compartilha os mesmos recursos físicos separados logicamente.

Muitos fatores são responsáveis pela grande adoção deste modelo, entre eles a rapidez para se começar a operar e o retorno sobre investimento (ROI). Segundo dados publicados pela *Microsoft* (2009), o mercado de SaaS envolve U\$ 5 bilhões, e ainda segundo o *Gartner Group*, SaaS representou cerca de 5% do mercado total de *software* no mundo entre 2005 até 2011. Ainda deverá representar 25% das vendas totais para o segmento corporativo.

Este trabalho irá abordar diversos conceitos relacionados ao modelo de *software* como serviço em computação em nuvem, auxiliando como material teórico para futuras pesquisas sobre o tema.

Será apresentado também um Estudo de Caso de implantação de um *software* de Gestão de Relacionamento com o Cliente, conhecido como CRM, fornecido por meio de serviço, dentro de uma nuvem privada já implantada na empresa.

OBJETIVO GERAL

O objetivo geral deste trabalho é abordar os conceitos relacionados ao uso de SaaS (*Software-as-a-Service*), dentro do conceito de Computação em Nuvem, no qual o usuário será o grande impulsionador, interessado em obter respostas rápidas. Será realizado levantamento de dados e apresentado um breve histórico da tecnologia de computação em nuvem, o conceito da utilização de *software* como serviço, a realidade pelo uso de empresas, a visão sobre a indústria de *software*, os custos de implantação e desenvolvimento, o futuro do mercado, as vantagens e benefícios, algumas ferramentas que já operam em nuvem, entre outros. O resultado desta pesquisa será mostrado através de um estudo de caso.

OBJETIVO ESPECÍFICO

Para que o Objetivo Geral possa ser realizado têm-se os seguintes objetivos específicos:

- Auxiliar a área administrativa na escolha do CRM;
- Analisar o desempenho do sistema na nuvem privada;
- Apresentar um estudo de caso da implantação do CRM.

JUSTIFICATIVA

O modelo de SaaS é parte principal do conceito de computação em nuvem. É onde o usuário final consegue enxergar os benefícios da tecnologia. Embora este modelo tenha surgido bem antes do conceito de computação em nuvem, seus benefícios são mais bem aproveitados com o surgimento deste conceito.

O SaaS é uma fórmula de distribuição e utilização de *softwares*, ou seja, o *software* passa a ser uma prestação de serviço.

Em longo prazo a adoção pelas empresas do uso de *software* como serviço trará muitos benefícios, seja na redução de custos ou nos recursos de tecnologia. Além disso, a empresa economiza tempo, pois a implementação neste modelo é muito mais rápida e prática, já que não necessita da instalação em cada usuário final.

Por ser um *software* executado diretamente da internet, a integração com diversas plataformas é mais simples, o usuário pode acessar de qualquer lugar e há uma maior integração com outras filiais da empresa.

Outro fator importante quanto à utilização de SaaS é a flexibilidade do *software*, que pode ser atualizado para todos os usuários de uma única vez, evitando assim diversos problemas, como perda de tempo e versões incompatíveis.

ORGANIZAÇÃO DO TRABALHO

A monografia será estruturada em 5 capítulos, quais sejam: Capítulo 1, que abordará os conceitos relacionados a Computação em Nuvem. Capítulo 2, que trará conceitos referentes ao *Software* como Serviço. No capítulo 3, relacionará questões de segurança. No capítulo 4, serão apresentadas algumas ferramentas que já operam em nuvem. E finalmente, será mostrado um Estudo de Caso da implantação de CRM em uma nuvem privada.

REVISÃO DA LITERATURA

O cenário da Tecnologia da Informação (TI) sofre cada vez mais mudanças, novos modelos e arquiteturas surgem com o avanço da tecnologia, fatores mundiais como guerras e mais recentemente a crise econômica são os grandes responsáveis por estes avanços.

Na visão de Taurion (2009, p. 103), “Provavelmente, a crise de escassez de recursos tende a acelerar os modelos de negócios”, crescendo assim a utilização do modelo de *Software* como serviço.

Segundo MOTA (2009, p. 17):

Software as a Service (SaaS) é uma tendência na indústria de *software* que se caracteriza por ser um modelo de negócios que oferece às organizações o acesso centralizado de suas informações por meio de hospedagem e acesso pela internet, por um custo menor em relação ao modelo de *software* tradicional.”

“Em poucas palavras, SaaS é um *software* distribuído como um serviço, implementado em plataforma web de forma nativa e acessado tecnologias e protocolos de internet.”(CAMBUCCI, 2009).

A grande vantagem da utilização do *software* como serviço é: “o fornecedor é responsável pela distribuição, a operação, e a manutenção da infra-estrutura de TI. Ele opera tipicamente em uma única instância da aplicação e controla todas as atualizações.” (MELO et al, 2009).

1. COMPUTAÇÃO EM NUVEM

Esta é uma nova idéia que vem amadurecendo com a evolução constante da TI. Com o interesse massivo pela grande maioria de executivos, o termo computação em nuvem aparece como uma nova visão da computação.

Em 2006, em uma palestra Eric Schmidt, fundador do Google, utilizou pela primeira vez o termo computação em nuvem. Depois de um tempo, novas soluções em nuvem foram surgindo, além da *Google*, a *Amazon* com *Elastic Cloud Computing (EC2)*.

1.1 DEFINIÇÃO

O termo nuvem e a sua imagem vêm de uma metáfora utilizada nos diagramas de redes, para descrever a internet, algo que não estava ao alcance, intangível. Hoje, com a computação em nuvem, aplicações podem executar ou usar recursos da nuvem. A computação em nuvem passa a ser o centro da computação.

Como computação em nuvem é algo novo e está sendo muito abordada em diversos trabalhos acadêmico e por diversos autores, encontram-se várias outras definições.

Conforme Taurion (2009, p. 2), uma definição simples pode ser:

“Um conjunto de recursos como capacidade de processamento, armazenamento, conectividade, plataformas, aplicações e serviços disponibilizados na Internet”.

A figura 1 mostra uma visão geral da definição, auxiliando no entendimento.

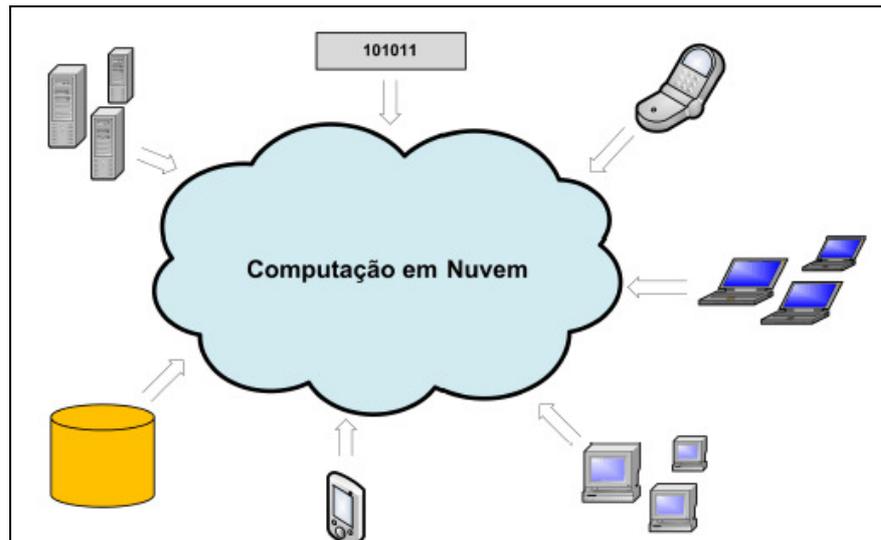


Figura 1 – Visão geral da computação em nuvem (SOUSA ET al. 2009).

Podemos então destacar que computação em nuvem possui algumas características principais. Conforme Taurion (2009) pode-se resumir em:

- Ilusão da disponibilidade de recursos infinitos acessáveis sob demanda.
- Eliminação da necessidade de adquirir recursos e provisionar recursos antecipadamente.
- Elasticidade permite que os recursos sejam utilizados na quantidade que forem necessários, aumentando ou diminuindo de forma dinâmica.
- *Pay-per-user*, o pagamento é pela quantidade de recursos utilizados.

Algumas tecnologias estão relacionadas com a computação em nuvem, sendo assim a computação aparece como evolução natural da convergência destas tecnologias.

Podemos destacar:

- *Grid Computing* ou Computação em Grade é um tipo de computação, que utiliza diversos computadores interconectados para realização de processamento paralelos, utilizados em pesquisas científicas e médicas,

assim como previsão do tempo, pois exigem um grande ciclo e acesso a muitos dados. O usuário não percebe a utilização, pois o foco está apenas em realizar o processamento, já que são utilizadas grades em localizações geográficas diferentes, e essas grades possuem um baixo acoplamento entre si e sistemas heterogêneos.

- *Utility Computing* pode ser entendido como serviços utilitários ou computação de vantagens, que tem como modelo de negócio a cobrança pelo que é consumido. É uma excelente alternativa para as empresas, pois gera uma economia e controle dos recursos computacionais, pode se comparar este modelo à forma de cobrança de energia elétrica, água e gás.
- Virtualização é uma tecnologia adotada para dividir os recursos de um computador em múltiplos ambientes de execução usando algumas técnicas. Particionamento de hardware e *software*, compartilhamento de tempo, simulação completa ou parcial da máquina e emulação são as técnicas utilizadas.
- *Web 2.0* com suas aplicações, pois utiliza poucos recursos para seu funcionamento. A *web 2.0* é a mudança na forma de desenvolver para web, tornando-a mais dinâmica, participativa e colaborativa pelos seus usuários. Esses atrativos, juntamente com a computação em nuvem tornam um modelo principal na distribuição de *software*.
- *Service-Oriented Architecture (SOA)* é uma arquitetura para desenvolvimento de *software*, orientada a serviços, que possui ações únicas, que pode ser usada por várias aplicações distintas. A principal vantagem da utilização desta arquitetura é a máxima reutilização de código, aproveitando em novas aplicações ou mantendo as existentes.
- *Software* como serviço é modelo de entrega do *software*, ou seja, a empresa contrata o *software* como uma prestação de serviço.

1.2 MODELOS DE NUVEM

A computação em nuvem é dividida em três modelos de serviços, modelos que definem um padrão de arquitetura para soluções de computação em nuvem, são eles: *Software-as-a-service (SaaS)*, *Plataforma-as-a-service (PaaS)* e *Infraestrutura-as-a-service (IaaS)*. A figura 2 ilustra a classificação deste modelo:

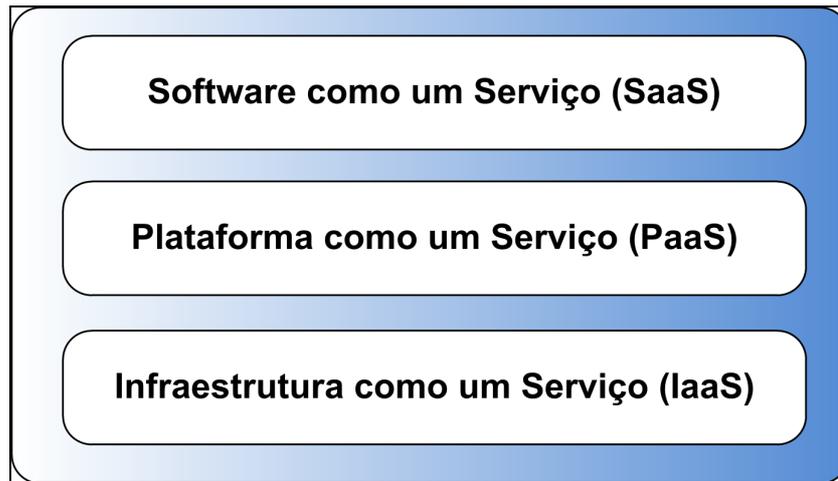


Figura 2 – Modelo de serviços na computação em nuvem (SOUSA ET al. 2009).

Podemos resumidamente descrever os modelos:

- *Infraestrutura-as-a-service (IaaS)* ou Infraestrutura como serviço, é a base de toda a arquitetura da computação em nuvem. Ela provê toda a infraestrutura necessária para o funcionamento dos outros modelos de negócios, tornando o acesso fácil aos recursos fornecidos, como servidores, rede, armazenamento e outros recursos. É neste modelo que são utilizadas as técnicas de virtualização, criando assim um ambiente no qual é possível escalar dinamicamente os recursos, aumentando ou diminuindo-os, dependendo da demanda das aplicações. São exemplos deste modelo, *Elastic Cloud Computing (EC2)* da *Amazon* e *Elastic Utility Computing Architecture Linking Your Programs To Useful System (Eucalyptus)* solução *open source*.
- *Plataforma-as-a-service (PaaS)* ou Plataforma como serviço, é uma estrutura de alto nível, ou seja, os usuários da nuvem não tem acesso diretamente a camada IaaS. Os usuários da PaaS são desenvolvedores de aplicações, que têm disponível neste modelo o controle e configurações destas aplicações,

hospedadas nesta infraestrutura. Geralmente, os ambientes fornecidos, possuem uma restrição no número de aplicações suportadas pela plataforma, pois existem limitações quanto ao ambiente, como sistema operacional e linguagem de programação específicos para este modelo. O *Google App Engine* da Google é um exemplo.

- *Software-as-a-service* (SaaS) ou *Software* como serviço será abordado mais especificamente no capítulo 2, é camada de aplicação, a qual fornece ao usuário final aplicações como serviço. São exemplos deste modelo, *Google Docs* da Google e *LotusLive* da IBM.

Dentro de um ambiente em nuvem, para melhor entendimento, os modelos dividem-se em “atores”, de acordo com os papéis empenhados. Segundo Sousa et al. (2009), o provedor é responsável pela disponibilização, gerenciamento e monitoramento de toda infraestrutura, podendo disponibilizar serviços nos três modelos. Os desenvolvedores utilizam os recursos fornecidos e disponibilizam para o usuário final. A figura 3 destaca estes atores.

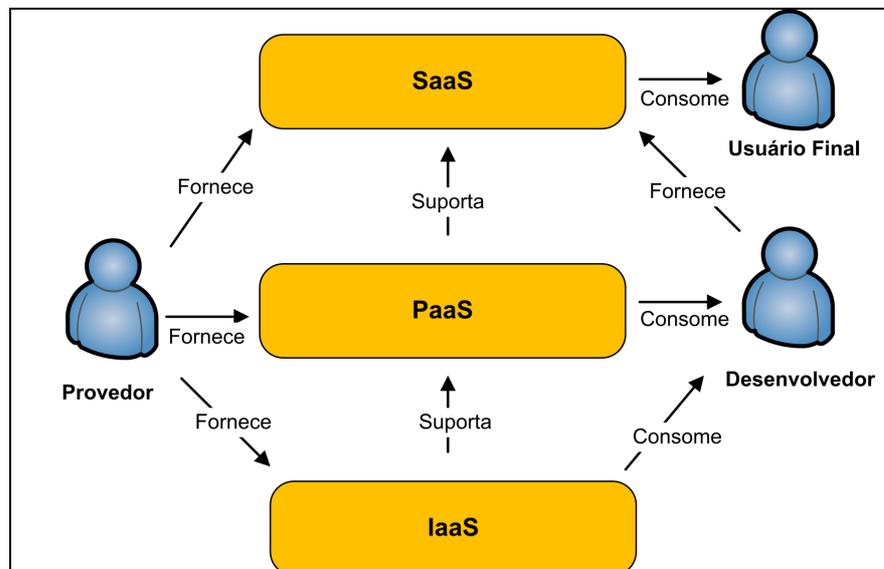


Figura 3 – Papéis na computação em nuvem (SOUSA et al, 2009).

1.3 TIPOS DE NUVEM

A nuvem, dependendo da sua utilização, é dividida em 3 tipos básicos. A definição do tipo que a nuvem deve ser dependerá do negócio, da informação e nível de visão da empresa. Os tipos de nuvem são divididos em:

- Nuvem pública: nuvem fornecida por provedores a uma empresa ou usuário, ficando assim o provedor responsável por toda sua infraestrutura, pode ser adquirido de forma simples através da web. Desta forma, o usuário consegue em poucos minutos ter acesso a recursos necessários para seu negócio.
- Nuvem privada: toda a infraestrutura da nuvem é utilizada exclusivamente pela empresa, sendo administrada pela própria empresa, localmente ou remota, juntamente às políticas de segurança da empresa. Algumas técnicas são utilizadas para gerenciamento deste tipo, como gerenciamento de redes e tecnologias de autorização e autenticação.
- Nuvem híbrida: é a junção de duas ou mais nuvens, que podem ser privadas ou públicas, permitindo assim a portabilidade dos dados, como integração de aplicações.

Para ilustrarmos estes tipos de nuvem, utilizamos a Figura 4.

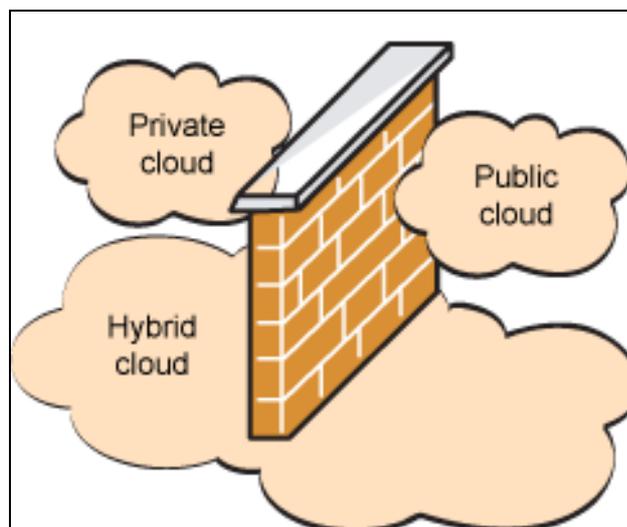


Figura 4 – Tipos de Nuvem (CCBRAZIL, 2010).

No capítulo 2, será abordado o *software* como serviço, parte principal do modelo de Computação em Nuvem.

2. SOFTWARE COMO SERVIÇO

O software é um modelo de entrega, onde o usuário passa a utilizá-lo somente, sem se preocupar com a instalação, manutenção e atualização. Um *software* SaaS é distribuído através da internet, podendo ser instalado em um ou mais servidores, acessado pelo navegador do usuário final.

Este modelo surgiu em meados dos anos 90, juntamente com a evolução de diversas tecnologias, como redes, ferramentas de programação e capacidade computacional.

A maneira de como adquirimos *softwares* hoje, através do licenciamento, a implantação e manutenção está se tornando complexa e isto está gerando um enorme custo. Levando em conta este contexto, os negócios estão exigindo maior rapidez e menores custos para área de TI. Tudo isso tornou o conceito de que o *software* possa ser entregue como serviço, reduzindo os custos quanto ao modelo atual.

A forma como o fornecedor entrega este *software*, altera o modelo de negócios, pois o fornecedor fica responsável pela infraestrutura, implementação e gerenciamento do *software*. Para o cliente, esta modificação reduz os custos e os recursos computacionais, pois este *software* será constantemente atualizado, para todos os usuários, de uma única vez, evitando assim também perda de tempo e versões incompatíveis.

Outra vantagem para o fornecedor é que este mesmo *software* pode ter vários clientes, tornando assim os custos dos serviços menores. Assim sendo, cada cliente é responsável por apenas uma parte dos custos de um servidor de hospedagem desta aplicação.

Esses vários clientes são chamados de inquilinos, permitindo então ao provedor oferecer o mesmo *software* para vários inquilinos, tornando este *software* uma aplicação multi-inquilino.

Com essa arquitetura, o *software* passa a ser usado por vários inquilinos, sendo que o inquilino não visualiza ou tem acesso aos dados de outro inquilino, criando uma ilusão que ele é o único a acessar o aplicativo.

Atualmente, as soluções presentes no mercado são aplicativos departamentais, como CRM, Pacote *Office* e entre outros, mas com o passar o tempo, o mercado exigirá soluções mais complexas e abrangentes.

Ainda há dúvidas quanto à adoção deste modelo, segurança e confiabilidade dos dados são obstáculos para utilização em massa do *software* como serviço.

O *software* como serviço pode ser dividido em duas principais categorias:

- Serviço orientado ao negócio: soluções voltadas para o meio corporativo, geralmente são soluções personalizadas para os processos de negócios como finanças, relacionamento com o cliente, entre outros.
- Serviço orientado ao cliente, público em geral: soluções que geralmente são gratuitas, como *webmail*, redes sociais, que são financiados por anúncios.

O SaaS não deve ser confundido com contexto de computação em nuvem. Existem hoje muitos equívocos entre o conceito de computação em nuvem e *software* como serviço. *Software* como serviço é apenas um modelo que utiliza a computação em nuvem para melhorar suas vantagens e benefícios.

Segundo Desisto, Plummer & Smith (2008), “computação na nuvem é um tema mais abrangente que não pode ser confundido com SaaS, esse utiliza a computação na nuvem apenas em sua arquitetura.”

Taurion (2009, p. 102) destaca que:

“Software-as-a-Service é um modelo disruptivo. Sua proposição de valor é funcionalidade oferecida e não a “propriedade” do produto. A idéia básica é que você na verdade não quer uma máquina de lavar roupa, mas quer a roupa lavada. SaaS oferece isso. Você não necessita instalar um pacote de CRM ou ERP, mas precisa das suas funcionalidades.”

“O efeito combinado de SaaS, SOA e Open Source vão transformar por completo o que nós entendemos que seja hoje a indústria de *software*.” acrescenta TAURION (2009, p.102).

2.1 DESENVOLVIMENTO DE SaaS

O desenvolvimento de um *software* para ser disponibilizado como um serviço muda em relação ao desenvolvimento tradicional de um *software*. Além do entendimento do ambiente do modelo SaaS, questões técnicas são importantes.

Muitas empresas de *software* modelam sua estratégia de negócios, incluindo em seus portfólios produtos e serviços em nuvem. Mas a transição para este modelo custa caro, empresas gastam a metade de suas receitas em vendas e *marketing*.

Certamente, o ponto crítico em relação à utilização de SaaS é a confiança em um terceiro para cuidar e guardar os dados de uma organização, que são umas das coisas mais importante desta.

Para se desenvolver um SaaS podem surgir algumas exigências por parte do cliente; ele pode determinar que sua base de dados seja separada dos outros clientes deste fornecedor. Mas para o fornecedor o ideal seria uma única base para todos os clientes, apenas separados logicamente. Caso o cliente tenha essa exigência, ele deverá estar ciente que o custo aumentará.

Uma solução SaaS pode trazer vários benefícios para um empresa, que fazem a diferença no negócio. Para se entender essas transformações são utilizados níveis de maturidade para medi-las. Esses níveis são:

- Primeiro nível: *ad-hoc*/personalizado, é o primeiro estágio de alteração do *software* tradicional para o *software* como serviço, essas aplicações mudam apenas de local, ou seja, passam a ser executada atrás da rede ou internet, acessada pelo navegador. O cliente tem sua própria versão do *software*, sem compartilhar com outros clientes deste fornecedor. Este nível apenas

proporciona para o fornecedor uma redução nos custos de integração da administração e hardware. Para o cliente, o fornecedor fica responsável pelo fornecimento, administração e atualização.

- Segundo nível: configurável, da mesma forma que o primeiro nível, pois as instâncias ainda continuam separadas. A diferença é que esta instância é única para cada cliente e o código fonte será compartilhado. Neste nível o fornecedor oferece as opções de configuração aos seus inquilinos. Qualquer mudança no código é alterada a todos os outros clientes, resultando na diminuição de custos com manutenção e a necessidade de atualização individual.
- Terceiro nível: configurável e eficiente para vários inquilinos, neste nível executam uma única instância deste *software* para vários inquilinos, há um tratamento de metadados, manutenção e configuração de banco de dados, ou seja, um único banco de dados para todos os clientes, separado logicamente. Por ser uma aplicação que oferece um compartilhamento total dos recursos e uma única instância para vários inquilinos, o custo é dividido, e consequentemente o custo com *hardware* vai diminuindo. Esta aplicação é uma aplicação multi-inquilino.
- Quarto nível: escalonável, configurável e eficiente para vários inquilinos, como no terceiro nível, com banco de dados separados, cada inquilino possui seu banco de dados, exigindo assim um hardware mais elevado. Para suportar a demanda, o fornecedor deve ter vários servidores fazendo a mesma tarefa. Conforme a demanda, as tarefas passam para outros servidores, surgindo a escalabilidade.

Resumidamente, o primeiro nível ad-hoc/personalizado, aplicação hospedada, o cliente passa a ter suas aplicações cuidadas por um terceiro, em um ambiente ainda separado para cada cliente. No segundo nível, passa a ter um compartilhamento de recursos, mas ainda cada cliente tem sua base de dados separada, configurável isolada fisicamente ou virtualmente. No terceiro nível, a aplicação passa a ser compartilhada sem que o cliente perceba. Configurável e multi-inquilino, este cliente

passa a compartilhar a mesma aplicação com outros clientes. Finalmente, o ambiente passa a ter um balanceamento de carga, escalável, configurável e multi-inquilino, e dependendo da demanda o processamento pode ser executado por outros servidores sem alteração nenhuma na aplicação.

O ciclo de desenvolvimento de um SaaS tem como base 3 princípios. Segundo Melo et al (2007, p. 3) são “licenciamento, localização e gerenciamento do ciclo de vida da aplicação”.

No modelo SaaS, o licenciamento da aplicação é por assinatura, o cliente não se preocupa com customização, hardware e implementação, diferente do modelo tradicional, no qual o cliente deve pagar um valor único e elevado pelo *software*. Este modelo de licenciamento beneficia-se da economia de escala, e em longo prazo, quanto mais cliente mais baixos são os custos para o consumidor.

A localização é onde a aplicação esta instalada. Pode ser por terceiros, acessada pela internet ou por *appliance*, amenizando os problemas da distancias de redes e largura de banda. Esta última opção, segundo Melo et al (2007, p. 5):

Nesse modelo uma aplicação pré-configurada é arrendada e instalada no cliente. Normalmente, os dados são processados e colocados em cachê no *appliance* para diminuir o número de micro-transações e aumentar a velocidade do acesso do usuário. Periodicamente, partes dos dados são sincronizados com os dados localizados no provedor de serviço de *software*.

No gerenciamento do ciclo de vida da aplicação, o fornecedor deve conhecer seus clientes, a segurança, performance e disponibilidade. O fornecedor deve cuidar da TI do cliente.

O grande problema é a latência de rede, a largura de banda atual das empresas podem não suportar o numero de transações feitas por um SaaS. Este problema pode gerar enormes lentidões na utilização do *software* como serviço, sendo assim, um grande problema para adoção do mesmo.

Soluções para disponibilização do *software* como serviço em uma nuvem privada estão surgindo, devido ao negócio da empresa a disponibilidade do serviço passa a ser muito importante. São vários fatores externos que podem prejudicar, entre eles: incêndios, roubos, ataques de *hackers* e entre outros.

SaaS exige uma relação entre o fornecedor e cliente muito forte, o *backup* e restauração, segurança e portabilidade dos dados devem ser fornecidos como prioridade, de fácil acesso ao cliente, para eventuais problemas, conseguir migrar para outra solução rapidamente, sem afetar os seu negócios.

O ideal para um *software* como serviço á utilização de serviços de integração por meio de SOA, oferecendo mecanismos fácies e rápidos, sem a necessidade de esforços no desenvolvimento.

A escalabilidade da aplicação é o diferencial da aplicação SaaS, pois ela deve suportar toda as requisições de seus inquilinos, ao mesmo tempo melhorando ou diminuindo o tráfego da rede.

2.1.1 Custos

A adoção de SaaS deve ser comparada aos gastos envolvidos e estimativa de uso das funcionalidades oferecidas. Colocando no papel, o custo de licenciamento de uma solução tradicional acaba sendo a menor parte do custo total da implantação, pois este *software* necessitará de um *hardware* no usuário final, custo de pessoal para operação e manutenção do *software*, tempo e treinamento dos usuários.

Esses fatores desaparecem no modelo SaaS, pois todos estes custos ficam a cargo do fornecedor, que terá seus lucros através da economia de larga escala, vendendo a mesma solução para vários clientes.

Surge então, a teoria da cauda longa, que foi descrita no livro "*The Long Tail*" de Chris Anderson, segundo CAMBIUCCI (2009), "o objetivo dessa abordagem é disponibilizar os mesmos recursos de *software* para um número muito maior de clientes. E essa visão tem suas bases no conceito da "*Cauda Longa*"."

CAMBIUCCI (2009) intera que:

Na cauda longa, um número muito grande de usuários poderá adotar nossa solução pagando pelo uso, por demanda, o que deve gerar um valor muito baixo de pagamento ou ticket. Nesse cenário, estamos realmente buscando o chamado "milhões de mercados de poucos" ao invés dos atuais "poucos mercados de milhões", sendo a essencial do modelo baseada na Cauda Longa.

2.1.2 Geração de receitas

O fornecedor de SaaS pode utilizar variadas formas na sua geração de receitas, ele pode optar por um pagamento mensal, trimestral, anual, cobrança de assinatura ou volume de transações.

Esses tipos de cobranças geram três tipos de clientes, cada um focado no tipo de cobrança, são eles: de maior valor, de valor negativo e valor médio. O de maior valor responsável pela maior parte de lucro, o de valor negativo resulta em prejuízo, pois o valor pago não cobre os custos.

Um cliente de valor negativo geralmente opta pelo tipo de pagamento pelo volume de transações e os valores geralmente não cobrem os custos para disponibilização deste serviço.

A fidelidade de clientes é o grande diferencial das empresas, elas devem criar laços fortes com seus clientes, seja financeiramente, por produtos, serviços ou marca diferenciada. Os clientes não estão apenas preocupados com a redução dos custos, eles também esperam que os benefícios sejam grandes.

No próximo capítulo, será discutida a segurança em computação em nuvem, principal fator para adoção de computação em nuvem e software como serviço.

3. SEGURANÇA EM COMPUTAÇÃO EM NUVEM

Para aproveitar todos os benefícios da computação em nuvem, grandes mudanças devem ocorrer nos departamentos de TI das organizações. Dentre essas mudanças está a redução do controle dos dados, como a confidencialidade e conformidade. Essa falta de controle faz com que a segurança seja umas das questões mais importantes ao considerar a adoção deste modelo.

Como toda nova tecnologia a questão da segurança e receio diante de uma novidade, sempre aconteceu, foi assim com o modelo cliente-servidor e no surgimento do comércio eletrônico com a utilização do cartão de crédito, ações que hoje são normais para qualquer pessoa.

A computação em nuvem, assim como qualquer outra tecnologia gera novos riscos, não é necessariamente mais ou menos segura que o ambiente atual. Novas oportunidades são criadas, exigindo uma reestruturação das aplicações antigas, assim como uma reestruturação da infraestrutura a fim de garantir a segurança.

Como qualquer outra área de segurança, a adoção do modelo de computação em nuvem deve ser baseada nos riscos dos ativos que estão sendo migrados para fora do domínio de uma organização, ou seja, o controle das informações passa a ser feito por um fornecedor externo.

Antes de se discutir a segurança em *cloud computing*, deve-se separar as nuvens pelo seu tipo, mais precisamente nuvem privada de nuvem pública. No contexto da nuvem privada, as políticas de segurança já são adotadas pela organização, pois cada organização tem os seus métodos e processos, assim como a tolerância aos riscos.

Em uma nuvem pública os métodos e processos de garantia da segurança ficam a cargo do provedor da nuvem, ele deve implementar e gerenciar esses métodos e processos, pois a segurança é primordial para o sucesso do seu negócio. Para reforçar a garantia, vários provedores de nuvem optam por passarem por auditorias externas como SAS 70 e certificações oficiais como ISO 27001.

Na adoção de qualquer opção de implantação de nuvem, não existe um caminho correto para garantir uma avaliação dos riscos, nem uma metodologia para determinar todos os requisitos de segurança. Utiliza-se um método rápido de avaliação dos ativos da organização, a fim de decidir se esse ativo é tolerante aos princípios básicos da segurança da informação.

Primeiramente deve se determinar exatamente qual dado ou função será movido para nuvem, assim como o volume desses dados e suas transações. Deve-se ser levado em conta questões de padrão de segurança e compatibilidade, como autenticação sólida e delegada, chaves de criptografia, proteções contra perda de informações e emissão de relatórios para cada tipo de ativo.

Outra forma de tomar decisão é através do mapeamento do fluxo de dados, por meio do qual será possível esclarecer da melhor forma possível o fluxo dos dados entre a organização e o provedor ou serviço de nuvem.

A responsabilidade da segurança esta relacionada com o modelo de serviço de nuvem adotado. Os modelos de serviços, SaaS, PaaS e IaaS, seguem um modelo de pilha, sendo a primeira camada o IaaS, a segunda o PaaS e terceira SaaS.

No modelo SaaS, a responsabilidade é do prestador de serviço, pois é ele que gerencia todo *software*. Em PaaS e IaaS a segurança a responsabilidade final é do desenvolvedor, os níveis de segurança de infraestrutura mais baixas são de responsabilidade do provedor de nuvem.

Pode-se ilustrar o desenvolvimento de uma nuvem confiável com a Figura 5.

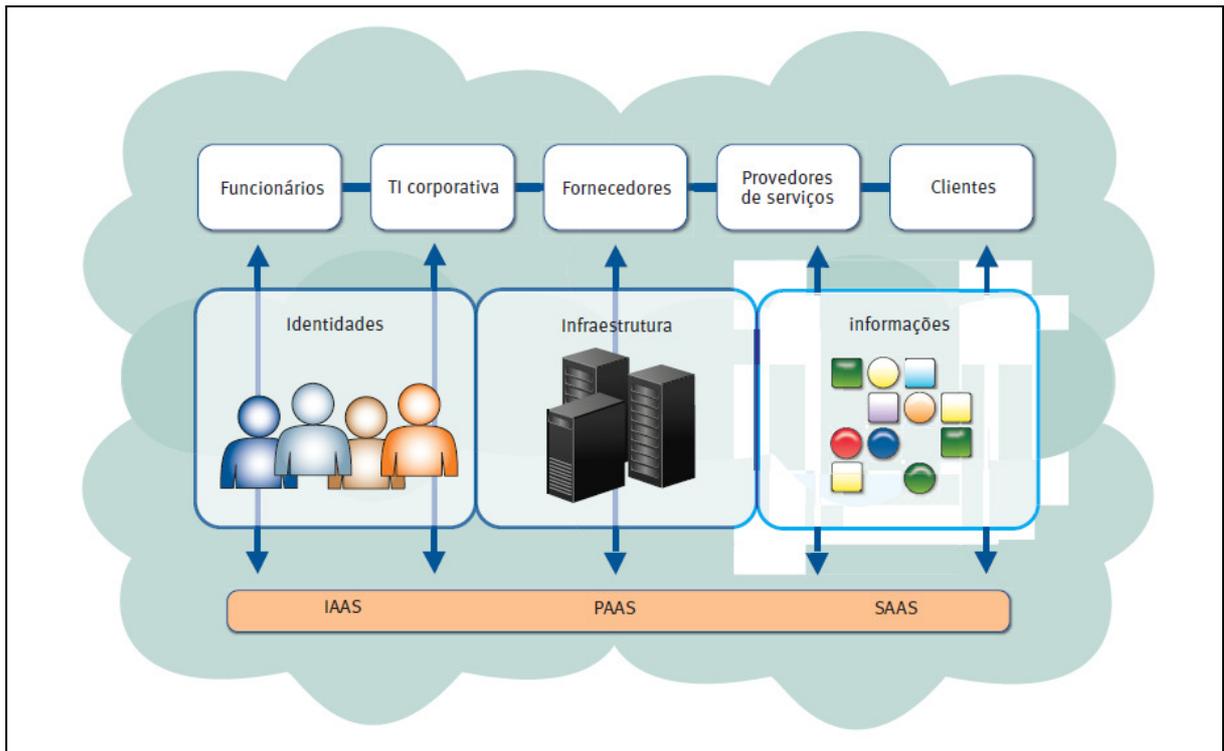


Figura 5 – Desenvolvimento de uma nuvem confiável (RSA, 2009).

Myerson (2011) reforça que “as políticas de segurança dos serviços em nuvem focam em diferentes aspectos da segurança em nuvem, dependendo do sistema de modelo de entrega ao qual se faz referência — SaaS, PaaS ou IaaS:”

- No modelo SaaS o foco da segurança está no gerenciamento de acesso, redução de riscos contra ataques de recursos maliciosos;
- No modelo PaaS tem foco na proteção dos dados, assim como gerenciamento de acesso aos aplicativos.
- No modelo IaaS a política de segurança esta focada no gerenciamento das máquinas virtuais, na proteção dos dados e no gerenciamento de acesso.

“A segurança do serviço em nuvem pode ser ameaçada por: um *hypervisor* falho, carência de políticas de limitação, excesso no equilíbrio de carga e criptografia insegura”, destacou Myerson (2011).

O *hypervisor* é uma camada subjacente que gerencia o ambiente das máquinas virtuais através de um único hardware, então os serviços em nuvem utilizam desses

recursos para serem executados. Quando acontece alguma falha nesse *hypervisor* todas as camadas acima falham.

O cliente deve avaliar as políticas de segurança do provedor de nuvem, além de certificar-se que as políticas de limitações de solicitações de dados, usuários e recursos estão presentes na política. A política de limitação de recursos é utilizada para monitoramento dos recursos da nuvem.

O equilíbrio de carga é utilizado para distribuir os recursos, de modo que caso ocorra alguma falha, outros recursos são disponibilizados de forma automática, evitando assim problemas. Quando ocorre um excesso desse equilíbrio, os recursos falhos podem ser disponibilizados às aplicações, ocorrendo problemas.

A criptografia é utilizada para garantir a confidencialidade e integridade dos dados. Todos os dados que são transportados e manipulados na nuvem devem ser criptografados.

Myerson (2011) conclui que:

A redução de riscos em serviços em nuvens ao mesmo tempo que se mantém um alto tempo de disponibilidade requer um planejamento de riscos proativo para resolver os problemas de quais ativos identificar para cada tipo de nuvem, quais riscos analisar, quais medidas defensivas têm um bom custo-benefício e o que avaliar depois que os riscos forem reduzidos.

Para exemplificar os conceitos abordados até aqui, o Capítulo 4 abordará exemplos de ferramentas que já operam na nuvem.

4. FERRAMENTAS QUE OPERAM NA NUVEM

Existem várias ferramentas que operam na nuvem; deve-se separar pelo modelo de serviço, ou seja, em IaaS, PaaS e SaaS. Muitos fornecedores atendem todos os modelos.

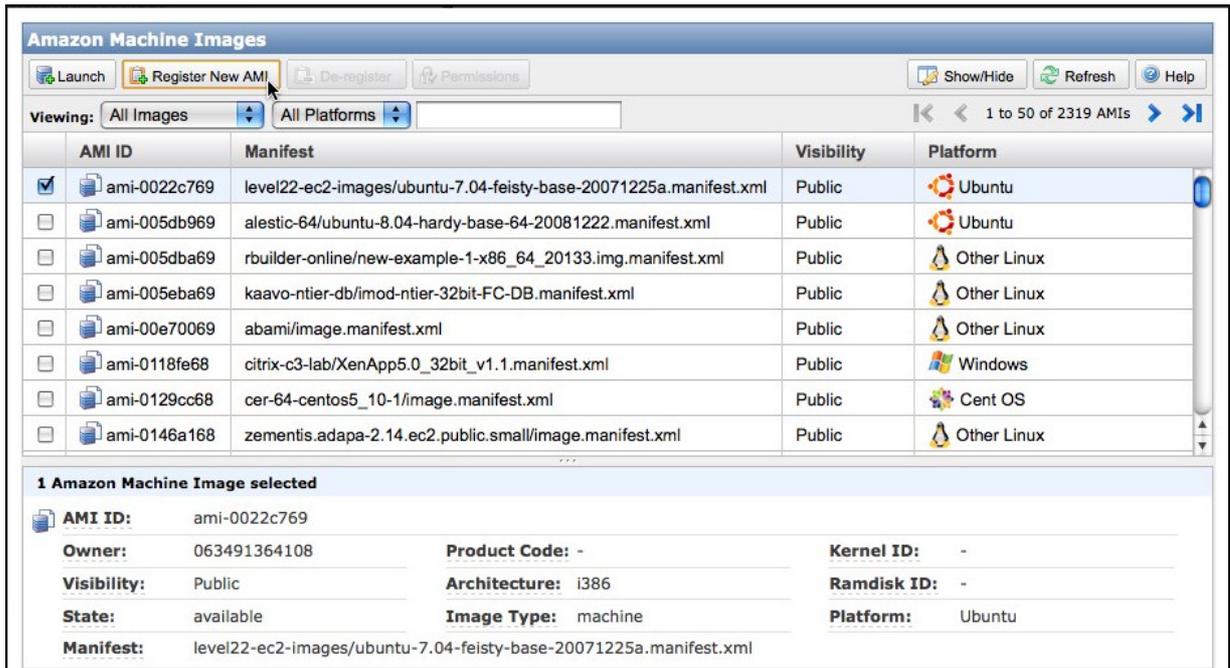
4.1 EXEMPLOS DE IAAS

4.1.1 Amazon Web Services

Os serviços de IaaS da Amazon Web Services foi o primeiro caso de sucesso a se tornar amplamente disponível no mercado. Fornece um conjunto de serviços acessados pela web, a partir de uma infra-estrutura pronta para o uso.

No serviço de IaaS da Amazon o cliente escolhe o sistema operacional, processador, memória do servidor. É um serviço de *data center*, com vantagens em escalabilidade, agilidade e com pagamento proporcional ao uso.

Na figura 6, mostra como é o ambiente de gerenciamento da ferramenta da *Amazon Web Services*.



The screenshot shows the Amazon Machine Images (AMI) console. At the top, there are buttons for 'Launch', 'Register New AMI', 'De-register', and 'Permissions'. Below these are 'Show/Hide', 'Refresh', and 'Help' buttons. The 'Viewing:' section shows 'All Images' and 'All Platforms'. A table lists several AMIs with columns for AMI ID, Manifest, Visibility, and Platform. The first AMI, ami-0022c769, is selected. Below the table, a summary section titled '1 Amazon Machine Image selected' provides details for the selected AMI.

AMI ID	Manifest	Visibility	Platform
<input checked="" type="checkbox"/> ami-0022c769	level22-ec2-images/ubuntu-7.04-feisty-base-20071225a.manifest.xml	Public	Ubuntu
<input type="checkbox"/> ami-005db969	alestic-64/ubuntu-8.04-hardy-base-64-20081222.manifest.xml	Public	Ubuntu
<input type="checkbox"/> ami-005dba69	rbuilder-online/new-example-1-x86_64_20133.img.manifest.xml	Public	Other Linux
<input type="checkbox"/> ami-005eba69	kaavo-ntier-db/imod-ntier-32bit-FC-DB.manifest.xml	Public	Other Linux
<input type="checkbox"/> ami-00e70069	abami/image.manifest.xml	Public	Other Linux
<input type="checkbox"/> ami-0118fe68	citrix-c3-lab/XenApp5.0_32bit_v1.1.manifest.xml	Public	Windows
<input type="checkbox"/> ami-0129cc68	cer-64-centos5_10-1/image.manifest.xml	Public	Cent OS
<input type="checkbox"/> ami-0146a168	zementis.adapa-2.14.ec2.public.small/image.manifest.xml	Public	Other Linux

1 Amazon Machine Image selected			
AMI ID:	ami-0022c769	Product Code:	-
Owner:	063491364108	Architecture:	i386
Visibility:	Public	Ramdisk ID:	-
State:	available	Image Type:	machine
Manifest:	level22-ec2-images/ubuntu-7.04-feisty-base-20071225a.manifest.xml	Platform:	Ubuntu

Figura 6 – Gerenciamento das Maquinas Virtuais (<http://aws.amazon.com/>).

A solução da *Amazon Web Services* fornece elementos comuns necessários para a infra-estrutura de aplicações web, são eles:

- Armazenamento: Através do produto *Amazon Simple Storage Service* (S3), que tem como vantagem um armazenamento escalável, confiável, altamente disponível e baixo custo.
- Computação: Através do produto *Amazon Elastic Compute Cloud* (EC2), que caracteriza na sua oferta de Platform-as-a-Service, permitindo aumentar ou diminuir recursos conforme a demanda.
- Sistema de Mensagens: Através do produto *Amazon Queue Service* (SQS), um sistema de mensagens ilimitado para componentes de aplicativos.
- Armazenamento de Dados: Através do produto *Amazon SimpleDB* (SDB) fornece uma base de dados escalável, indexado em conjunto com processamento e enfileiramento para os conjuntos de dados.

A vantagem da utilização de toda a infra-estrutura da Amazon é que torna-se possível combinar os serviços, pois são desenvolvidos para trabalhar em conjunto, permitindo assim uma comunicação rápida entre os serviços.

4.1.2 Eucalyptus

A *Elastic Utility Computing Architecture for Linking Your Programs To Useful System* (Eucalyptus) é uma solução *open source* para implementação de computação em nuvem. Inicialmente era um projeto de pesquisa do Departamento de Ciência da Computação da Universidade Santa Barbara da Califórnia e hoje é comercializado pela System Inc.

O Eucalyptus é formado por cinco elementos, são eles:

- Controlador de Nuvem (CLC): componente principal de gerenciamento de toda nuvem, ele envia pedidos aos componentes, coletam e enviam respostas dos componentes aos clientes.
- Controlador de Cluster (CC): componente responsável por gerenciar a rede de instâncias, mantendo as informações sobre os controladores de nó, ele encaminha pedidos para iniciar novas instâncias.
- Controlador de nó (NC): componente de controle do sistema operacional e o hypervisor.
- Walrus (W): componente que gerencia o acesso aos serviços de armazenamentos.
- Controlador de armazenamento (SC): serviço de armazenamento do Eucalyptus que implementa a interface S3 da Amazon. Os dados armazenados são de forma compacta e criptografada.

Um exemplo da utilização do Eucalyptus é o NEBULA da NASA, que o utiliza como base de sua plataforma. Outro exemplo é a empresa farmacêutica Eli Lilly que utiliza o Eucalyptus para armazenamento e processamento sob demanda.

Chaganti (2010) “Eucalyptus é um ambiente de serviços em nuvem bem arquitetado que pode ser usado para se criar uma nuvem pública na infraestrutura do Amazon EC2 ou criar nuvens privadas no seu datacenter”.

O Eucalyptus tem como vantagem o fato de ser um *software* disponibilizado de forma gratuita e de desenvolvimento rápido. Além disso, existe uma grande comunidade de usuários. Pode ser instalado em nuvem pública e privada. Permite portabilidade e flexibilidade, podendo ser utilizado em nuvem híbrida e possui um suporte comercial.

4.2 EXEMPLOS DE PAAS

4.2.1 Google App Engine

A Google App Engine é uma plataforma na nuvem que permite ao desenvolvedor executar aplicativos na infra-estrutura do Google. Atualmente as linguagens Python e Java são suportadas.

Inicialmente pode-se utilizar uma conta gratuita, na qual os aplicativos podem consumir até 500 MB de armazenamento e 5 milhões de exibições de páginas por mês. É possível escalar os aplicativos facilmente, caso o aplicativo necessite de mais recursos podem ser orçados conforme a necessidade do cliente.

O App Engine fornece os seguintes serviços:

- Armazenamento de dados: através da BigTable, possui um sistema de banco de dados de alto desempenho.
- Armazenamento em cache: através do Memcached, um sistema de armazenamento de cache de alto desempenho.
- Autenticação: utilizam-se as contas do Google.
- Correio eletrônico: utiliza-se o Gmail.

Existem algumas restrições impostas para utilização da App Engine, são elas:

- Somente pode utilizar as bibliotecas disponíveis Python e Java.
- Número restrito de solicitações da CPU, memória, tamanho de arquivos.
- Solicitações demoram 30 segundos para ser respondidas.
- Não há acesso ao sistema de arquivos e estes somente podem ser lidos de forma estática.
- O armazenamento de dados somente pela BigTable.
- O código só pode ser executado a partir de uma solicitação HTTP.

Apesar de ser uma plataforma muito limitada, a utilização da App Engine é uma ótima maneira para a criação e execução de aplicativos Web de forma escalável, além de ter todo o suporte e segurança do Google.

4.2.2 Windows Azure

É uma plataforma baseada em serviços na nuvem, hospedada pela Microsoft. Esta plataforma tem um sistema operacional próprio, o Azure, e um framework de serviços.

O Windows Azure (Figura 7) está dividido em 4 partes, pelas quais é possível abstrair os benefícios da tecnologia, que são escala, processamento e custos. As partes são descritas abaixo:

- Windows Azure: sistema operacional para executar as aplicações.
- SQL Azure: sistema gerenciador de banco de dados relacional hospedado na nuvem com arquitetura do SQL Server.
- Windows Azure AppFabric: infra-estrutura de serviços para as aplicações.
- Windows Azure Marketplace: ferramenta de disponibilização dos aplicativos dos desenvolvedores a clientes em potencial.

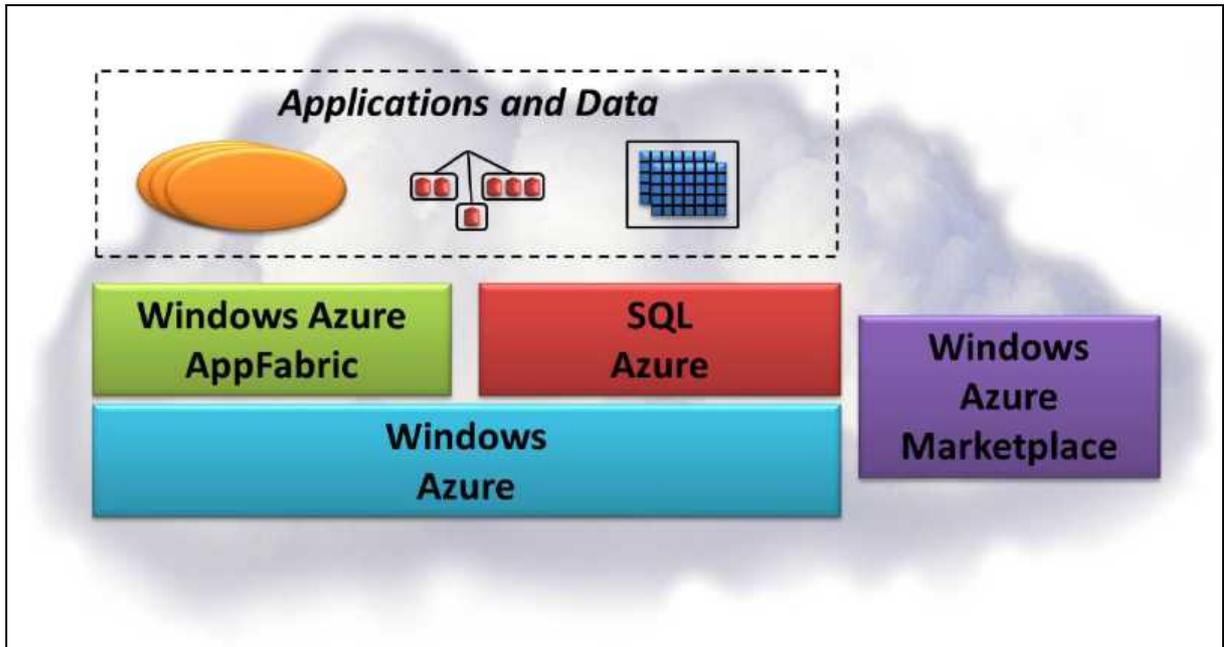


Figura 7 – Serviços disponibilizados pelo Windows Azure. (Chappel, 2010).

Shonnard et. al (2009) salienta que: “A malha do Windows Azure provê duas áreas principais de funcionalidade: computação (por exemplo, execução de um aplicativo) e armazenamento (por exemplo, de dados em disco), os blocos de construção que são a base de todos os aplicativos em nuvem.”

4.3 EXEMPLOS DE SAAS

Como foi apresentado no Capítulo 2, o serviço pode ser dividido em duas categorias: serviço orientado ao cliente e serviço orientado ao negócio. Essa divisão também pode ser chamada de *software* horizontal e *software* vertical respectivamente.

4.3.1 Google Doc's

O Google Doc's (Figura 8) é uma solução SaaS desenvolvida pela empresa Google, que fornece esse *software* na categoria de *software* horizontal, no qual seus usuários tem a possibilidade de criar arquivos de textos, planilhas, apresentações e entre outros.

Por ser um *software* horizontal, é oferecido a todos os públicos sem custo e é financiado por anúncios que são disponibilizados aos seus usuários. Este tipo de *software* agrega muitos benefícios para pequenas empresas que não precisam adquirir licenças de uso para os *softwares* oferecidos pelo Google Doc's.

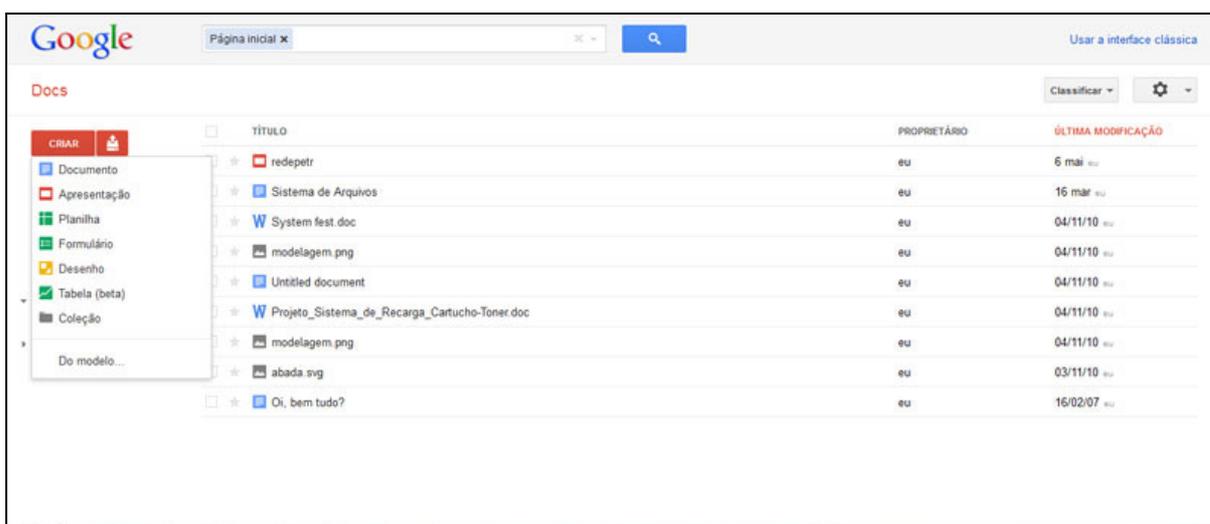


Figura 8 – Google Doc's. (<http://docs.google.com>).

A grande vantagem na utilização deste *software* é que todos os arquivos criados são armazenados pelo *software*, podendo assim o usuário acessá-lo de qualquer lugar. Outra característica é a segurança fornecida e um *software* livre de falhas. O backup é garantido pelo fornecedor, e os usuários podem permitir a visualização de seus documentos por outros usuários.

4.3.2 Social Enterprise

O Social Enterprise é uma ferramenta nas nuvens da Salesforce. Essa ferramenta é classificada como *software* vertical, pois é voltada para linha negócios e contém conhecimento específico sobre relacionamento com o cliente.

Ele engloba diversas outras ferramentas que podem ser incorporadas ou não à plataforma Social Enterprise. Dentre esses serviços destacam-se:

- Sales Cloud: aplicativo de venda, permitindo o acesso às informações em tempo real.
- Service Cloud: aplicativo de atendimento ao cliente, tendo acesso aos recursos necessários para atendimento ao cliente e através de redes sociais.
- Chatter: aplicativo de rede social corporativa, auxiliando na colaboração entre equipes e compartilhamento de informações.

A Salesforce vem quebrando barreiras com suas soluções, hoje são mais 100 mil usuários, demonstrando uma grande aceitação do modelo de SaaS no meio corporativo.

O grande diferencial da solução é a integração com ferramentas externas e até integração com outras nuvens, podendo integrar com o Google Apps, Facebook e aplicativos EC2 e S3 da Amazon.

Como resultado desta pesquisa e para que o objetivo geral possa ser realizado, o capítulo 5 apresentará um estudo de caso de implantação de um *software* CRM em um nuvem privada.

5. ESTUDO DE CASO

Conforme foi descrito no objetivo deste trabalho, este capítulo descreverá o processo de implantação de um *software* CRM em nuvem privada já implantada na empresa, tendo como foco a visão do departamento de TI.

5.1 CRM

Customer Relationship Management é definido não somente como um *software*, mais também como estratégia de negócio voltada para auxiliar as organizações a fim de melhorar sua lucratividade, vendas e satisfação com clientes. As empresas devem organizar seu comportamento perante o cliente, através da adoção de processos e tecnologias de seus canais de atendimento.

Um CRM compõe as seguintes operações:

- Suporte a vendas: através de integração com canal de vendas, pedidos e oportunidades.
- Suporte a serviços de pós venda: através de serviços de atendimento a clientes.
- Suporte a *marketing*: através do gerenciamento de campanhas, segmentação e posicionamento de clientes, avaliação e dados detalhados de clientes.

Segundo BRETZKE(2000) o CRM é focado na estratégia de negócio, buscando ações voltadas para satisfação e retenção de clientes, suportada pela tecnologia da informação. O objetivo é prover meios para que a empresa possa atender, reconhecer e cuidar dos clientes, permitindo que esses dados sejam compartilhados por toda organização, a fim de que esta seja de conhecimento de todos e não somente dos atendentes centrais que receberam a informação.

A implantação de um CRM pode destacar alguns benefícios, são eles:

- Conhecimento sobre os clientes através da função de registrar, manter e tratar os dados dos clientes e recuperá-los de forma rápida, mantendo um relacionamento exclusivo e diferenciado.
- Melhorar os processos mercadológicos e empresariais através da integração com os departamentos da organização.
- Produtos ou serviços para cliente, diferenciando os produtos e serviços do concorrente.
- Comunicação com clientes atuais e futuros, mantendo uma comunicação periódica simples e personalizada para cada cliente.
- Diminuindo o custo de venda e pós venda, aumentando a eficácia no primeiro atendimento a fim de conquistar novos clientes e auxiliando a fidelização de clientes através de novas vendas.
- Agregação de valor para o cliente, auxiliando o desenvolvimento de novos produtos ou serviços para a empresa, a partir do conhecimento acumulado das necessidades do cliente.

Um CRM tem 3 componentes básicos em sua arquitetura, são:

- CRM operacional é uma aplicação de automatização dos canais de relacionamento com clientes, através do *Call Center*, *Help Desk* e Vendas.
- CRM analítico é uma aplicação de inteligência dos processos, tem como objetivo a identificação de cada tipo de cliente.
- CRM colaborativo é uma aplicação que permite a automação e integração entre os contatos da empresa com o cliente, através dos meios de comunicação, como email, telefone e mala direta.

O sucesso de implantação de sistema de CRM envolve tecnologia, processo e estratégia. A não adoção de um modelo estratégico de relacionamento duradouro com cliente e a falta de uma reavaliação dos processos, tornará a implantação apenas em uma informatização dos departamentos de venda e atendimento.

Recomenda-se uma estratégia de 4 etapas na implantação de um CRM:

- Definição e planejamento do modelo de relacionamento: defini-se o modelo de atendimento, como ele será tratado, plano de comunicação, coerência com o posicionamento e plano de *marketing*.
- Redesenho dos processos de atendimento: atividades que visam o atendimento do cliente.
- Seleção da solução: definição do *software* a ser utilizado e do *hardware* necessário.
- Implantação da tecnologia de CRM: superação dos obstáculos como resistência as mudanças de culturas, aprendizado lento e esquecimento rápido.

5.2 A EMPRESA

A empresa selecionada atua na prestação de Serviço de Comunicação Multimídia (SCM), através da venda de planos de internet via rádio. Atualmente sua área de atuação esta na cidade de Maracaí-SP.

Líder de mercado no seu segmento, a empresa tem em sua carteira de cliente aproximadamente 950 clientes. Atuando desde 2006, a empresa teve um crescimento acima do esperado nos últimos anos.

Para se manter líder, a empresa optou por uma filosofia de trabalho a fim de manter a alta qualidade nos serviços de atendimento, buscando respostas rápidas aos pedidos e reclamações dos clientes através de um atendimento exclusivo.

De pequeno porte, com 9 colaboradores, sendo 4 atendentes internos, 3 atendentes técnicos externo, 1 diretor técnico e 1 proprietário. Todos têm contato direto com os clientes, e a troca de informações, com aproximadamente 60 atendimentos por dia, era prejudicada sem o auxílio de um *software* de relacionamento com o cliente.

A empresa definiu como estratégia “ter um atendimento diferenciado e de alta qualidade, simples e inovador”.

5.3 SOLUÇÃO

A escolha de uma solução passou por um processo de análise entre o setor administrativo e o departamento de TI. A seleção obedeceu aos seguintes requisitos:

- ser um *software* open source;
- flexibilidade para integração com outros sistemas;
- fornecer ferramentas compatíveis com negócio da empresa como venda, suporte e estoque;
- possuir idioma em português do Brasil;
- apresentar forte adoção por outras organizações e possuir suporte especializado;
- permitir personalização do *software*;

5.3.1 vTiger CRM

O vTiger CRM é uma solução open source desenvolvida por uma grande comunidade mundial. O objetivo do projeto vTiger CRM é fornecer uma solução CRM de custo baixo para pequenas e médias empresas, através do open source.

O vTiger CRM é utilizado mundialmente, com mais de mil downloads por dia e mais de 2 milhões de downloads até outubro de 2011. Possui diversos desenvolvedores e usuários colaboradores ativos em diversos países. Está disponível em diversos idiomas.

O projeto open source é gerenciado pela Vtiger, uma empresa privada, com escritórios na Índia e EUA. vTiger permite recursos de personalização para atender aos negócios de cada empresa.

Na figura 9, ilustra a interface do usuário do *software*.

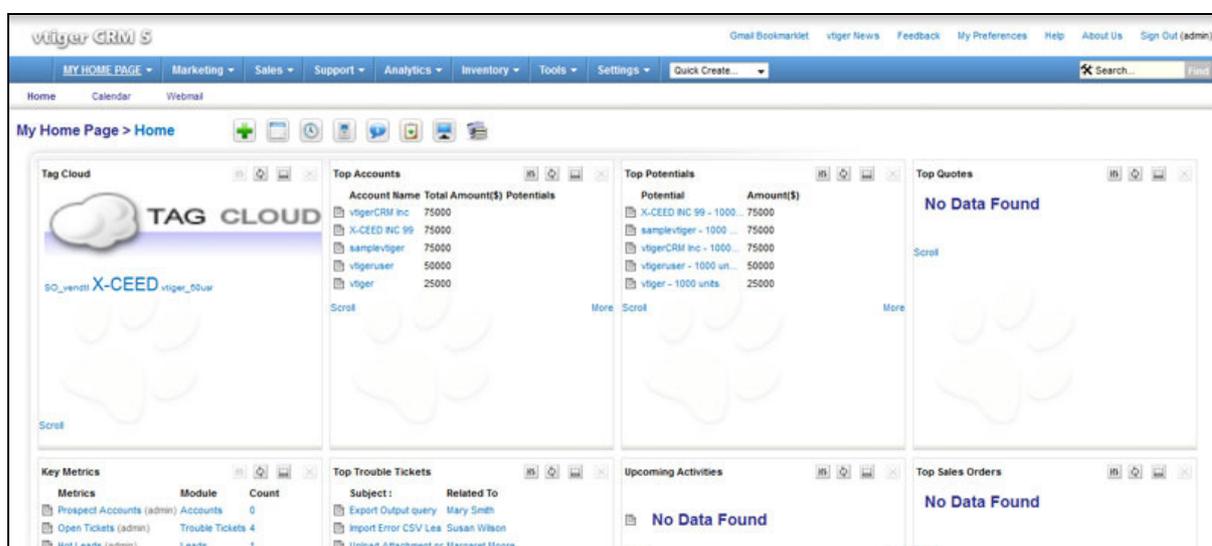


Figura 9 – vTiger CRM (<http://demo.vtiger.com/>).

O *software* possui várias funcionalidades como:

- Automatização das vendas: gestão de *leads*, contas, contatos, oportunidade e cotações de vendas.

- Suporte e serviços ao cliente: gestão *tickets*, base de conhecimento e auto-atendimento de clientes.
- Automatização de *Marketing*: gestão de campanhas e envio de email em massa.
- Gestão de estoque: catálogo de produtos, lista de preços, gestão de vendas, ordens de encomendas, ordens de vendas e faturas.
- Gestão de atividades: gestão de reuniões, eventos, calendário compartilhado, histórico de atividades e notificações por email.
- Relatórios: relatórios configuráveis e métricas chaves.
- Customização do produto: permite a personalização de campos, listas e módulos.
- Gestão de segurança: gerenciamento de perfis, regras e controle de acesso.

As principais vantagens na utilização do vTiger CRM são:

- *Software open source* sem a necessidade de licenças de uso.
- Interface amigável e de fácil utilização.
- Facilidade de instalação.
- Integração com outros aplicativos como: Microsoft Outlook®, Mozilla/Thunderbird, Microsoft Office®.
- Portal de auto-atendimento para cliente.

5.4 IMPLANTAÇÃO NA EMPRESA

5.4.1 Instalação da solução

A empresa selecionada tem como seu ramo de atividade a tecnologia da informação. Este fator ajudou na instalação da solução, pois seus servidores já estavam em um ambiente de nuvem.

O primeiro passo para a instalação foi a criação de uma máquina virtual com as configurações que suportam a instalação da solução. A solução exige as seguintes configurações:

- Processador: 1.8 GHz;
- Memória: 2 Gb;
- Espaço em disco: 10 Gb;
- Sistema Operacional: Debian 4.0/5.0, CentOs 4.2, Fedora Core 6/9.

A máquina virtual que abrigará o *software* vTiger CRM utilizou-se as seguintes configurações:

- Processador: 2 núcleos;
- Memória: 2 Gb;
- Espaço em disco: 20 Gb;
- Sistema Operacional: Debian 5.0.

Com o auxílio da virtualização e de *templates* pré-instalados dentro da nuvem privada, o provisionamento da VM que atenderá o servidor dedicado para a solução

durou cerca de 10 minutos. Todo este trabalho foi executado pelo diretor técnico, através de uma interface web de gerenciamento da nuvem.

Após a disponibilização da VM, o passo seguinte foi a instalação do *software* vTiger CRM. Para que o *software* possa ser executado, os seguintes pacotes devem estar instalados e configurados adequadamente:

- Servidor Web: Apache 2.0.40 ou superior;
- Banco de dados: MySql 4.1.x ou superior;
- Php: Php 5.0.x.

A versão do vTiger CRM escolhida para instalação foi 5.2.1, última versão estável. O procedimento de instalação é semelhante à instalação de um *software* desktop simples, só que em um ambiente web.

5.5 TESTES

Os testes de carga e desempenho são utilizados para medir os projetos web que buscam melhorar seus resultados.

Para a nossa solução foi utilizado o seguinte teste:

- Testes de estresse: identificar o desempenho da aplicação durante níveis altos de acesso.

Para a simulação foi assumida uma conexão de 10 usuários acessando o sistema simultaneamente e o intervalo de tempo entre os cliques de 20 segundos, durante 1 minuto. A ferramenta utilizada foi Webserver Stress: 7.2.2.261 - Versão de Avaliação.

Os testes foram realizados antes da disponibilização do serviço para o setor administrativo, pois a garantia do sistema devia ser consolidada e, após entrar em operação, os níveis de interrupção deveriam ser o mínimo possível.

Para o teste, o programa foi acessado através de URL`s, com características diferentes, passagem de parâmetro e utilização. Cada uma delas está detalhada na Tabela 1:

Nº URL	URL	Parâmetros	Username	Password
1	http://cloud.izaz.com.br/vtigercrm			
2	http://cloud.izaz.com.br/vtigercrm/index.php	module=Users&action=Authenticate&return_module=Users&return_action=Login&user_name=admin&user_password=admin&login_theme=softed&login_language=pt_br&Login.x=35&Login.y=13	admin	admin
3	http://cloud.izaz.com.br/vtigercrm/index.php?module=Calendar&action=index&parenttab=My%20Home%20Page			
4	http://cloud.izaz.com.br/vtigercrm/index.php?module=Products&action=index&parenttab=Inventory			

Tabela 1 – Descrição das URL`s para os testes.

Os primeiros resultados são referentes ao tempo de resposta do clique por URL`s. A figura 9 apresenta os resultados obtidos.

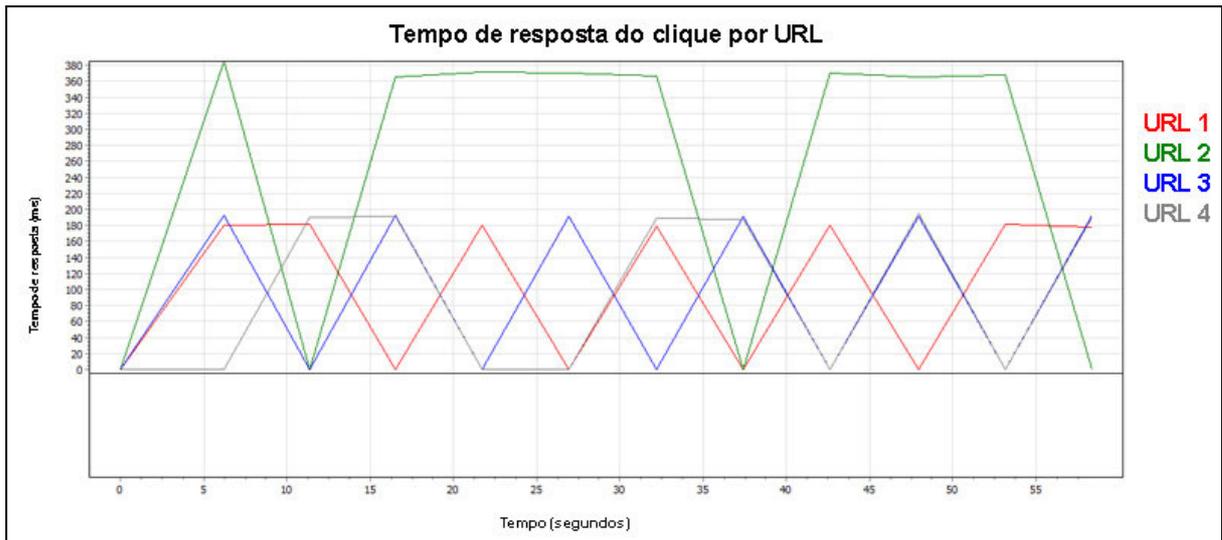


Figura 9 – Resultados do tempo de resta do clique por URL.

Analisando o gráfico da figura 9, o tempo de resposta na URL 2 que utiliza-se parâmetros é superior aos demais, neste caso são parâmetros utilizados para o *login* do sistema. As demais URL`s são apenas direcionamento para funções do *software* que mantém uma média de tempo de resposta.

Nenhumas das URL`s apresentou erros de acesso, como pode ser analisado na Tabela 2.

URL No.	Erros	Erros [%]
1	0	0
2	0	0
3	0	0
4	0	0

Tabela 2 – Erros de acesso a URL.

Outro teste analisado foi o número de requisições abertas e o tráfego utilizado no envio e recebimento de requisições por segundo, em um tempo de 1 minuto. O gráfico da figura 10 apresenta os resultados obtidos.

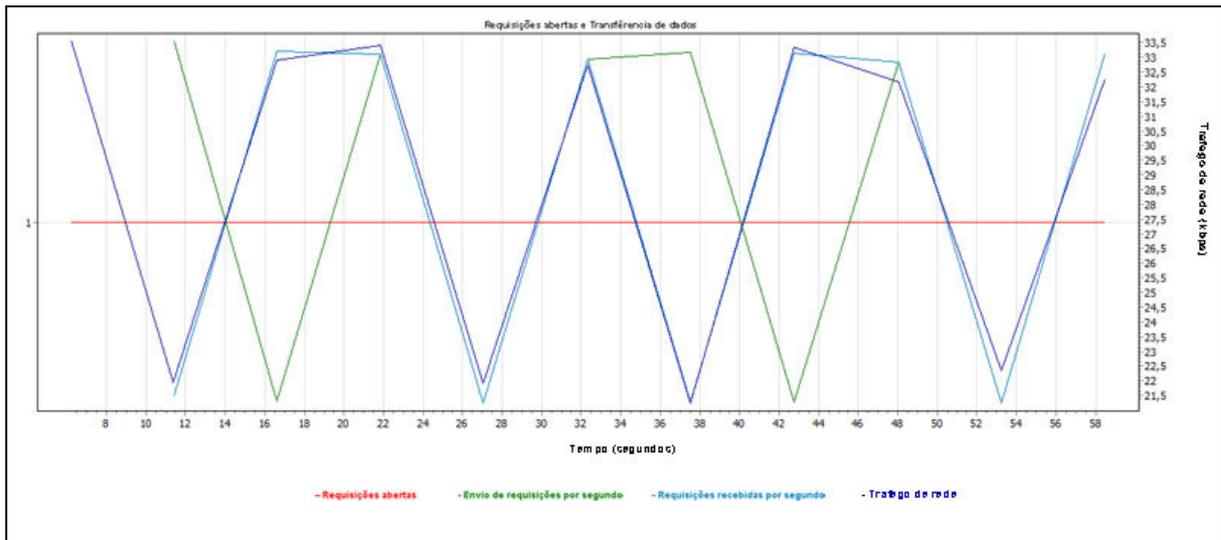


Figura 10 – Gráfico de requisições abertas e transferência de dados.

Pode-se perceber que para o teste com 10 usuários simultâneos o tráfego de rede se manteve estável e atingiu no máximo de 33,5 kbps com o decorrer do tempo proposto para o teste. A rede atual da empresa trabalha no padrão FastEthernet, ou seja, o limite de transferência é de 100 Mbps.

5.6 AVALIAÇÃO DOS RESULTADOS

As informações provenientes dos testes foram satisfatórias para o projeto de implantação do CRM dentro da nuvem privada da empresa. O grande resultado obtido foi que o *software* entrou em operação em apenas 1 dia, pois a utilização dos recursos já disponíveis ajudou no sucesso do projeto.

Para realização desta implantação não houve agregação de custos as empresas, pois não houve a necessidade de aquisição de licença de uso para o programa, por se tratar de um *software open source*. Também não foi necessário adquirir recursos de *hardware* extras, como aquisição de servidores, pois os recursos utilizados já estavam disponíveis para utilização.

A segurança e a escalabilidade do *software* tornaram o projeto mais confiável. O fato de a empresa ter o domínio dos recursos tornou o projeto escalável, dependendo da demanda necessária, com simples configurações os recursos podem ser aumentados ou diminuídos. O desempenho dentro da empresa foi excelente, os padrões de qualidade dos recursos de rede adotados auxiliaram no desempenho da aplicação.

Por se tratar de um programa de uso específico desta empresa, no nível de maturação proposto para o *software* como serviço, esta solução manteve no primeiro nível, ou seja, no nível ad-hoc/personalizado, apenas separado logicamente de outras aplicações e acessado como um serviço.

CONSIDERAÇÕES FINAIS

Atualmente o cenário da Tecnologia da Informação vem sofrendo mudanças; novos desafios, modelos e arquitetura surgem. Melhorias quanto à utilização de recursos computacionais são destacados. A computação em nuvem permite que esses recursos possam ser adquiridos de forma rápida e por demanda.

É visível que os benefícios da utilização desta nova tecnologia agregam aos negócios resultados positivos, através da redução de custos gerados pela economia de recursos. Os modelos de serviços gerados permitem uma interação entre as aplicações.

Software como serviço é o modelo que apresenta mais benefícios para os usuários. Este *software* pode ser utilizado por outras empresas, através das tecnologias disponíveis.

Por ser uma nova tecnologia como qualquer outra gera novos riscos, a questão da segurança e o receio diante de uma novidade fazem com que a adoção seja lenta. Esta é uma fase de aprendizado, muitos novos desafios surgirão.

Outro fator que afeta esta nova tecnologia são os serviços de banda larga oferecidos no Brasil. A baixa velocidade, se comparada à oferecida em outros países, torna-se um fato inibidor à sua expansão.

O estudo de caso apresentou resultados positivos para implantação de *software* como serviço dentro de uma nuvem privada. A velocidade da implantação surpreendeu, os testes foram os melhores possíveis. O estudo serviu como guia e motivador para implantação desta nova tecnologia.

Desafios como integração e interoperabilidade entre os *Softwares* como Serviços devem ser temas para trabalhos futuros.

REFERÊNCIAS

ANDERSON, Chris. **A Cauda Longa: do mercado de massa para o mercado de nicho**. Trad. Afonso Celso da Cunha Serra. Rio de Janeiro: Elsevier, 2006.

BRETZKE, Miriam. **Marketing de Relacionamento e Competição em Tempo Real**. São Paulo: Atlas, 2000.

CAMBIUCCI, Waldemir. **Uma introdução ao Software + Serviços, SaaS e SOA**. Microsoft Corporation. Disponível em: <<http://msdn.microsoft.com/pt-br/library/dd875466.aspx>>. Acessado em: 03 de abril de 2011.

CHAGANTI, Prabhakar. **Serviços em nuvem para sua infraestrutura virtual, Parte 1: Infrastructure-as-a-Service (IaaS) e Eucalyptus**. DeveloperWorks IBM. Disponível em: <http://www.ibm.com/developerworks/br/library/os-cloud-virtual1/>. Acessado em: 08 de outubro de 2011.

GARCIA, Marco A.B. A. **Definições, funcionamento e aplicações da Computação em Nuvem**. Departamento de Ciências e Estatísticas – IBELC – UNESP. Disponível em: <<http://www.dcce.ibilce.unesp.br/~aleardo/cursos/hpc/Marco.pdf>>. Acessado em: 04 de abril de 2011.

DESISTO Roberto. PLUMMER, Darly, SMITH, David. **A Relação entre Computação em Nuvem e SaaS**. Garther. Disponível em: <<http://info.abril.com.br/corporate/noticias/072008/28072008-1.shtml>>. Acesso em: 20 de junho de 2011.

MELO, Cássio A. ACOVERDE, Daniel F. MORAES, Éfrem R. A. PIMENTEL, João H. C. FREITAS, Rodrigo . **Software com serviço: Um modelo de negócio**

emergente. Centro de Informática - Universidade Federal de Pernambuco (UFPE). Publicado em 2007. <<http://www.cin.ufpe.br/~jhcp/publica/jhcp-saas.pdf> >. Acesso em: 01 de abril de 2011.

MOTA, Valeria Pereira. **Desenvolvimento da modelagem de uma ferramenta para gerenciar aplicações SaaS.** Centro Universitário Luterano de Palmas (ULBRA). Publicado em 2009. Disponível em: <http://www.followscience.com/library_uploads/f2bebad713499abed6ad776cc2985c6e/586/desenvolvimento_da_modelagem_de_uma_ferramenta_para_gerenciar_aplicacoes_saas.pdf>. Acessado em: 29 de março de 2011.

MYERSON, Judith M. **Serviços em nuvem: reduza os riscos, mantenha a disponibilidade.** DeveloperWorks IBM. Disponível em: <<http://www.ibm.com/developerworks/br/cloud/library/cl-cloudservicerisks/index.html>>. Acessado em: 07 de outubro de 2011.

SKONNARD, Aaron. BROWN, Keith. **Introdução à plataforma Windows Azure AppFabric para desenvolvedores.** Whitter Papers Windows Azure. Disponível em: <<http://www.microsoft.com/windowsazure/pt/br/whitepapers/AppFabricForDevelopers> >. Acessado em: 08 de outubro de 2011.

SOUSA, Flávio R. C., MOREIRA, Leonardo O., MACHADO, Javam C. **Computação em nuvem: Conceitos, Tecnologias, Aplicações e Desafios.** Publicado em 2009. Disponível em: <http://www.es.ufc.br/~flavio/files/Computacao_Nuvem.pdf>. Acessado em: 02 de abril de 2011.

TAURION, Cezar. **Cloud Computing - computação em nuvem: transformando o mundo da tecnologia da informação.** Rio de Janeiro: Brasport, 2009.