



LIGA DE ENSINO DO RIO GRANDE DO NORTE
CENTRO UNIVERSITÁRIO DO RIO GRANDE DO NORTE
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

VITOR EDUARDO SARAIVA COELHO

IMPLANTAÇÃO DE SAAS (SOFTWARE AS A SERVICE) ESCALÁVEL E DE
ALTA DISPONIBILIDADE USANDO WINDOWS AZURE

NATAL/RN

2014

VITOR EDUARDO SARAIVA COELHO

**IMPLANTAÇÃO DE SAAS (SOFTWARE AS A SERVICE) ESCALÁVEL E DE
ALTA DISPONIBILIDADE USANDO WINDOWS AZURE**

Trabalho de conclusão de curso apresentado
ao Centro Universitário do RN - UNIRN
como requisito final para obtenção do título
de bacharel em sistemas de informação.

Orientador: Prof^o Alexandre Damasceno

NATAL/RN

2014

Catálogo na Publicação – Biblioteca do UNI-RN
Setor de Processos Técnicos

Coelho, Vitor Eduardo Saraiva.

Implantação de SAAS (SOFTWARE AS A SERVICE) escalável e de alta disponibilidade usando Windows Azure / Vitor Eduardo Saraiva Coelho. Natal, 2014.

33 f.

Orientador: Profº Alexandre Damasceno.

Monografia (Graduação em Sistemas de Informação) – Centro
Universitário do Rio Grande do Norte.

1. Cloud Computing – Monografia. 2. Computação nas nuvens – Monografia. 3. SaaS – Monografia. 4. Escalabilidade – Monografia. I. Pinheiro, Joseane Alves. II. Título.

RN/UNI-RN/BC CDU 004.92

VITOR EDUARDO SARAIVA COELHO

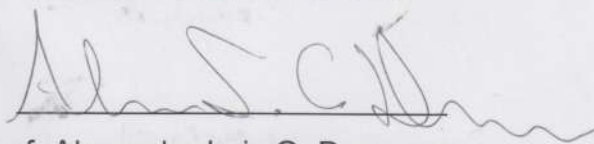
**IMPLANTAÇÃO DE SAAS(SOFTWARE AS A SERVICE) ESCALÁVEL E DE ALTA
DISPONIBILIDADE USANDO WINDOWS AZURE**

Trabalho de conclusão de curso
apresentado ao Centro
Universitário do RN - UNIRN
como requisito final para
obtenção do título de bacharel
em sistemas de informação.

Orientador: Profº Alexandre Luiz
G. Damasceno

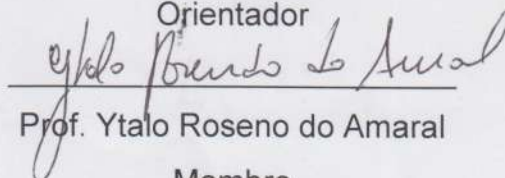
Aprovado em 1 / 1 /

BANCA EXAMINADORA



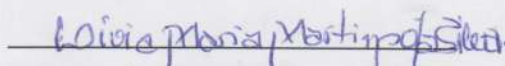
Prof. Alexandre Luiz G. Damasceno

Orientador



Prof. Ytalo Roseno do Amaral

Membro



Prof. Livia Maria Martins da Silva

Membro

A minha família que me incentivou e me proporcionou todo suporte pra superar os obstáculos encontrados.

AGRADECIMENTOS

Agradeço primeiramente a Deus que me deu forças para enfrentar as dificuldades da vida.

Agradeço a minha querida esposa que me incentivou a concluir o curso.

Agradeço a todos os professores da UNIRN que me ajudarão a ser um profissional melhor a cada dia.

"Uma mente que se abre a uma nova
ideia jamais voltará ao seu tamanho
original".

(Albert Einstein)

RESUMO

Este trabalho tem o objetivo de mostrar os principais conceitos de cloud computing e como realizar a implantação de um software no data center Windows Azure, levando em consideração temas como escalabilidade e alta disponibilidade, como também os cuidados que devem ser levados em consideração no desenvolvimento da aplicação que será implantada no ambiente de nuvem.

Palavras-Chave: Cloud Computing. Computação nas nuvens. SaaS. Escalabilidade.

ABSTRACT

This work aims to show the main concepts of cloud computing and how to perform the deployment of software on the Windows Azure data center, taking into account issues such as scalability and high availability, but also the care that must be taken into consideration in the development of application that will be deployed in the cloud environment.

Keywords: Summary. Intended. Work. Subject.

LISTA DE FIGURAS

Figura 1 – Crescimento-Rápido	13
Figura 2 – Liga-Desliga.....	13
Figura 3 – Picos imprevisíveis	14
Figura 4 – Picos previsíveis	14
Figura 5 – Modelos de serviço na nuvem	17
Figura 6 – Escalabilidade Horizontal	21
Figura 7 – Escalabilidade Vertical	22
Figura 8 – Modelo de CDN disponibilizado pelo Windows Azure	26
Figura 9 – Tela de cadastro Windows Azure	28
Figura 10 – Botão sites do Windows Azure	28
Figura 11 – Botão novo do Windows Azure.....	28
Figura 12 – Tela de criação de serviços de site	29
Figura 13 – Configurando implantação.....	29
Figura 14 – Definindo repositório.....	30
Figura 15 – Repositório GIT	30
Figura 16 – Publicando aplicação pelo git	31

SUMÁRIO

1 INTRODUÇÃO	11
1.1 OBJETIVOS	11
2 FUNDAMENTAÇÃO TEÓRICA	12
2.1 O QUE É CLOUD COMPUTING?	12
2.1.1 Crescimento rápido	12
2.1.2 Liga Desliga	13
2.1.3 Picos imprevisíveis	14
2.1.4 Picos Previsíveis	14
2.2 TIPOS DE SERVIÇOS	14
2.2.1 IaaS	15
2.2.2 PaaS	15
2.2.3 SaaS	16
2.3 VANTAGENS E DESVANTAGENS	17
2.3.1 Vantagens	17
2.3.2 Desvantagens	20
2.4 O QUE É ESCALABILIDADE	20
2.4.1 Escalabilidade horizontal	21
2.4.2 Escalabilidade vertical	21
2.5 GIT	22
2.6 GitHub	22
3 DESENVOLVIMENTO	23
3.1 WINDOWS AZURE	23
3.2 Modelos de roles(Papéis) no Windows Azure	23
3.3 TIPOS DO AZURE STORAGE	24
3.3.1 Table	24
3.3.2 Blob	25
3.3.3 Queues	25
3.4 SQL AZURE	25
3.5 CDN – REDE DE DISTRIBUIÇÃO DE CONTEÚDO	25
3.6 WINDOWS AZURE CACHE	26
3.7 FAZENDO DEPLOY DE UMA APLICAÇÃO PARA O WINDOWS AZURE	27

3.7.1 Realizando o cadastro no Windows Azure	27
3.7.2 Cadastrando um serviço de site no Windows AZure	28
3.7.3 Integrando o GIT com Windows AZure	29
3.7.4 Publicando uma aplicação no Azure com auxílio do GIT	30
4 CONCLUSÃO	32
REFERÊNCIAS	33

1 INTRODUÇÃO

Esse trabalho tem o foco em mostrar os principais conceitos da computação nas nuvens, quais as suas características, principais vantagens e desvantagens de ter uma aplicação hospedada em um ambiente de nuvem. Este trabalho também visa mostrar as principais alterações que uma aplicação deve implementar para que possa usufruir dos recursos que o ambiente de nuvem oferece, e como o desenvolvedor pode fazer uso desses benefícios. O ambiente de nuvem que vamos trabalhar é o Windows Azure, e vamos fazer integração do GIT no ambiente do Windows Azure, para realizar a implantação de um sistema exemplo, desenvolvido em C# usando o framework web .net MVC 4.

1.1 OBJETIVOS

Mostrar como deve ser feito a implantação de uma aplicação desenvolvida na linguagem C# utilizando o framework .net MVC 4 para o desenvolvimento de aplicações web no ambiente de nuvem Windows Azure, usando como ferramenta o GIT para realizar a implantação da aplicação. Este trabalho não vai mostrar como foi feito o desenvolvimento da aplicação, mas somente como é feito a implantação da aplicação para o ambiente de nuvem e quais as alterações devem ser feitas em uma aplicação que pretende ser implantada no ambiente de nuvem da Microsoft, Windows Azure.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 O QUE É CLOUD COMPUTING?

Cloud computing ou computação nas nuvens, é um termo atualmente usado para definir um novo paradigma na forma como os softwares são construídos, utilizados e comercializados, é de fato uma nova forma de se fazer software. Escalabilidade, alta disponibilidade e redução nos custos são algumas das características de um software implantado em um ambiente nas nuvens. Podemos imaginar uma aplicação que em um determinado período necessite aumentar a capacidade de processamento do servidor onde está hospedada, para atender uma demanda significativa de seus usuários cadastrados, mas que essa demanda dure apenas algumas horas de um determinado dia. No modelo tradicional de desenvolvimento de software (On premise), teremos que solicitar um hardware de maior capacidade de processamento ao fornecedor de hardware para que a demanda seja atendida. Mas, e depois disso o que faremos com toda essa capacidade de processamento a mais no servidor? (HARA, 2014)

A computação na nuvem disponibiliza diversas ferramentas que ajudam ao desenvolvedor ou as empresas de tecnologia a minimizar esses custos através da escalabilidade de recursos, ou seja, o recurso pode ser escalável automaticamente ou manualmente, somente no momento necessário e após essa necessidade o recurso pode voltar a usar a capacidade anterior de recurso do servidor. Podemos destacar alguns cenários onde a utilização da computação nas nuvens é extremamente aconselhável devido a sua capacidade de escalabilidade (HARA, 2014).

2.1.1 Crescimento rápido

Imagine que temos um software para implantar no Windows Azure e que não sabemos exatamente a quantidade de usuários que esse software pode atingir. O software pode ter poucos usuários e nunca sair disto, como também pode iniciar com um número pequeno de usuários e ter um crescimento inesperado. Na computação na nuvem é possível escalar de forma automática de acordo com a

demanda dos recursos necessários para que o software continue no ar sem apresentar problemas. (SOUZA, 2014)

Figura 1 – Crescimento-Rápido



Fonte: CHANNEL9, 2014.

2.1.2 Liga Desliga

Neste modelo o software é disponibilizado ao usuário final por um tempo, e pode ser desativado quando for necessário podendo voltar a funcionar quando quiser. No momento em que a empresa desenvolvedora retira o software do ar, não existe cobrança pela hospedagem do software, já que o software não está consumindo recursos do servidor na nuvem. No momento em que o software for disponibilizado novamente para o usuário final, o provedor de cloud computing inicia a contagem dos custos a serem cobrados.

Figura 2 – Liga-Desliga

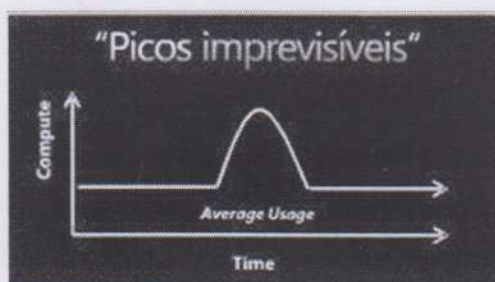


Fonte: CHANNEL9, 2014.

2.1.3 – Picos imprevisíveis

Se por acaso o software tem uma característica onde pode consumir recursos do servidor de forma imprevisível, a nuvem está preparada para isso, pois os recursos do software podem ser escalados de forma automática (SOUZA, 2014).

Figura 3 – Picos imprevisíveis

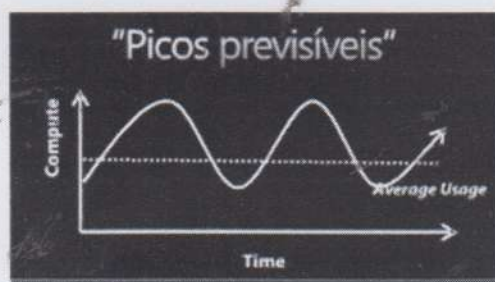


Fonte: CHANNEL9, 2014.

2.1.4 Picos Previsíveis

Da mesma forma, pode-se também controlar como a aplicação deve escalar seus recursos de acordo com uma demanda previamente planejada.

Figura 4 – Picos previsíveis



Fonte: CHANNEL9, 2014.

2.2 TIPOS DE SERVIÇOS

É importante também destacar como a nuvem disponibiliza seus recursos para seus usuários, sejam eles desenvolvedores, empresas de software, os usuários finais. Basicamente a nuvem oferece seus recursos baseado no conceito que tudo é entregue como um serviço, onde o serviço é cobrado de acordo com o seu

consumo. Podemos comparar esses serviços como os serviços de utilidade pública, como por exemplo água, luz e telefone, onde o usuário paga somente aquilo que consome. Esses serviços da nuvem podem ser divididos em tres categorias, sendo elas (SOUZA, 2014):

- a) Infraestrutura como serviço (IaaS);
- b) Plataforma como serviço (PaaS);
- c) Software como serviço (SaaS).

2.2.1 IaaS

É o serviço que disponibiliza um ambiente onde o usuário é responsável por instalar e administrar desde o sistema operacional até os aplicativos, como também recursos de rede e configuração do sistema operacional. Este serviço é disponibilizado ao usuário através de virtualização, onde o usuário pode alocar a quantidade necessária de recursos computacionais para seu uso. Neste modelo de serviço o usuário pode alocar uma quantidade maior de recurso sempre que necessário, de forma simples e rápida, bastando apenas alterar uma configuração nas máquinas virtuais disponíveis na nuvem. Isso simplifica o processo de aumento de recursos computacionais, referente ao modelo On-Premisse, que neste caso seria necessário migrar a aplicação para outro servidor.

2.2.2 PaaS

Neste ambiente de computação nas nuvens, os desenvolvedores de softwares tem todo o suporte de infraestrutura necessária para instalar, testar e desenvolver seus software. Neste caso a equipe de desenvolvimento não necessita se preocupar em montar e configurar toda a infraestrutura de um projeto. Isso permite que a equipe de desenvolvimento esteja focada no negócio propriamente dito, ou seja, no desenvolvimento do software especificamente, deixando de se preocupar com a infraestrutura em que o software estará funcionando, pois ao utilizar a PaaS todo o controle da infraestrutura é feito pelo provedor de PaaS,

bastando apenas que o usuário realize as configurações necessárias para que toda a infraestrutura da nuvem esteja disponível a ele.

2.2.3 SaaS

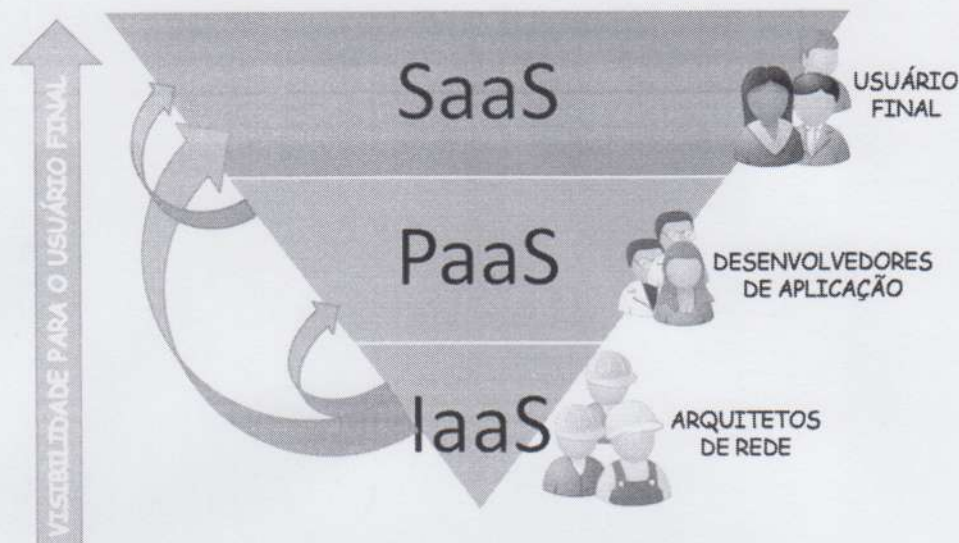
Este serviço de nuvem é o serviço onde o usuário final tem acesso. Basicamente o SaaS é o software que fica instalado na PaaS, e faz uso de todos os benefícios da computação na nuvem, como escalabilidade dinâmica e alta disponibilidade.

O modelo de negócio em um SaaS é diferente, pois o usuário final não deve se preocupar com investimentos relacionados a infraestrutura, pois toda a infraestrutura é gerenciada pela empresa desenvolvedora através da PaaS. Através deste modelo, a empresa desenvolvedora tem a possibilidade de ganhar clientes que antes no modelo on-premise não era possível, visto que o investimento desse cliente com infraestrutura no modelo on-premise era muito alto, ou até inviável, no modelo de nuvem esse investimento não existe, fazendo com que a negociação entre as empresas seja viável. A empresa desenvolvedora pode vender seus softwares para clientes de grande, médio e pequeno porte e os clientes que antes não usavam aplicação da empresa desenvolvedora por uma questão de custo com infraestrutura, hoje podem usar um SaaS que é totalmente viável.

O usuário do SaaS usa o software no modelo de um serviço que tem uma assinatura mensal, que pode ser cancelado quando o usuário quiser, e voltar a usar este serviço posteriormente.

A empresa desenvolvedora tem a possibilidade de diminuição dos custos na medida em que o software aumenta o número de usuários, ou seja, todo o custo acaba sendo rateado pelos usuários do software, permitindo que a empresa desenvolvedora, possa aplicar uma mensalidade mais competitiva no mercado em relação aos seus concorrente.

Figura 5 – Modelos de serviço na nuvem



Fonte: CHANNEL9, 2014.

2.3 VANTAGENS E DESVANTAGENS

É muito importante conhecer bem a tecnologia em que se trabalha, pois como se sabe, uma tecnologia não pode ser aplicada em qualquer situação, ela deve ser estudada a fundo para que se saiba aonde deve ser aplicada corretamente. Abaixo vamos conhecer um pouco das vantagens e desvantagens da computação nas nuvens. (HARA, 2014)

2.3.1 Vantagens

Sabemos que o desenvolvimento de um software não se resume apenas no fato de escrever o código fonte da aplicação, e entregar um dispositivo que contém o código fonte da aplicação para o cliente, mas exige também uma série de processos e outras responsabilidades que resulta na entrega definitiva de uma solução para o cliente. Um exemplo claro disso é o ambiente em que a aplicação será instalada, que em alguns casos o ambiente é todo estruturado localmente no cliente. Isso demanda a compra de servidores, periféricos e de profissionais que ao longo do tempo realize manutenção em toda a infraestrutura que deve ser mantida pela empresa. Caso o software tenha a estimativa de ser usado por muito tempo, talvez esse investimento possa valer a pena a longo prazo, mas caso este software tenha a

premissa de ficar no ar por um tempo determinado, o investimento em toda a infraestrutura acaba ficando caro demais. É aí que a computação nas nuvens é apresentada o como um grande benefício, pois toda a infraestrutura é de inteira responsabilidade dos serviços de nuvem onde a aplicação fica instalada sendo necessária apenas o monitoramento do ambiente que fica nas nuvens apenas para evitar desperdícios de alocação dos recursos utilizados, para assim diminuir o custo desse serviço, que é cobrado pelo uso dos recursos que o software está consumindo ao longo do tempo.

Como o serviço é cobrado pelo uso, podemos e devemos configurar todos os recursos que desejamos utilizar, como por exemplo a quantidade de instâncias em execução e também a configuração dessas instâncias como a quantidade de memória e processamento disponíveis. Esse tipo de configuração pode ser utilizada em alguns momentos onde podemos prever que a aplicação vai exigir mais recursos que o normal, devido algum tipo de operação realizada no cliente, neste caso podemos configurar todo o ambiente para que o ambiente possa disponibilizar para o nosso SaaS uma quantidade maior de recursos, isso vai aumentar de acordo com o recurso utilizado, o custo do serviço, mas também é possível configurar o ambiente para que ele libere recursos quando sabemos que o software não está utilizando recursos do servidor, é neste momento que o SaaS promove um ganho incrível ao custo de um projeto. Pense em uma aplicação de uma empresa que é usada simultaneamente por muitos usuários durante várias horas do dia e que em algumas horas do dia esses usuários realizam alguma operação que exige bastante recurso do servidor, mas durante o restante do dia o consumo do servidor é bem menor, e que ainda durante a noite o software não é utilizado pela grande massa de usuários e sim por alguns gestores de alto nível da empresa. Em uma aplicação SaaS podemos configurar os recursos necessários para que a aplicação funcione durante o dia, mas durante o horário onde os usuário realizam alguma atividade que vai exigir mais recursos do servidor, podemos configurar o ambiente para que ele tenha mais recursos disponíveis e durante a noite podemos deixar uma quantidade menor de recurso, mas suficiente para que os gestores da empresa tenha o software disponível e utilizando uma quantidade menor de recurso. Nesta mesma situação, se o cliente tivesse optado por investir na infraestrutura própria, talvez o custo desta infraestrutura seria maior que o custo da infraestrutura disponibilizada pelo SaaS,

pois o cliente teria que investir em equipamentos que suportem o nível máximo de recursos utilizados pela aplicação, ou seja, não existiria escalabilidade e o cliente não poderia economizar recurso no momento em que a aplicação não está exigindo tal recurso.

A preocupação com backups é dispensada, pois o banco de dados da aplicação também fica hospedado na nuvem e acaba sendo uma responsabilidade do servidos do SaaS que também acaba sendo responsável pela segurança desses arquivos, já que o mesmo encontra-se no seu data center.

A atualização do software também tem benefícios no modelo SaaS, pois esta é feita de forma transparente para o usuário, sem que o cliente tenha que disponibilizar acesso aos seus servidores locais, para que a empresa desenvolvedora possa realizar alguma atualização. O cliente pode ser notificado por email ou pela própria aplicação que houve uma atualização no software, para que ele saiba como operar o sistema caso a atualização altere alguma processo no uso do software.

Segundo Carraro & Wolter(2006, p4) no caso de um software mult-tenant ou mult-inquilino, onde o mesmo software é utilizado por vários clientes ao mesmo tempo, o custo desse software é rateado por todos os cliente que utilizam o software, e a empresa desenvolvedora pode tornar o preço da sua solução mais competitivos no mercado, levando em consideração que seus concorrentes não utilizam o SaaS como modelo de negócio.

No modelo SaaS a empresa desenvolvedora pode ganhar tempo no desenvolvimento do software, pois ela não vai mais se preocupar em preparar uma infraestrutura para testes e homologação do seu software, uma vez que todo o ambiente fica disponível na nuvem e pode ser configurável para consumir o mínimo de recurso para tal operação, acaba sendo dispensado o custo de preparar um ambiente para testes na empresa desenvolvedora e a equipe de desenvolvimento fica com tempo apenas para se concentrar em desenvolver o software o que é muito bom para o negócio em questão.

Também é importante destacar que o SaaS não pode ser pirateado, pois como a aplicação roda nas nuvens, é necessário ter acesso previamente disponibilizado pela empresa desenvolvedora para que o usuário possa usar o software.

2.3.2 Desvantagens

Como sabemos que a aplicação roda diretamente do servidor na nuvem, é claro que se o usuário estiver sem acesso à internet, ele não poderá acessar a aplicação e assim ele ficará sem condições de trabalhar com o software. Esse é um fator extremamente importante, imagine um cliente que é localizado em uma cidade distante das capitais, onde o acesso à internet é precário ou até não existe acesso a internet, neste caso a solução de SaaS é inviável para esse cliente, pois ele ficaria dependente do serviço de internet para acessar a aplicação e dependendo do negócio, ficar sem acessar a aplicação resulta em prejuízos ao estabelecimento. Nestas condições fica claro que o SaaS não é a solução mais viável para este cliente.

Quando um cliente decide usar uma aplicação hospedada na nuvem, a empresa desenvolvedora se depara com um possível problema que é a importação e exportação dos dados. Caso o cliente já venha utilizando uma outra aplicação, muitas vezes o cliente solicita que todo o cadastro seja importado para a nova aplicação, sendo assim a empresa desenvolvedora do SaaS deve criar mecanismo para importar esses dados para sua base de dados na nuvem. Da mesma forma, deve acontecer quando o cliente mudar o sistema atual para um concorrente, a empresa desenvolvedora do SaaS deve disponibilizar de alguma forma todos os dados desse cliente, definido anteriormente no contrato.

2.4 O QUE É ESCALABILIDADE

Ao desenvolver uma aplicação, devemos levar em consideração a quantidade de usuários que possivelmente vai usar o software, tendo em vista a quantidade de recursos computacionais que devem estar disponíveis para a aplicação funcionar sem apresentar problemas. Mas imagine que a aplicação em questão não tem ao certo o número de usuários que podem usar a aplicação, ou pode a qualquer momento aumentar de usuários de forma inesperada (CAMBIUCCI, 2014).

Nos dias de hoje é difícil pensar em desenvolver uma aplicação e não se preocupar com a escalabilidade dessa aplicação, pois é através dela, que podemos garantir que mesmo que a aplicação venha a sofrer um número grande de

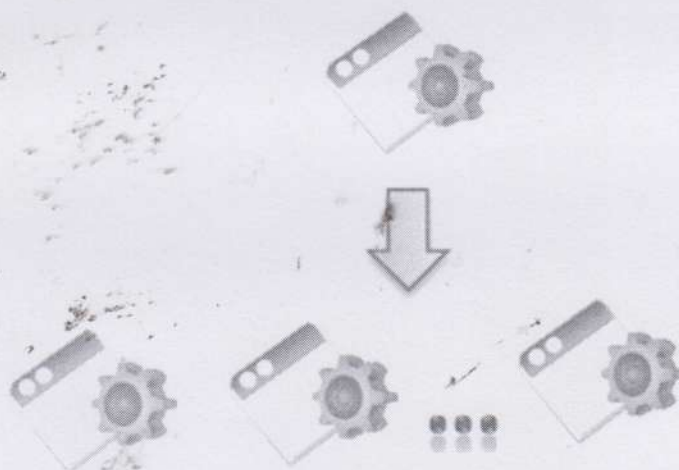
requisições ou um alto poder de processamento, ela não venha a ficar fora do ar por falta de recursos computacionais. Uma aplicação escalável garante que os recursos necessários para o seu funcionamento estarão disponíveis no momento que em a aplicação precisar, seja recursos memória, processamento ou de escrita no disco rígido.

Podemos escalar uma aplicação de duas formas. De forma horizontal ou vertical.

2.4.1 Escalabilidade horizontal

Entende-se por escalabilidade horizontal, quando alocamos mais servidores para fazer um compartilhamento de recursos para a mesma aplicação, ou seja, podemos ter quantos servidores forem necessários rodando a mesma aplicação, fazendo assim a divisão do trabalho pesado entre as máquinas.

Figura 6 – Escalabilidade Horizontal

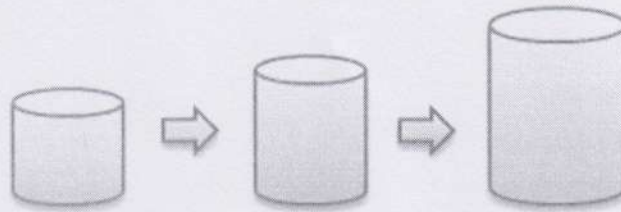


Fonte: CHANNEL9, 2014.

2.4.2 Escalabilidade vertical

Na escalabilidade vertical, podemos aumentar a quantidade de processamento em um único servidor, aumentando a quantidade de memória, velocidade do processador ou velocidade na leitura ou escrita dos discos rígidos deste servidor.

Figura 7 – Escalabilidade Vertical



Fonte: CHANNEL9, 2014.

2.5 GIT

O GIT é um sistema gerenciador de código fonte, que tem a finalidade de auxiliar uma equipe de desenvolvedores de sistemas, no controle e gerenciamento dos arquivos fonte dos sistemas de uma empresa de software. O GIT foi desenvolvido por Linus Torvalds, quando ele estava desenvolvendo o Kernel Linux, e hoje em dia é utilizado por muitos desenvolvedores no mundo. O git é um sistema gratuito com licença GNU General public Licence.

2.6 GitHub

É um repositório web de código fonte para aplicações que usam o GIT como gerenciador de código fonte, permitindo que o usuário possa compartilhar o código fonte hospedado no GitHub com outros usuários ou fazendo com que o código seja guardado em um repositório privado, onde somente os usuários autorizados possam visualizar esse código fonte. Para ter um repositório de códigos fonte privado no GitHub é necessário ter uma conta paga, mas caso o repositório seja público, pode ser usado uma conta gratuita. (BELL, BEER, 2014)

3 DESENVOLVIMENTO

Nesta seção será apresentado como realizar a implantação de um software no ambiente de nuvem da Microsoft, Windows Azure, e as particularidades importantes para ser implementada nos softwares implantados nesta plataforma.

3.1 WINDOWS AZURE

O Windows Azure é a PaaS (Plataform as a Service) da Microsoft com a finalidade de disponibilizar aos seus usuários a execução de serviços e aplicativos, baseados no conceito de nuvem. Todos os recursos tecnológicos do Windows Azure é disponível aos usuários através de uma rede de datacenters da Microsoft que encontram-se atualmente em vários locais do planeta inclusive no Brasil. Através dessa plataforma o usuário pode controlar de forma fácil e rápida recursos como balanceamento de carga entre instâncias diferentes de servidores ou máquinas virtuais, escalar a capacidade de processamento, memória ou escrita dos dados em Hard Disk dos servidores utilizados, pagando apenas pelo recurso que for usado.

O Windows Azure pode ser dividido em quatro categorias, são elas:

Windows Azure: É responsável por todos os recursos computacionais de hospedagem de aplicações e serviços, como também todo o gerenciamento de armazenamento de dados, como arquivos binários, usando o serviço Azure Storage.

SQL Azure: Este serviço tem a responsabilidade de prover ao usuário da plataforma todos os recursos necessários para implantação de um banco de dados Sql Server, a sincronização de dados com o ambiente on-premise e todo o gerenciamento desses dados na nuvem.

Windows Azure AppFabric: Recursos para publicação e gerenciamento de serviços, através do Service Bus, e controle de acesso, através do Access Control. Funcionalidades como Azure AppFabriccaching oferecem recursos adicionais para aplicação na nuvem com alta performance.

Windows Azure DataMarket: Oferece mecanismos para o controle de faturamento do Windows Azure, gerenciamento de pagamentos, informações sobre serviços publicados e relatórios para o controle das aplicações hospedadas na plataforma (CAMBIUCCI, 2014).

3.2 Modelos de roles(Papéis) no Windows Azure

No Windows Azure existe uma fronteira de isolamentos dos processos

executados na nuvem. Esse isolamento é feito pelos componentes nomeados Roles. Um serviço do Windows Azure pode ser composto por diversas Roles e cada uma pode ser comunicar com componentes internos e externos através de protocolos HTTP, HTTPS e TCP. Temos assim no Windows Azure três tipos de Roles:

Web Role: Esta role é responsável por todos os serviços referente a hospedagem web que utilizem o ISS (internet Information Services).

Worker Role: Serviço responsável por executar aplicações em background, como serviço ou processo que pode por exemplo consumir um serviço de enfileiramento de mensagens.

VM Role: Disponibiliza para o usuário um servidor web através de virtualização.

Com o objetivo de uma maior abstração na plataforma, as Web Roles e Worker Roles são disponibilizadas aos usuários da plataforma, em um formato de serviço, onde eles não necessitam se preocupar como esses serviços funcionam e não são responsáveis pela atualização de infraestrutura ou sistema operacional, se preocupando apenas com a implementação das aplicações que estão implantadas na plataforma. Já no caso as Vm Roles, os usuários são responsáveis por todo o controle e customização dos softwares e sistema operacional, oferecendo ao usuário maior controle sobre o ambiente contratado.

3.3 TIPOS DO AZURE STORAGE

O Azure Storage é o serviço da plataforma Windows Azure responsável por guardar os arquivos binários, como fotos, vídeos e outros tipos de documento de forma segura e escalável. Logo abaixo será descrito os principais tipos de Azure Storage disponível na plataforma:

3.3.1 Table

Armazena dados na mesma estrutura de uma tabela, onde pode ser criado uma estrutura do tipo Table e nessa estrutura pode ser armazenado vários registros. Essa estrutura não pode fazer relacionamento com outras tabelas como acontece nos banco de dados relacionais.

3.3.2 Blob

Os tipos blobs são responsáveis por armazenar dados binários ou textos, que são agrupados por containers para auxiliar o armazenamento desses arquivos.

3.3.3 Queues

O tipo Queue, ou fila, oferece um mecanismo de comunicação assíncrona entre componentes de uma solução no Windows Azure. Os tipos Queues permitem a comunicação entre processos, web roles e worker roles, por exemplo, através de mensagens colocadas na fila, mantendo um desacoplamento e alto desempenho para aplicações publicadas na plataforma.

3.4 SQL AZURE

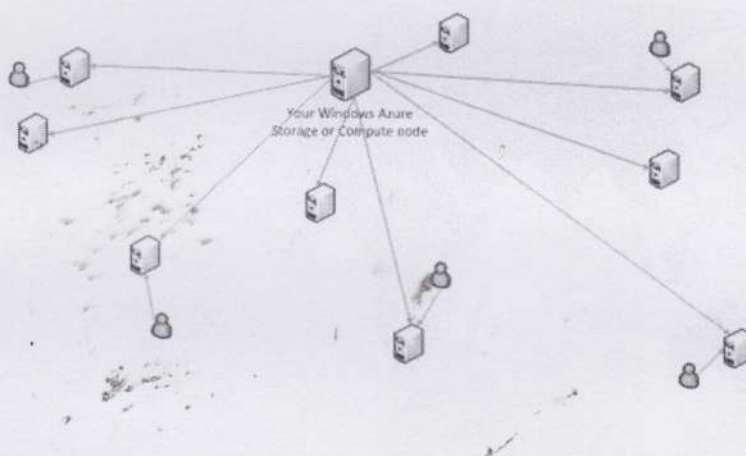
Através do SQL Azure é possível hospedar na nuvem, uma instância de um banco de dados SQL Server da Microsoft e utilizar recursos deste serviço como por exemplo persistência de dados utilizando o seu modelo de programação onde o desenvolvedor que trabalha com o ambiente SQL Server 2008 local (on-premise) pode trabalhar da mesma forma no ambiente de dados SQL Azure, utilizando os mesmos recursos do .Net Framework. Outros recursos como business intelligence, sincronização entre base de dados, replicação de dados, relatórios, data mining, podem ser destacados nessa tecnologia.

3.5 CDN – Rede de distribuição de conteúdo

Os aplicativos e serviços hospedados no Windows Azure, podem ser acessados pelos seus usuários através da internet em qualquer parte do mundo. Isso quer dizer que ao desenvolver uma aplicação, devemos considerar o fato de que um usuário da aplicação pode estar em um país diferente do data center aonde a aplicação está hospedada, e isso significa que ao acessar a aplicação este possível usuário terá um tempo de espera maior que o usuário que acessar a aplicação no mesmo país do data center aonde a aplicação está hospedada. Chamamos isso de latência ou

tempo de resposta. Enquanto mais distante for o país de acesso ao país aonde a aplicação estar hospedada, maior a latência. Para resolver este problema o Windows Azure tem o serviço de CDN, que tem a finalidade de criar cópia dos arquivos acessados pelos usuários, em servidores estratégicos para que os usuários que acessam a aplicação que encontram-se em países diferentes do país aonde a aplicação estar hospedada, tenham um tempo de resposta menor que o tempo de resposta atual. Esse serviço deve ser ativado e configurado pelo usuário do Windows Azure informando o local aonde o Windows Azure deve armazenar uma cópia dos arquivos, para quando um usuário próximo ao data center escolhido acessar a aplicação, o Windows Azure vai se encarregar de transferir os arquivos dessa data center para o usuário, diminuindo a latência ou tempo de resposta da aplicação.

Figura 8 – Modelo de CDN disponibilizado pelo Windows Azure



Fonte: CHANNEL9, 2014.

3.6 WINDOWS AZURE CACHE

Ao desenvolver uma aplicação web, é comum o uso de variáveis de sessão. Elas são importantes para guardar qualquer informação do usuário no servidor e a aplicação pode fazer uso dessa informação a qualquer momento. As variáveis de sessão são manipuladas pela aplicação e guardadas no servidor web. Devemos tomar alguns cuidados ao usar variáveis de sessão no Windows Azure, por exemplo quando temos uma aplicação sendo executada em mais de uma instancia, pois

dessa forma o Windows Azure utiliza automaticamente um load balancer (Balanceador de carga). Isso quer dizer que a sua aplicação pode funcionar hora em um servidor e hora em outro servidor, e dessa forma a aplicação vai perder o controle das variáveis de sessão criadas por ela.

Neste caso podemos fazer uso do Windows Azure Cache, que vai garantir que independente da instancia onde a aplicação estiver sendo executada, as variáveis de sessão serão acessadas e controladas pela aplicação.

3.7 FAZENDO DEPLOY DE UMA APLICAÇÃO PARA O WINDOWS AZURE

O Windows Azure disponibiliza diversas formas de realizar deploy de uma aplicação para seu ambiente. Uma aplicação web, por exemplo, pode ser publicada no Windows Azure apenas com o auxílio do Visual Studio, bastando realizar algumas configurações para que essa tarefa seja feita sem muitas complicações ou pode ser feito através da transferências dos arquivos via FTP. Existem outras formas de implantar um software no Windows Azure, mas nesta sessão será apresentado uma forma de realizar a publicação de uma aplicação web, integrando o Windows Azure com o GIT. Essa forma de realizar deploy, auxilia o desenvolvedor a publicar uma aplicação no Windows Azure de forma rápida, mas para isso, deve ser feito algumas configurações no Windows Azure.

3.7.1 Realizando o cadastro no Windows Azure

Acesse o site www.azure.com e click em minha conta, selecione avaliação gratuita, depois em testar agora. Neste momento você fará um cadastro para usar os serviços do Windows Azure e será solicitado um telefone, para envio de SMS com o código de ativação da conta, os dados do seu cartão de crédito para ser feito a cobrança dos serviços do Windows Azure, Lembrando que somente os três primeiros meses serão gratuitos e quando expirar o período de gratuidade, todos os serviços serão suspensos automaticamente, evitando transtornos com cobranças desnecessárias, e o usuário deve ativar os serviços que deseja utilizar manualmente. Afigura 9 mostra a tela de cadastro do Windows Azure.

Figura 9 – Tela de cadastro Windows Azure

Fonte: AZURE, 2014.

3.7.2 Cadastrando um serviço de site no Windows Azure

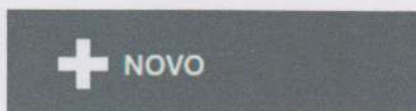
Após realizar o cadastro, efetue login no Windows Azure e clique no menu "Sites" como mostra a figura 10 e depois clique no botão "novo" como mostra a figura 11.

Figura 10 – Botão sites do Windows Azure



Fonte: AZURE, 2014.

Figura 11 – Botão novo do Windows Azure

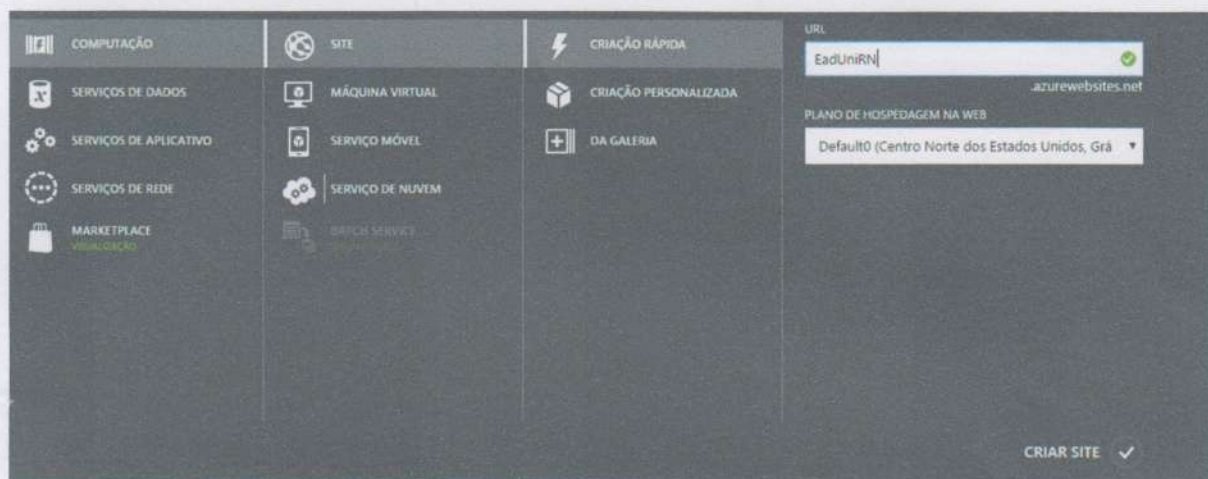


Fonte: AZURE, 2014.

Escolha a opção "criação rápida" e uma URL para nomear a aplicação que você deseja publicar e defina também a região onde a aplicação será hospedada.

Depois disso clique em Criar site. A figura 12 mostra a tela que deve ser inserida as informações.

Figura 12 – Tela de criação de serviços de site

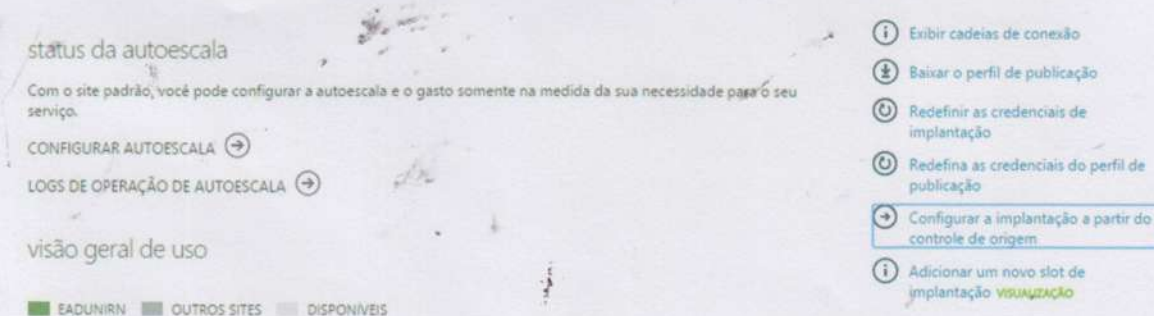


Fonte: AZURE, 2014.

3.7.3 Integrando o GIT com Windows Azure

Agora selecione o site que você acabou de criar e clique no menu “painel”. Ao lado direito da tela, clique no link “Configurar a implantação a partir do controle de origem” como é mostrado na figura 13.

Figura 13 – Configurando implantação

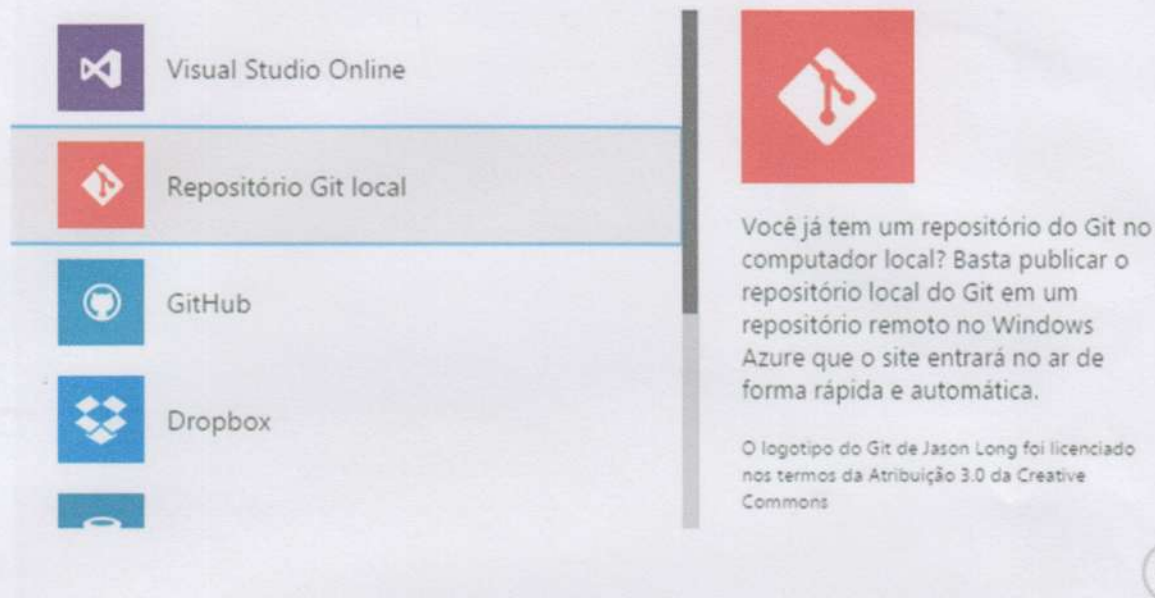


Fonte: AZURE, 2014.

Logo em seguida será mostrado uma lista com várias opções como é mostrado na figura 14. Nesta lista selecione “Repositório GIT local” e depois selecione o menu implantação

Figura 14 – Definindo repositório

Onde está seu código-fonte?



Fonte: AZURE, 2014.

Abra o git instalado na máquina aonde estão os arquivos do software a ser implantado no Windows Azure e na pasta do software execute o comando "git clone" mais o endereço da url capturada na tela de implantação como é mostrado na figura 15 para copiar na máquina local o repositório GIT criado no Windows Azure.

Figura 15 – Repositório GIT

SEU REPOSITÓRIO

URL DO GIT

<https://tecsoftdesenvolvimento@eadunirn.scm.>



Fonte: AZURE, 2014.

3.7.4 Publicando uma aplicação no Azure com auxílio do GIT

Faça a cópia dos arquivos de publicação do software para a pasta que acabou de ser criada e com os arquivos copiados faça o seguinte comando no Git: Git add, Git commit -m "Primeira publicação" e Git push origin master. A figura 16 mostra os comandos executados no Git.

Figura 16 – Publicando aplicação pelo git

```

producao01@PRODUCAO_001 /d/EadUniRN (master)
$ git add .
warning: LF will be replaced by CRLF in site.txt.
The file will have its original line endings in your working directory.
producao01@PRODUCAO_001 /d/EadUniRN (master)
$ git commit -am "primeira publicação"
[master (root-commit) dcd5187] primeira publicação
warning: LF will be replaced by CRLF in site.txt.
The file will have its original line endings in your working directory.
1 file changed, 1 insertion(+)
 create mode 100644 site.txt

producao01@PRODUCAO_001 /d/EadUniRN (master)
$ git push origin master
Password for 'https://tecsoftdesenvolvimento@eadunirn.scm.azurewebsites.net:443':
Counting objects: 3, done.
Writing objects: 100% (3/3), 232 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote: Updating branch 'master'.
remote: Updating submodules.
remote: Preparing deployment for commit id 'dcd518761a'.
remote: Generating deployment script.
remote: Generating deployment script for Web Site
remote: Generated deployment script files
remote: Running deployment command...
remote: Handling Basic Web Site deployment.
remote: KuduSync.NET from: 'D:\home\site\repository' to: 'D:\home\site\wwwroot'
remote: Deleting file: 'hostingstart.html'
remote: Copying file: 'site.txt'
remote: Finished successfully.
remote: Deployment successful.
To https://tecsoftdesenvolvimento@eadunirn.scm.azurewebsites.net:443/EadUniRN.git
 * [new branch]      master -> master

```

4 CONCLUSÃO

Implantar um software escalável e de alta disponibilidade não é uma tarefa tão complicada quando temos o auxílio de um data center como o Windows Azure, disponibilizando para os desenvolvedores de software, todos os benefícios de segurança e robustez de um data center. Um desenvolvedor autônomo ou pequenas empresas podem fazer uso de recursos tecnológicos extremamente atuais, sem fazer altos investimentos iniciais, e isso traz de certa forma uma maneira de competir com grandes empresas, ou pode trazer para pequenas empresas a chance de ter soluções mais competitivas no mercado.

É possível também se beneficiar através de recursos de escalabilidade que o Windows Azure disponibiliza aos seus usuários. Ter uma aplicação disponível na internet, e pagar somente pelos recursos que a aplicação esta consumindo, traz a oportunidade da empresa ter soluções com preços mais competitivos no mercado, pois o custo da aplicação pode ser controlado afim de que a empresa desenvolvedora pague somente pelos recursos que utilizar, evitando o desperdício de recursos alocados ao projeto.

Pode se verificar que o Windows Azure é uma plataforma de nuvem bastante completa, que permite a integração com diversas ferramentas usadas pelos desenvolvedores, como por exemplo o Visual Studio e o GIT, e que também dá suporte a implantação de softwares de diversas linguagens de programação, deixando claro que a intenção não é apenas permitir que produtos da Microsoft interajam com essa plataforma, mas que produtos de outras empresas possam fazer uso dessa plataforma de nuvem.

REFERÊNCIAS

AZURE Papeis de parede. Disponível em: <<http://www.azure.com.br>>. Acesso em: 05 dez. 2014.

BELL, Peter; BEER, Brent. **Introdução ao GitHub**: Um guia que não é técnico. São Paulo: Novatec, 2014.

CAMBIUCCI, Waldemir. **Introdução sobre o SQL Azure**. Disponível em: <<http://msdn.microsoft.com/pt-br/library/hh147515.aspx>>. Acesso em: 05 dez. 2014.

CAMBIUCCI, Waldemir. **Introdução sobre Windows Azure**. Disponível em: <<http://msdn.microsoft.com/pt-br/library/hh150078.aspx>>. Acessado em: 05.12.2014.

CHANNEL9. **Blogs**: O que é computação na nuvem?. Disponível em: <<http://channel9.msdn.com/posts/O-que-e-computao-na-nuvem>>. Acesso em: 05 dez. 2014.

CORDEIRO, Rogerio. **Migrando uma aplicação para o Azure**: Introdução. Disponível em: <http://channel9.msdn.com/posts/Migrando-uma-aplicao-para-o-Azure-Introducao>. Acessado em: 05 dez. 2014.

GODINHO, Rafael. **Introdução ao Windows Azure Compute(Roles)**. Disponível em: <<http://msdn.microsoft.com/pt-br/windowsazure/500326>>. Acesso em: 05 dez. 2014.

HARA, Fábio. **Fundamentos de computação em nuvem**. Disponível em: <<http://www.microsoftvirtualacademy.com/training-courses/fundamentos-de-computa-o-em-nuvem>>. Acesso em: 05 dez. 2014.

SOUZA, Vinícius. **Introdução sobre Windows Azure Storage**. Disponível em: <<http://channel9.msdn.com/posts/Introducao-sobre-o-Windows-Azure-Storage>>. Acesso em: 05 dez. 2014.

SOUZA, Vinícius. **O que é computação nas nuvens?**. Disponível em: <<http://channel9.msdn.com/posts/O-que-e-computao-na-nuvem>>. Acesso em: 05 dez. 2014.