



ARQUITETURA ORIENTADA A SERVIÇOS E MICROSSERVIÇOS, COMPUTAÇÃO EM NUVEM E MOBILE

BRASÍLIA-DF.

Elaboração

Jorge Umberto Scatolin Marques

Produção

Equipe Técnica de Avaliação, Revisão Linguística e Editoração

Sumário

APRESENTAÇÃO.....	5
ORGANIZAÇÃO DO CADERNO DE ESTUDOS E PESQUISA	6
INTRODUÇÃO.....	8
UNIDADE I	
PRINCÍPIOS DA COMPUTAÇÃO EM NUVEM	9
CAPÍTULO 1	
CONCEITO	9
CAPÍTULO 2	
ALINHAMENTO ESTRATÉGICO E TECNOLOGIA DA INFORMAÇÃO.....	18
CAPÍTULO 3	
COMPONENTES DE COMPUTAÇÃO EM NUVEM	24
CAPÍTULO 4	
BENEFÍCIOS, DESAFIOS E RISCOS	29
UNIDADE II	
SERVIÇOS EM NUVEM.....	36
CAPÍTULO 1	
INFRAESTRUTURA COMO SERVIÇO – IAAS	37
CAPÍTULO 2	
PLATAFORMA COMO SERVIÇO – PAAS	41
CAPÍTULO 3	
SOFTWARE COMO SERVIÇO – SAAS	48
CAPÍTULO 4	
TECNOLOGIA DA INFORMAÇÃO COMO SERVIÇO – ITAAS.....	51
UNIDADE III	
INFRAESTRUTURA EM NUVEM – DATACENTER.....	55
CAPÍTULO 1	
ASPECTOS GERAIS.....	55
CAPÍTULO 2	
EFICIÊNCIA ENERGÉTICA.....	59

CAPÍTULO 3	
ARQUITETURA E INFRAESTRUTURA	63
CAPÍTULO 4	
COMPUTAÇÃO EM NUVEM COMO NEGÓCIO	67
UNIDADE IV	
APLICAÇÕES	71
CAPÍTULO 1	
WEB API	71
CAPÍTULO 2	
MOBILE	80
CAPÍTULO 3	
MICROSSERVIÇOS	86
CAPÍTULO 4	
SEGURANÇA	92
REFERÊNCIAS	99

Apresentação

Caro aluno

A proposta editorial deste Caderno de Estudos e Pesquisa reúne elementos que se entendem necessários para o desenvolvimento do estudo com segurança e qualidade. Caracteriza-se pela atualidade, dinâmica e pertinência de seu conteúdo, bem como pela interatividade e modernidade de sua estrutura formal, adequadas à metodologia da Educação a Distância – EaD.

Pretende-se, com este material, levá-lo à reflexão e à compreensão da pluralidade dos conhecimentos a serem oferecidos, possibilitando-lhe ampliar conceitos específicos da área e atuar de forma competente e conscienciosa, como convém ao profissional que busca a formação continuada para vencer os desafios que a evolução científico-tecnológica impõe ao mundo contemporâneo.

Elaborou-se a presente publicação com a intenção de torná-la subsídio valioso, de modo a facilitar sua caminhada na trajetória a ser percorrida tanto na vida pessoal quanto na profissional. Utilize-a como instrumento para seu sucesso na carreira.

Conselho Editorial

Organização do Caderno de Estudos e Pesquisa

Para facilitar seu estudo, os conteúdos são organizados em unidades, subdivididas em capítulos, de forma didática, objetiva e coerente. Eles serão abordados por meio de textos básicos, com questões para reflexão, entre outros recursos editoriais que visam tornar sua leitura mais agradável. Ao final, serão indicadas, também, fontes de consulta para aprofundar seus estudos com leituras e pesquisas complementares.

A seguir, apresentamos uma breve descrição dos ícones utilizados na organização dos Cadernos de Estudos e Pesquisa.



Provocação

Textos que buscam instigar o aluno a refletir sobre determinado assunto antes mesmo de iniciar sua leitura ou após algum trecho pertinente para o autor conteudista.



Para refletir

Questões inseridas no decorrer do estudo a fim de que o aluno faça uma pausa e reflita sobre o conteúdo estudado ou temas que o ajudem em seu raciocínio. É importante que ele verifique seus conhecimentos, suas experiências e seus sentimentos. As reflexões são o ponto de partida para a construção de suas conclusões.



Sugestão de estudo complementar

Sugestões de leituras adicionais, filmes e sites para aprofundamento do estudo, discussões em fóruns ou encontros presenciais quando for o caso.



Atenção

Chamadas para alertar detalhes/tópicos importantes que contribuam para a síntese/conclusão do assunto abordado.

**Saiba mais**

Informações complementares para elucidar a construção das sínteses/conclusões sobre o assunto abordado.

**Sintetizando**

Trecho que busca resumir informações relevantes do conteúdo, facilitando o entendimento pelo aluno sobre trechos mais complexos.

**Para (não) finalizar**

Texto integrador, ao final do módulo, que motiva o aluno a continuar a aprendizagem ou estimula ponderações complementares sobre o módulo estudado.

Introdução

De todo o imenso leque de profissões relacionadas à Tecnologia da Informação, o desenvolvimento de software é uma das que mais mudam ao longo de um dia. A cada minuto, é lançado um conceito, uma biblioteca, uma técnica, uma ferramenta nova que facilite a vida do desenvolvedor.

Essas mudanças, por sua vez, alteram drasticamente a forma de trabalho nas organizações, como a possibilidade de trabalhos remotos, em que o desenvolvedor pode trabalhar de casa, o chamado *home office*, ou de outros lugares, como de uma pousada tranquila, de um posto de alimentação em uma autoestrada. Essa mobilidade toda se deve ao fato de que as grandes empresas estão aderindo a um novo conceito de computação: a cloud computing, computação em nuvem.

Com a computação em nuvem, o desenvolvedor tem em mãos toda a estrutura de desenvolvimento necessária para criar, dar manutenções ou fazer alterações em sistemas feitos em qualquer linguagem de programação ou arquitetura de hardware. Como se não bastasse a disponibilização das ferramentas de desenvolvimentos, oferecidos por meio de um serviço chamado PaaS, o desenvolvedor conta com a facilidade de criar ambientes de testes ou escalonar ambientes de banco de dados ou processamento com tanta facilidade que pode melhorar a carga do time de operação, em um conceito DevOps, por exemplo.

Além da facilidade da mobilidade de sua caixa de ferramenta ou de uma arquitetura de TI completa, as empresas poderão pagar somente o quanto forem utilizar e configurar todo o ambiente por meio de dashboards intuitivos e com ambiente web.

Objetivos

- » Compreender o conceito de computação em nuvem.
- » Estudar técnicas e exemplos de como decidir a migração para a nuvem.
- » Analisar as principais ferramentas oferecidas pelas grandes empresas como a Microsoft, o Google, a Amazon, a IBM e a Red Hat.

CAPÍTULO 1

Conceito

Antes de definirmos o que é computação em nuvem, vamos abordar alguns conceitos sobre computação em geral. Quando falamos sobre aquela computação local, que um dia será levada para a nuvem, estamos falando da capacidade de processamento, armazenamento, desenvolvimento de aplicações, ou seja, muitas atividades que as empresas costumam ter em seus departamentos. Por exemplo, hoje ainda há empresas que possuem um data center interno, juntamente com o departamento de TI, que conta com vários funcionários com habilidades que vão desde o desenvolvimento de software até a manutenção de hardwares e rede.

Um data center de médio porte deverá estar sob um ambiente físico apropriado, com refrigeração, backups de hardware, backups de software e backups de energia. Além de todo esse ambiente, também conta com os licenciamentos e contratos de softwares aplicativos e sistemas operacionais. Por mais que gerentes e CIOs mantenham o cenário de infraestrutura organizado, este acaba se tornando inviável, pois o processo de alinhá-lo às estratégias da organização é burocrático e lento.

Foi baseando-se nesses cenários de caos que provedores começaram a disponibilizar o poder computacional local, para fora dos limites físicos das empresas. Criar um ambiente computacional em que seu hardware pudesse ser compartilhado entre cada ambiente isolado por clientes pareceu algo muito lucrativo para ambos os lados. A empresa que contratasse o ambiente poderia fazer a aquisição de hardware e software sob demanda e não mais se preocupar em atualizar e manter um data center próprio.

O provedor poderia disponibilizar um ambiente otimizado e específico para esse propósito, podendo espalhar suas sedes pelo mundo, conforme vantagens oferecidas. Embora esse ambiente pareça perfeito, veremos mais adiante que essas mudanças de paradigmas poderão tornar o processo mais lento.

Definição

Podemos começar a definir Computação em Nuvem como uma forma de disponibilizar recursos computacionais fora dos limites físicos da organização, dos quais o usuário tem total liberdade de provisionar os recursos que ele está adquirindo.

Além de obrigatoriamente estar fora dos limites físicos da empresa, deverão ser disponibilizados serviços de infraestrutura (IaaS), Softwares como Serviços (SaaS), Plataforma como Serviços (PaaS) e Tecnologia da Informação como Serviço (ITaaS) que, além de serem oferecidos para consumo, ferramentas administrativas e aquisições sob demanda, deverão ser parte do pacote, ou seja, se o cliente precisar contratar recursos mínimos e depois for necessário escalonar, esse recurso também tem que ser disponibilizado.

O cliente não precisa saber como o data center do provedor está organizado, não precisa saber se possui refrigeração, nem a marca dos equipamentos etc. O que ele precisa é ter um bom contrato que disponibiliza o serviço em um determinado período, normalmente 24 horas por 7 dias da semana e a garantia de que os recursos estarão lá sempre que for preciso, independentemente de localização geográfica.

Histórico

Antes do surgimento da computação em nuvem, existia um tipo de computação chamado cliente/server, que dispunha de todos os recursos centralizados em um servidor, como armazenamento, dados, aplicativos e demais controles. Se um usuário precisasse utilizar alguns desses recursos, deveria ter autorização para efetuar o acesso apropriado para poder fazer seu trabalho, por exemplo. Isso se tornaria uma grande dificuldade a partir do crescimento da necessidade de utilização desses recursos.

Diante dessa situação, computadores que estavam na mesma rede foram modificados para servir e ajudar a distribuir esses recursos, o que fez surgir, então, a computação distribuída, conceito que se tornou Computação em Nuvem quando seus recursos começaram a sair dos limites físicos das organizações e serem vendidos para outras empresas.

É possível ver um breve histórico sobre a computação em nuvem por meio do link: <https://www.ipm.com.br/blog/historia-da-computacao-em-nuvem-como-surgiu-a-cloud-computing>.

» 1960 – O surgimento

Algumas literaturas mencionam que o conceito de computação em nuvem surgiu por declarações de dois especialistas: John McCarthy, que, em um discurso no *Massachusetts Institute of Technology* (MIT), nos EUA, em 1961, sugeriu que a computação em nuvem fosse oferecida como serviço, como água, luz etc. e paga somente pelo uso, e sugeriu o compartilhamento de equipamentos, de forma simultânea, entre dois ou mais usuários; e Joseph Carl Robnett Licklider, que, em 1961, conseguiu um jeito de pessoas compartilharem dados de forma global, criando a ArpaNet para a agência americana *Advanced Research and Projects Agency* (Arpa), com o objetivo de unir bases militares e departamentos de pesquisa do governo americano.

» 1990: Nuvem

O professor de sistemas de informação Ramnath Chellappa, em uma palestra acadêmica, em 1997, usou o termo nuvem baseado no desenho utilizado para representar a Internet.

» 1999 – Primeira Empresa

A americana Salesforce.com ficou conhecida como a primeira empresa a disponibilizar aplicações empresariais pela web.

» Anos 2000: a força da nuvem

A partir dos anos 2000, a *cloud computing* começou a ganhar mais força e passou a ser oferecida comercialmente.

» 2003 – Princípio dos grandes nomes

A Amazon entra no negócio de computação em nuvens, lançando um conjunto de serviços que incluem armazenamento, computação e inteligência humana. O pacote é baseado em nuvem, chamado Amazon EC2/S3, que se torna o primeiro serviço de nuvem acessível em larga escala.

» 2008 e 2009 – Novos grandes nomes

As gigantes Google e Microsoft lançam no mercado seus serviços de custo baixo e inovadores: o Google AppEngine e Microsoft Azure. Depois disso, inúmeras empresas ao redor do mundo foram lançando produtos cada vez mais inovadores e de baixo custo.

Padronização do conceito

Segundo o NIST (2011), a computação em nuvem é um modelo com acesso constante, conveniente e sob demanda da rede a um conjunto de recursos de computação configuráveis que pode ser rapidamente provisionado e liberado com o mínimo esforço de gerenciamento ou interação do provedor de serviços.

O modelo sugerido pelo NIST possui cinco características essenciais:

- » Autoatendimento sob demanda: o provisionamento poderá ser feito pelos consumidores de recursos, conforme sua necessidade e capacidade, sem que haja interação com seres humanos – por exemplo, poder aumentar ou contratar espaço de armazenamento, utilizar ligações telefônicas para pedir por esse serviço.
- » Amplo acesso a rede: os recursos contratados deverão estar disponíveis pela rede e acessados por equipamentos que utilizam interfaces web ou clientes em linha de comando.
- » Agrupamento de recursos: recursos poderão ser criados para que haja facilidades na contratação; um exemplo seria a contratação de um servidor para virtualizar um sistema operacional. Poderia haver modelos prontos disponíveis que informassem se o cliente gostaria de um equipamento com maior poder de processamento, armazenamento ou memória.
- » Rápida elasticidade: deverá haver elasticidade nos recursos já provisionados, de acordo com a demanda do consumidor. Para este, os recursos deverão parecer ilimitados, podendo, a qualquer momento, e em qualquer quantidade, contratar ou cancelar recursos.
- » Serviço medido: os provedores de serviços em nuvem deverão controlar e otimizar automaticamente a utilização de recursos, e o monitoramento deverá ser feito, apropriadamente, ao seu tipo de serviço. Para que tudo fique transparente entre o consumidor e o provedor, os recursos deverão ser monitorados e controlados, emitindo relatórios dos serviços utilizados.

Quatro modelos de implementação

Nuvem privada

O conceito de nuvem privada se dá quando a infraestrutura em nuvem é criada e provisionada para uma única empresa. Normalmente cria-se um ambiente seguro e restrito do qual somente membros autorizados podem acessar a nuvem, que está protegida por um firewall, garantindo privacidade, segurança e flexibilidade.

A organização é feita em nível lógico, ou seja, de software, pelos administradores da empresa ou terceiros. Depois que o ambiente for devidamente provisionado, as configurações poderão ser efetuadas como se todos estivessem em uma mesma infraestrutura, pois este é o contexto que a nuvem privada utiliza: IaaS.

Economicamente, a IaaS não tem grandes vantagens em relação às demais arquiteturas, principalmente da nuvem pública, pois é efetiva somente no consumo de energia do servidor, e isso poderá ser resolvido com a utilização de outros modelos de nuvens. Outro fator que poderá afetar a economia de recursos é a possibilidade de falhas individuais em ambientes virtualizados, localizados dentro das empresas. Em uma nuvem, quando isso acontece, as partições são trocadas, e os demais servidores não são afetados.

Na nuvem privada, o acesso aos recursos não é compartilhado entre os demais consumidores, ou seja, se uma quantidade específica de processamento foi contratada, mesmo que não seja utilizada em sua carga de pico, todo o processamento estará disponível, e a empresa pagará por isso.

Uma das vantagens da nuvem privada é que não ocorre o compartilhamento de recursos com outras organizações, ou seja, o consumidor terá à disposição uma infraestrutura de hardware e software única, provisionada para sua demanda; o controle é total sobre o ambiente, facilitando a governança e a segurança, mantendo a confidencialidade dos dados. A desvantagem é que demandará mais profissionais de TI, com nível de conhecimento maior, disponíveis para fazer diversos tipos de interferência.

Nuvem pública

A infraestrutura em um data center de nuvem pública é aberta ao público em geral, porém cada empresa continuará tendo acessos restritos ao seu ambiente, e os recursos continuam sendo provisionados conforme a necessidade de cada consumidor. A nuvem pública permite o compartilhamento de recursos entre os usuários isoladamente, e o nome dessa técnica chama-se arquitetura de multilocatário. Seu acesso deverá ser feito

por meio de *dashboards* acessados via Web. A grande diferença da nuvem privada, porém, está em como cada consumidor acessará esses recursos. Na nuvem pública, por exemplo, o processador será compartilhado entre dois ou mais consumidores, ou seja, o consumidor “A” contrata uma determinada quantidade de processamento e, quando não estiver utilizando, ela será automaticamente utilizada pelo consumidor B. Isso torna a nuvem pública com um custo menor e com a possibilidade de disponibilizar período de testes gratuitos, porém pode torná-la um pouco menos eficiente que a privada. Imagine, no exemplo anterior, se o processamento, que foi realocado para o consumidor B, precisasse voltar para o consumidor A. A perda se daria justamente nos milissegundos dessa troca. É pouco tempo de gap, mas, ao longo de um dia de muitas trocas, isso seria percebido. Essa arquitetura se chama arquitetura de multilocatário, pois, como mencionado, os recursos são compartilhados entre usuários.

Economicamente, é o modelo mais viável para o cliente e para o provedor, pois ambos se beneficiam do compartilhamento isolado de recursos, e a remuneração é feita conforme o uso. Outra vantagem está no sistema de falhas. Nessa arquitetura, apesar de a estrutura do data center ser composta por diversos servidores, para os técnicos e sistemas operacionais ele é visto como um único equipamento, ou seja, caso um disco falhe, nenhuma informação será perdida, pois ela estará disposta em uma matriz formada por diversos outros discos que dispõem de armazenamento redundante dentro e fora da estrutura física do data center. Como os equipamentos são Hot Swap (possibilidade de troca com o servidor funcionando), o cliente não perceberá que houve problema, e os serviços não serão interrompidos.

A segurança é um fator muito bem monitorado em um ambiente de nuvem pública, e isso se deve ao fato de as informações precisarem estar protegidas de acessos indevidos. Para isso, equipes devem ser muito bem treinadas e munidas de muitas ferramentas de monitoramento e proteção, bem como ter conhecimento de todos os tipos de sistemas operacionais.

Os benefícios da computação em nuvem dependerão dos projetos, de arquiteturas de TI que cada organização definirá. Existe a possibilidade de um projeto mal elaborado ser associado a um tipo de nuvem incompatível. Entretanto, em termos gerais, as soluções de nuvem pública permitem que os consumidores escalem seus recursos, em uma proporção muito alta, quase infinita, o que não conseguiria em um data center local. Conforme a empresa cresce, ela não precisará adquirir hardware ou manter uma estrutura em expansão; isso acontecerá também com os aplicativos. Ou seja, os usuários não

vão precisar se preocupar com instalações ou alterações de aplicativos em seus dispositivos. Em vez disso, estarão hospedados na nuvem e podem permanecer continuamente atualizados com as mais recentes opções e segurança.

Nuvem da comunidade

Na nuvem comunitária, o formato se assemelha com a nuvem pública, porém o número de usuários que a utilizam é mais reduzido. Isso se deve ao fato de que apenas empresas com objetivos em comum, sejam eles relacionados à missão, política, segurança, entre outros interesses, beneficiam-se da nuvem comunitária.

As empresas se beneficiam dos custos e das facilidades da nuvem pública, como troca de equipamentos, segurança, proteção a falhas e também da confidencialidade da nuvem privada. Essas empresas possuem preocupação e interesses comuns que podem ser compartilhados e gerenciados por elas, em vez de apenas provisionar espaço em uma nuvem pública, podendo testar e trabalhar em uma plataforma em nuvem segura, “dedicada” e até compatível com certos regulamentos.

A parte realmente interessante é que, com uma nuvem da comunidade, a presença pode ser no local ou fora dele. Com a rede comunitária, é possível haver processos de migração gradual e, por isso ela, se torna muito flexível em vários níveis, são compatíveis com todos os usuários e também pode ser provisionada conforme o uso.

Outro ponto forte dessa arquitetura é a necessidade de o parque tecnológico poder acompanhar os negócios das organizações, como troca de funcionários, de departamentos, aquisições e criações de filiais, sistema de gerenciamento em comum.

Contra problemas de falhas em dados, é possível que replicações e caches de dados sejam efetuados para manter sua integridade e proteção. Também, como os dados não estarão localmente, qualquer desastre em uma das filiais não afetará o core de negócios das demais.

Em relação ao time de funcionários de TI, os requisitos de conhecimento poderão ser de um nível um pouco menor, pois, para manter todos os serviços em funcionamento, os provedores contam com aplicativos avançados de configuração, monitoramento, segurança e atualizações.

Para finalizar, esse modelo de nuvem está aumentando gradativamente, pois muitas empresas estão podendo testar, gerenciar e simular muitas aplicações de negócios em comum.

Nuvem híbrida

O próprio nome já simplifica um pouco a explicação desse tipo de nuvem, que é uma combinação de duas ou mais infraestruturas de nuvem – pública, privada e/ou comunitária –, as quais, para a utilização e acesso dos consumidores, permanecerão como arquitetura única, porém vinculadas com outras tecnologias, livres ou proprietárias, permitindo a portabilidade de dados e aplicações. O sucesso de uma nuvem híbrida dependerá da ajuda do hypervisor e das camadas de software na nuvem. As implementações de software e serviços permitem que as instâncias migrem entre a nuvem privada e pública. Com a ajuda de serviços e softwares, aplicativos avançados podem ser criados nas plataformas pública e privada. Outra funcionalidade é o balanceamento de carga entre as nuvens, flutuações entre computação de demandas de consumo, oferecendo a essa categoria a capacidade de lidar com problemas diversos, como estouro de carga, por exemplo.

As empresas utilizam esse tipo de infraestrutura para eliminar alguns problemas quando as demais nuvens são incompatíveis, por isso sua configuração requer um conhecimento técnico maior, pois é necessário implementar servidores, redes, dispositivos de armazenamentos, cluster etc.

Nesse cenário, a virtualização de contêineres é muito utilizada para criar camadas de software em nuvem privada para o fornecimento de serviços. Ela é considerada híbrida quando possuir estes três componentes: utilizar uma infraestrutura pública, uma privada e uma WAN (Wide Area Network) para conectar as duas arquiteturas.

Seu maior desafio está na manutenção e implementação, pois, toda vez que a infraestrutura for modificada a ponto de se tornar incompatível aos usuários específicos, o custo inicial de implementação e manutenção é maior e tende a ficar cada vez mais caro.

Os modelos de serviços sugeridos pelo NIST podem ser divididos em:

- » Software como serviço (SaaS).
- » Plataforma como serviço (PaaS).
- » Infraestrutura como serviço (IaaS).

Em relação a esses três modelos, teremos capítulos específicos para cada um e resumiremos todos os seus conceitos e processos em um quarto modelo: o ITaaS (Tecnologia da Informação como um Serviço).

CAPÍTULO 2

Alinhamento estratégico e tecnologia da informação

O segmento, departamento, conceito, que sofreu o maior número de investimento e atenção das corporações foi o de TI (Tecnologia da Informação). Seus modelos de implementação sempre foram algo particular dentro de cada empresa, pois depende muito de seus níveis de atuação. Um dos modelos mais utilizados foi o departamento que faz manutenções e suporte em equipamentos, softwares, atendimento aos usuários, desenvolvimento de sistemas de gestão e até instrutor de informática. Esses departamentos foram considerados, por muito tempo, o departamento de TI da empresa.

Entretanto, o cenário alterou o relacionamento entre os departamentos de TI e a realidade das corporações. Com o advento das terceirizações, foi possível contratar serviços, equipamentos, softwares sob demanda, ou seja, os departamentos de TI foram tomando outra forma de atuar nas empresas.

Cada vez mais modelos de negócios foram surgindo e alterando a necessidade de um TI mais próximo dessas mudanças – aqui não estamos falando de mudanças físicas que acontecem rotineiramente dentro das empresas, mas sim de mudanças estratégicas, que devem ser acompanhadas pelos departamentos de TI.

O TI transita em vários níveis estratégicos das empresas, e muitas destas conseguem gerar valor com o alinhamento entre processos de negócios e TI. Porém um número muito grande de empresas acaba alocando o TI como um setor de custos, e isso se deve a vários fatores. De imediato, sem citar qualquer cenário, um dos problemas que as empresas podem encontrar em seus departamentos de TI é a dificuldade na flexibilização diante de alterações de ambiente, seja físico ou estratégico.

Esse problema acontece quando os departamentos de TI estão para “apagar incêndio”, ou seja, estão envolvidos com tarefas rotineiras, e não com planejamentos estratégicos. O cenário se converte facilmente em sistemas legados que não se comunicam, informações distorcidas e não confiáveis, problema relacionados à segurança da informação, como vírus, vazamento de informações etc., muitas relacionadas ao licenciamento de software mal gerenciado, problemas com alocações de hardware, problema de atualizações e inovações.

Há tempos a Tecnologia da Informação se tornou um conjunto de processos que participa da estratégia das organizações. Muitas empresas nascem com a cultura da Tecnologia da Informação plena, ou seja, todos os processos estão calçados 100% sob plataformas tecnológicas. As que nasceram há mais tempo foram se adaptando e se encontram em vários estágios de implantação e maturidade de implementações. As que não se adequaram aos modelos tecnológicos infelizmente deixaram de existir ou estão indo para esse caminho. É inevitável para a sobrevivência de uma empresa, nos dias atuais, estar com seus processos rodando sob as melhores práticas de TI.

Segundo Veras (2015, p. 12), um dos conceitos-chaves para a transformação da TI foi o alinhamento estratégico, que é um componente central da governança de TI e permite, quando bem feito, executar os projetos que são priorizados de acordo com a estratégia.

Alinhar estrategicamente garante que haja uma ligação entre o plano de negócios da empresa com o departamento de TI, para definir, validar a proposta diante das operações estratégicas das empresas.

Ainda em Veras (2015, p. 12), três são os planos que se completam para o alinhamento estratégico.

Planejamento Estratégico de TI (Peti)

Planejamento Estratégico de TI é um projeto, documento ou forma com que a empresa busca estruturar estrategicamente as informações organizacionais, a TI, os sistemas de informação, os sistemas de gestão do conhecimento. O Peti deve também englobar a gestão dos recursos humanos envolvidos, os demais necessários para auxiliar tomadas de decisões, ações e processo das organizações e deve ser coerente com a missão, a visão e os valores da instituição.

Por se tratar de projeto, ele deverá ser único, ou seja, não existe uma receita pronta para criar ou implantar um Peti. Sabemos que um projeto precisa obrigatoriamente possuir, algumas características mínimas como, uma data de início e término, um escopo bem definido e um limite de recurso financeiro para sua execução. Diante disso, para a criação de um Peti, devemos passar por algumas fases do PMBOK.

Segundo Souza (2018, p. 5), um Peti de sucesso deverá passar por três estágios: de preparação, diagnóstico e planejamento. O estágio de preparação representa o início do projeto de desenvolvimento do Peti e deve observar as seguintes tarefas, que compõem a fase de preparação do desenvolvimento:

- » definir um período de validade para o ITSP;

- » definir o escopo;
- » definir a equipe de construção e os participantes no ITSP;
- » definir a metodologia;
- » definir a missão e a visão da instituição;
- » definir os documentos usados como referências;
- » alinhar o ITSP com os documentos da agência.

A etapa de diagnóstico é marcada pelo esforço de entender o estado atual da TI na organização, de acordo com essa estrutura, identificando necessidades (problemas ou oportunidades) cuja solução é esperada. As seguintes tarefas formam o estágio de diagnóstico:

- » aumento da situação atual de TI do parque de TI;
- » identificação de requisitos a serem atendidos;
- » avaliação de serviços prestados;
- » inventários;
- » preenchimento de formulários de avaliação e entrevistas com prazos para respostas.

A etapa de planejamento é marcada por atender às necessidades, estabelecendo planos e ações adequados para alcançar os objetivos esperados. Para esse fim, inclui processos relacionados à priorização de necessidades e planejamento de metas e ações, cobrindo aspectos de pessoal, orçamento e risco. As seguintes tarefas formam o estágio de planejamento de TI:

- » definir ações a serem tomadas para atender a necessidades diversas;
- » definir o modo de execução para as ações;
- » planejar ações, identificando, para cada requisito listado no estágio anterior, as metas prioritárias e ações necessárias para alcançá-los;
- » planejar pessoas, identificando necessidades de RH decorrentes de atividades não abordadas no planejamento de execução;
- » planejar a execução, procurando identificar os recursos humanos e orçamentários necessários para realizar a ação planejada para atender às necessidades e estimar os prazos para o início e a conclusão das ações.

Planos Táticos de TI (PTTI)

Segundo Veras (2015, p. 13), trata dos planos de execução dos projetos prioritários e da alocação de recursos.

Plano Diretor de TI (PDTI)

Segundo Veras (2015, p. 13), é gerado depois dos Planos Estratégicos de TI e do Plano Tático de TI, é e o responsável pelo estabelecimento das diretrizes para a TI.

Alinhamento estratégico

Segundo o PMI (Project Management Institute – 2013), o alinhamento estratégico é o elo de conexão entre a estratégia e a execução, sendo descrito em quatro categorias:

Planejamento estratégico

É um processo duplo que envolve a participação e o compromisso de muitas pessoas dentro de uma organização. É preciso fazer uma avaliação do ambiente em que a empresa opera considerando sua condição macroeconômica, mudanças políticas e legislativas, sua concorrência e seus clientes e alternativas. Além disso, deve-se elencar o que exatamente a empresa precisa fazer para aumentar e melhorar suas chances de sucesso.

Comunicação da estratégia

Segundo Robert S Kaplan, professor da Harvard Business School e cocriador do balanced scorecard, “você precisa comunicar os planos aos funcionários sete vezes de sete maneiras” para que eles realmente entendam esses planos.

Sistema de gestão do desempenho

Fornece definições claras de metas e objetivos para a estratégia de uma organização. Os scorecards são as métricas mais utilizadas em um sistema de gerenciamento de desempenho.

Seleção do portfólio de projetos

Para priorizar de forma rápida e objetiva os projetos da organização que dão suporte à estratégia definida, pode-se utilizar uma matriz de priorização simples. O departamento

responsável, com auxílio do gerenciamento de projetos (PMO), responde às perguntas elaboradas para fornecer uma pontuação conforme as respostas.

Service Level Agreement (SLA)

Os acordos de nível de serviço (SLAs) têm sido um contrato comum no suporte aos serviços oferecidos por prestadores de serviços de telecomunicações por muitos anos. Entretanto, atualmente os SLAs estão sendo considerados serviços de não comunicação (rede) e estão sendo adotados interna e externamente para definir o desempenho e a qualidade acordados do serviço ou produto como parte importante do programa de gerenciamento de relacionamento com o cliente.

Alcançar metas de qualidade e desempenho para os produtos ou serviços pode exigir que a empresa estabeleça e gerencie vários SLAs. A crescente complexidade da globalização de serviços reúne uma infinidade de fornecedores e tecnologias, todos com potencial requisito de desempenho diferentes. Para uma empresa executar sua tarefa pretendida, seja oferecendo um serviço ou produto como um utilitário (por exemplo, eletricidade, água) ou como uma empresa geradora de receita responsável perante seus acionistas, as expectativas do nível de serviço a ser oferecido serão definidas entre o fornecedor de origem (o negócios, empresa ou departamento) e o usuário final (o cliente).

A Open Group disponibilizou um guia de boas práticas que pode ser utilizado na confecção de SLA, por meio do link: <https://pubs.opengroup.org/onlinepubs/9299959499/toc.pdf>. Acesso em: 12 out. 2019.



Veja alguns PETI que foram colocados em prática por diversas entidades governamentais:

» **Ministério da Ciência, tecnologia, Inovações e Comunicações**

<http://planejamentoestrategico.mctic.gov.br/biblioteca/2017-08/Plano-Estratgico-de-Tecnologia-da-Informao-2017-2019pdf.pdf>. Acesso em: 1º out. 2019.

» **Comissão de Valores Mobiliários**

http://www.cvm.gov.br/menu/acesso_informacao/planos/peti/planejamento-estrategico-de-ti.html. Acesso em: 1º out. 2019.

» **ANVISA**

<http://portal.anvisa.gov.br/documents/33900/0/PETI+-+Planejamento+Estrat%C3%A9gico+de+TI/dd1e278c-e584-4e6a-89e0-bee87c617ea8>. Acesso em: 1º out. 2019.

» **Superintendência de Tecnologia da Informação**

https://www.bnb.gov.br/documents/50268/1604820/PETI+2017-2021_VersaoPublicacao.pdf/4890504a-c359-90dd-a4c9-20a355d92663. Acesso em: 1º out. 2019.

› **INPE**

http://www.inpe.br/ti/arquivos/PETI-2015_2016.pdf. Acesso em: 1º out. 2019.

CAPÍTULO 3

Componentes de computação em nuvem

A computação em nuvem é uma tecnologia estratégica emergente, que pode ser usada tanto em grandes corporações quanto em *startups*, pois sua flexibilidade em provisionar recursos, conforme a necessidade, é fundamental. Além do mais, garante a segurança e disponibilização dos dados como nenhuma outra tecnologia inventada até o momento.

Não importa onde esteja, seja em um país distante ou em uma grande empresa, como Google, IBM, Amazon e Microsoft, a arquitetura dos datacenters será basicamente a mesma, bem como seus componentes. Elas são compostas por:

- » **Um Front End:** é uma interface entre os serviços disponibilizados pelo provedor para que os usuários possam personalizar seus serviços e suas aquisições. Suas atribuições são as mais variadas possíveis para que o torne a usabilidade atraente ao consumidor. Entre elas, podemos citar: criação de modelos de rede, criação e provisionamento de recursos.
- » **Um Back End:** é normalmente uma API na qual o Front end se conecta e controla todas as parametrizações, controles, monitoramentos, gerenciamento de segurança, armazenamento etc. No back end, também é possível encontrarmos software de gerenciamento, planos de contingência, auditoria de conformidades, softwares de implantação, que dará início aos serviços contratados pelo consumidor.
- » **Um Hypervisor:** consiste em um controlador e gerenciador de máquinas virtuais. Eles controlam os sistemas operacionais virtuais que, por sua vez, controlam os hardwares e softwares dos sistemas operacionais hospedeiros.
- » **Servidores:** são compostos por servidores físicos e servidores virtuais. Os servidores físicos são equipamentos com capacidade suficiente para abrigar implementações de vários servidores virtuais. Já os servidores virtuais são softwares instalados com o propósito de servir algum recurso, como banco de dados, servidor web, e-mail etc. Os servidores físicos também serão conhecidos como storage quando tiver como propósito servir armazenamento, que, como os demais, deverá ter fácil acesso aos meios de provisionamento de capacidade. Sua configuração e

implementação também beneficiam o usuário, fornecendo acesso remoto com a ajuda da Internet; seus serviços são muito rápidos para acessar, e as informações são mantidas, alteradas e recuperadas de um servidor de armazenamento em nuvem remoto na Web. Quando oferecido por meio do modelo IaaS, sua capacidade de armazenamento é escalável, flexível e redundante.

Os servidores podem ser divididos em clusters e possuem os seguintes componentes:

- » **Front end:** interfaces usadas para acesso externo, que pode ser acessado por servidores de aplicativos ou usuários que estão enviando tarefas ou recuperando resultados da tarefa do cluster.
- » **Nós principais:** os nós principais são responsáveis por gerenciar os nós de computação no cluster e otimizar a capacidade geral de computação. Geralmente, o nó principal é o único nó que se comunica com o mundo externo.
- » **Malha de alta velocidade de back end:** essa malha de alta velocidade é o meio principal para o nó mestre calcular as comunicações entre nós e entre os computadores.
- » **Nós de computação:** o nó de computação executa um kernel do SO otimizado ou completo e é o principal responsável por operações intensas da CPU, como processamento de números, renderização, compilação ou outra manipulação de arquivos.
- » **Path de armazenamento:** o path de armazenamento pode usar interfaces Ethernet ou Fibre Channel. As interfaces Fibre Channel geralmente se conectam a um switch SAN.
- » **Sistema de arquivos comum:** o cluster do servidor usa um sistema de arquivos paralelo comum que permite acesso de alto desempenho a todos os nós de computação. Os tipos de sistema de arquivos variam de acordo com o sistema operacional.

Virtualização

A virtualização em nuvens proporciona aos usuários consumidores a criação de diversos tipos de plataformas, como sistemas operacionais, armazenamento etc., que ajudará esse usuário a disponibilizar várias máquinas ao mesmo

tempo, conforme sua necessidade de provisionamento. Esse tipo de virtualização também propicia o gerenciamento da carga de trabalho e abstrai um ambiente completo de hardware, software e rede.

Virtualização do sistema operacional

A virtualização de um sistema operacional, além de ser diferente das máquinas virtuais, também é diferente do modo como os usuários acessam suas funcionalidades. Para cada usuário, o sistema operacional se mostra distinto, podendo ser configurado de forma independente, acessando aplicativos diversos, operacionalizando tarefas diferentes, sem interferir em outros aplicativos.

O SO trabalha somente com um kernel, e esse tipo de virtualização gera um intervalo de perda muito insignificante, o que não atrapalha a execução, num mesmo hardware, de muitos servidores virtuais isolados entre si. Porém, a desvantagem é que não permite a execução de sistemas operacionais diferentes, ou seja, *kernels* diferentes, mesmo que haja compatibilidade de bibliotecas. Nesse tipo de virtualização, o mesmo aplicativo aberto em várias instâncias não interfere no funcionamento de outro, funcionando no mesmo computador. Quem fará esse controle será o kernel, que consegue fazer cada um funcionar em seus espaços isoladamente, em ambientes chamados de contêineres de software. Sua segurança em relação aos acessos às informações é reforçada em ambientes virtuais.

Em relação ao uso do disco, é utilizado de duas formas. O disco particular é utilizado no modelo de rede privada, em que cada empresa poderá armazenar informações conforme as configurações atribuídas. Já o disco compartilhado pode ser acessado por vários clientes ao mesmo tempo. As informações são manipuladas individualmente por cada cliente, e as configurações de um não afetam as do outro.

As principais vantagens da virtualização de sistemas operacionais são:

- » elimina espaço físico utilizado pelo parque de TI local, normalmente uma sala apropriada com refrigeração, backup de energia, backup de equipamentos, racks etc.;
- » proporciona economia de energia elétrica, pelo fato de não possuir um espaço físico reservado pra isso;
- » permite implantações rápidas, conforme necessidades estratégicas da empresa.

Virtualização de servidores

A virtualização de servidores compreende em particionar um servidor físico em vários servidores virtuais, em que cada um executa seu próprio sistema operacional, aplicativos, acesso a discos, isoladamente. Esse processo é utilizado por empresas que possuem superservidores que estão sub alocados. Um software específico administra, monitora, gerencia a segurança e converte um servidor físico em vários servidores virtuais, utilizando efetivamente todos os recursos do servidor físico, como a CPU, que utiliza vários processadores, podendo convertê-los em vários núcleos, melhorando o desempenho na execução de tarefas.

Para os consumidores, o acesso e gerenciamento é feito em uma instância do servidor virtual e pode ser utilizado como servidores web, serviços de hospedagem, servidor de e-mail etc. Com a virtualização de servidores, os clientes economizam com espaços físicos, energia elétrica, provisionamento conforme necessidade, custo de hardware etc. Em relação ao desempenho, poderão ser montados projetos de balanceamento de carga, divisão dos servidores por funções (servidor web, banco de dados, e-mail) e execução de tarefas dedicadas.

Existem basicamente três tipos de virtualização de servidores em nuvem:

- » **Hypervisor:** camada localizada entre o sistema operacional do servidor físico e seu hardware. É ele quem controla toda a carga e fluxo de informações, como enfileiramento, solicitações, acesso dos sistemas operacionais virtuais à abstração de hardware. Há o hypervisor tipo 1 (bare metal), que acessa diretamente o hardware, com as máquinas virtuais funcionando diretamente sob ele, e tipo 2, que possui um software de gerenciamento que funciona diretamente no sistema operacional normal, com as VMs (virtual machine) funcionando sob essas duas camadas.
- » **Paravirtualização:** no caso da paravirtualização, o sistema operacional deverá ser alterado para que seja compatível, o que melhora muito o desempenho. Se isso não for feito, o sistema funcionará como um emulador, o que compromete o desempenho.
- » **Virtualização completa (full):** o hypervisor emula o hardware das máquinas virtuais. Palma (2008) explica como funciona o hypervisor nesse tipo de virtualização:

O sistema operacional da máquina virtual nem percebe que está rodando em um ambiente virtualizado, e tudo o que ele precisa são os *devices*

drivers para os dispositivos que o hypervisor emula. Para realizar isso, o hypervisor usa uma técnica chamada “*binarytranslation*” para executar instruções de modo *kernel* da máquina virtual, substituindo estas instruções críticas por uma sequência de instruções que simulam o comportamento esperado no hardware virtualizado.

CAPÍTULO 4

Benefícios, desafios e riscos

Como podemos observar, a terceirização de serviços em nuvem foi inevitável. E isso se deu pela evolução da conectividade de banda larga, pelo crescimento da geração de dados das empresas e pela aceitação do baixo custo de contratação e migração. Mesmo diante de cada benefício, temos que sempre questionar e efetuar mensurações periódicas para analisar se realmente o investimento empregado para migrar os serviços para a nuvem é vantajoso. Entre os vários indicadores que podemos utilizar para responder às perguntas sobre o investimento, está o ROI. Segundo Isaca (2012 p. 6), o ROI considera os custos de um investimento e seus ganhos esperados e fornece uma estimativa do quão favorável ao investimento será. Para calcular o ROI (ROI simples), o custo de um investimento deve ser subtraído do ganho (retorno) do investimento, e o resultado dividido pelo custo do investimento; o resultado é expresso como uma porcentagem ou taxa.

$$ROI = \frac{(Ganho\ do\ investimento - Custo\ do\ investimento)}{Custo\ do\ investimento}$$

Veja, por exemplo, o ROI para um serviço em nuvem com investimento estimado de R\$ 60.000,00 ao longo de um período de dois anos, com benefícios de R\$ 90.000,00.

$$ROI = \frac{(90.000,00 - 60.000,00)}{60.000,00}$$

Esse serviço, portanto, fornecerá um ROI de 50%.

Entretanto, como única medida financeira, o ROI pode não ajudar a prever o risco envolvido no investimento. Existem outros indicadores que a empresa pode utilizar, como o TCO (custo total de propriedade), que considera apenas o custo associado à aquisição durante um período; o VPL (Valor Presente Líquido), que compara os benefícios e custos previstos por um período determinado; e a TIR (Taxa Interna de Retorno), que é a taxa utilizada em alguns indicadores.

Segundo Isaca (2012, p. 7), para investimentos mais complexos, como serviços de computação em nuvem, o cálculo do ROI pode ser complexo e enganoso. A geração de um resultado relevante depende da consideração de todas as

variáveis quantificáveis e da definição de um período claro e uniforme. Benefícios intangíveis e riscos não podem ser incluídos no cálculo, a menos que a empresa seja capaz de atribuir um valor baseado em dados históricos ou estatísticos. Investimentos baseados somente em objetivos empresariais podem ser mais bem justificados usando um caso de negócios apoiado por vários indicadores financeiros.

Quando se fala em levar serviços para a nuvem, muitos entusiastas garantem vantagens e podem demonstrar cálculos de curto prazo facilmente. Porém, existem algumas variáveis que são intangíveis e é difícil incluí-las em cálculos financeiros.

A seguir, são demonstradas as variáveis tangíveis e intangíveis (ISACA, 2012, p. 8):

Tangíveis

- » **Redução de custos:** ao utilizar o recurso adequado, o custo da computação é deslocado de despesas para custos operacionais, reduzindo os custos nas seguintes áreas:
 - › Mão de obra: número de funcionários para administração do sistema de TI.
 - › Software aplicativo (somente SaaS).
 - › Compra e renovação de licenças.
 - › Suporte técnico e suporte ao usuário.
 - › Manutenção (atualizações, patches etc.).
 - › Hospedagem (instalações físicas, energia, refrigeração etc.).
- » **Maior produtividade:** mobilidade do usuário e acesso de qualquer lugar podem aumentar a produtividade.
- » **Utilização otimizada de recursos:** as empresas utilizam somente os recursos de computação de que precisam.
- » **Maior segurança/conformidade:** os provedores de serviços em nuvem podem oferecer controles de segurança robustos como um diferencial de mercado.
- » **Acesso a habilidades e recursos:** os clientes se beneficiam de habilidades e recursos de alto nível ao mesmo tempo em que evitam custos trabalhistas.

- » **Dimensionamento:** o provisionamento ou os recursos de computação sob demanda eliminam o custo do planejamento de capacidade.
- » **Agilidade:** a agilidade contribui para a redução de custos e melhoria do desempenho devido ao provisionamento mais rápido de sistemas:
 - › implantação mais rápida de aplicativos (SaaS);
 - › desenvolvimento/testes mais rápidos de aplicativos (PaaS).
- » **Satisfação dos clientes:** a utilização eficaz de aplicativos em nuvem pode aumentar a colaboração entre a empresa e seus clientes ou reduzir o tempo de resposta para solicitações dos clientes.
- » **Confiabilidade:** os provedores de serviços em nuvem têm sites redundantes que podem fornecer continuidade dos negócios e recuperação de desastres de forma mais eficiente.
- » **Desempenho:** o monitoramento contínuo e uniforme das operações pelo provedor de serviços em nuvem pode resultar em melhor desempenho e maior tempo de operação.

Intangíveis

- » **Menos oportunidades de negócios perdidas:** um aplicativo em nuvem (SaaS) pode ser o elemento crucial para o estabelecimento de uma nova empresa ou a expansão para novos mercados.
- » **Foco no negócio principal:** os recursos de TI podem ser alocados para suportar as funções do negócio principal.
- » **Satisfação dos funcionários/ inovação:** mobilidade e desempenho mais rápidos podem melhorar a satisfação dos funcionários e promover a inovação.
- » **Colaboração:** colaboração em tempo real pode aumentar a qualidade e a inovação.
- » **Transferência de riscos:** alguns riscos podem ser transferidos para o CSP (provedor de serviços em nuvem – por exemplo, violações de segurança, perda de dados, recuperação de desastres); isso pode representar um benefício tangível ou intangível.

Existem três tipos de custos que as soluções em nuvem incluem, além dos custos óbvios de software e hardware: custos iniciais (investimento inicial), custos operacionais (custos recorrentes) e custos únicos (custos de alteração ou de rescisão).

Custos iniciais

- » **Prontidão técnica:** investimento em banda larga para alocar a nova estrutura, por exemplo.
- » **Implementação e integração:** serviços profissionais podem ser necessários para gerenciar ou integrar a transição para a nuvem.
- » **Treinamento:** recursos de TI podem necessitar de treinamento para gerenciar fornecedores e serviços em nuvem. Os usuários podem precisar de treinamento em novos aplicativos.
- » **Mudança organizacional:** os processos podem precisar de reengenharia para acomodar as necessidades específicas da nuvem.

Custos recorrentes

- » **Taxas de Assinatura:** são cobradas periodicamente para utilização dos serviços em nuvem.
- » **Gestão de Fornecedores:** custos de atividades de monitoramento e execução de SLAs.
- » **Suporte ao usuário final e administração:** são custos que podem ser cobrados junto com a taxa de assinatura ou podem ser pagos diretamente pela empresa à parte.
- » **Mitigação de risco:** são medidas que talvez devam ser tomadas para controle de riscos relacionados à computação em nuvem.
- » **Redimensionamento:** alguns fornecedores podem realizar uma cobrança extra por redimensionamentos necessários referentes aos recursos de computação.

Custos de rescisão

- » **Utilizar os serviços internamente ou realizar a mudança para um provedor diferente:** caso a empresa necessite voltar a um modelo interno devido a regulamentações ou problemas

econômicos que inviabilizem a utilização dos serviços em nuvem, é possível que essa decisão tenha alguns custos, entre eles:

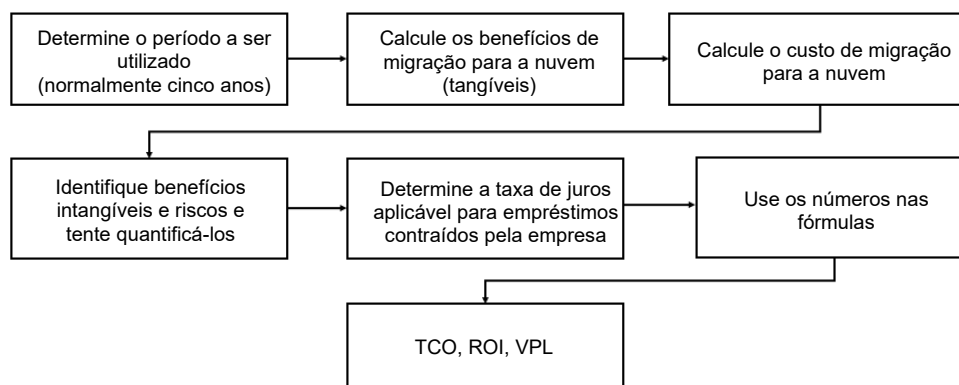
- › extração e exclusão de dados da nuvem e hardware de processamento;
- › validação da precisão e integridade dos dados;
- › configuração dos sistemas internos para receber e armazenar os dados que estavam em nuvem;
- › multas contratuais devido à rescisão antecipada;
- › realocação de recursos de TI para dar suporte aos serviços que serão realizados internamente;
- › compra de recursos físicos para hospedar os serviços que estavam em nuvem.

Principais desafios ao utilizar a nuvem:

- » **Incompatibilidade:** os serviços em nuvem podem não atender às necessidades da infraestrutura de TI e dos sistemas existentes.
- » **Tempo de Operação:** os provedores de serviços em nuvem podem não ser capazes de garantir o tempo de operação contratado ou podem sofrer instabilidades devido ao serviço de internet da própria empresa.
- » **Desempenho:** se a capacidade do serviço e a velocidade da internet não forem bem planejadas, o desempenho pode cair ao longo do tempo.
- » **Segurança:** a computação em nuvem representa riscos tradicionais e novos que precisam ser considerados em sua escolha.
- » **Conformidade:** a natureza abstrata da nuvem pode causar a não conformidade com as expectativas da empresa.
- » **Pagamento progressivo:** a empresa deve implementar controles para evitar cobranças excessivas que podem ocorrer quando os sistemas permanecem conectados.
- » **Falta de liberdade:** a empresa pode ficar presa a uma tecnologia específica ou a um fornecedor específico, o que pode impedir a portabilidade.

- » **Consumo excessivo de serviços em nuvem:** para evitar que isso aconteça, a empresa deve adaptar uma estrutura de governança de forma que somente a área de TI possa controlar a compra de serviços em nuvem.
- » **Personalização limitada:** pode haver dificuldade de personalização ou modificação dos aplicativos em nuvem sempre que mudanças ocorrerem, o que pode gerar novos custos ou limitar processos de adequações importantes para a empresa.

Figura 1. Sequência para atingir o ROI.



Fonte: Isaca (2012, p. 14).

Segundo Isaca (2012, p. 15), os pontos a seguir devem ser considerados ao aplicar essa abordagem:

Fase 1

- » Começa pela avaliação de um modelo de nuvem simples e econômico. Assim, é possível facilitar a execução de uma prova de conceito e entender melhor os recursos, benefícios e riscos do modelo.
- » Caso a empresa esteja certa do seu modelo de nuvem ideal, pode pular as etapas da fase 1, porém é preciso que o ROI do modelo ideal já tenha sido estimado.
- » Compreender o risco relacionado aos serviços em nuvem pode ser desafiador devido à imensa variedade de serviços oferecidos, à falta de transparência dos controles e às dificuldades de comparação entre os fornecedores.

Fase 2

- » Com a conclusão da avaliação completa do modelo atual, essa fase pode ser relativamente rápida, dependendo de quanta documentação está disponível para avaliação. É preciso atenção para não deixar de lado possíveis áreas de risco desconhecidas, o que levaria a empresa a subestimar muitos custos da opção atual e/ou os benefícios da opção alternativa.

Fase 3

- » Quando uma empresa migra para a nuvem, redireciona grande parte de seus custos operacionais de TI para a gestão da nuvem e de seus riscos. Isso pode representar custos diferentes entre a opção atual e a opção alternativa.

Antes de iniciarmos os estudos sobre os tipos de serviços em nuvem, oferecidos pela maioria das empresas, veja a tabela a seguir onde se encontra cada recurso. Além disso, também precisamos estar alinhados sobre o conceito de serviços. Segundo Veras (2015, p. 22), serviços é um benefício que uma organização entrega para outra e, no caso do TI, quando ela suporta o processo de negócios, ela está provendo um serviço para a organização. As características dos serviços são:

- » agregam algum tipo de valor;
- » são intangíveis;
- » são produzidos e consumidos ao mesmo tempo;
- » são produzidos pela atuação organizada de um conjunto de processos.

Quadro 1. Hierarquia de serviços em nuvem.

SaaS					
	PaaS				
			IaaS		
Apps/Aplicativos Hospedados	Ferramentas de desenvolvimento, gerenciamento de banco de dados, análise de negócios	Sistemas operacionais	Servidores e armazenamentos	Segurança/Firewall de rede	Construção/ Planta física do data center
Google Apps	Google AppEngine		Amazon EC2		
Salesforce	Microsoft Azure		Eucalyptus		
MS Cloud Services	Force.com		Open Nebula		

Fonte: adaptada de <https://azurecomcdn.azureedge.net/cvt-add6f4a23186e176ede2ce0357a4c26bb9352332c93c37a18e80bd3245a301b9/images/page/overview/what-is-paas/what-is-paas.png>. Acesso em: 27/2/2020.

CAPÍTULO 1

Infraestrutura como Serviço – IaaS

Segundo Veras (2015, p. 133), Infraestrutura como Serviços, ou IaaS (Infrastructure as a Service), é a capacidade que o provedor tem de oferecer uma infraestrutura de processamento e armazenamento de forma transparente para o cliente, em que esses usuários não têm o controle da infraestrutura física, mas que, por meio de interfaces de virtualização, possuem controle sobre máquinas virtuais, armazenamento, aplicativos instalados e controles limitados de recurso de redes.

Uma IaaS deverá possuir uma interface amigável, na qual clientes possam controlar seu ambiente de infraestrutura sem precisar gerenciá-la fisicamente no local. Essa interface configura repositórios de recursos computacionais físicos, como processador, memória, armazenamento e redes, e entregam modelo de negócios compartilhado, que possibilita atender múltiplos usuários. Ele poderá fazê-lo por meio de um dashboard, API ou CLI (Command Line Interface). Algumas das empresas mais conhecidas que fornecem IaaS são Amazon AWS, Microsoft Azure e IBM.

O serviço deverá possuir componentes que possam ser adquiridos e provisionados separadamente, deixando a cargo do cliente a possibilidade de provisioná-lo conforme necessidade de aumentar ou diminuir para que não precise pagar por custos extras. Nesse modelo, cabe ao provedor a manutenção física dos servidores, e ao cliente a instalação de software, como sistemas operacionais, banco de dados, aplicativos em geral. Os componentes são distribuídos em *Datacenters* que, por sua vez, estão protegidos por um provedor de serviços.

Algumas vantagens de trabalhar com IaaS são:

- » Redução de aquisição de hardware, sua depreciação e, consequentemente, seus upgrades, segurança e manutenção.
- » Liberação de espaço físico na empresa.
- » Provisionamento facilitado para ampliar ou reduzir a capacidade de processamento e/ou armazenamento.
- » Os gestores de TI poderão ficar mais focados em processos de negócios da empresa e não em se preocuparem com eventuais situações que uma infraestrutura demanda, como manutenção, atualização, solução de problemas etc.

- » Alocação inicial de investimentos em infraestrutura e projeções eventuais crescimentos de necessidades futuras.
- » Com a elasticidade de provisionamento de recursos, fica fácil garantir uma evolução tecnológica de acordo ao crescimento da empresa.
- » Suporte, atualizações de hardware, garantia de funcionamento, garantia de rede, de backup fica tudo a cargo do provedor.
- » Facilidade de disponibilizar novas redes ou novos servidores em minutos.
- » A disponibilização do serviço, normalmente, é 24/7 (vinte e quatro horas por dia e sete dias por semana).
- » A nuvem está disponível por 24 horas e pode ser acessada de qualquer lugar, o que a torna escalável.
- » A economia é evidente, pois o pagamento do aluguel dos recursos é feito conforme o uso.

Projeto de IaaS

Uma empresa deverá decidir se levará seus serviços de infraestrutura para a nuvem, qual o modelo de serviço adotar e qual o recurso mínimo que necessita por meio de um projeto bem elaborado. Uma das vantagens da infraestrutura em nuvem em relação a local é que, na nuvem, podemos iniciar alocando requisitos mínimos e aumentando conforme a necessidade. Se fosse montar uma infraestrutura local, o cálculo para adquirir processamento, armazenamento ou memória já teria que ser feito para os próximos dez anos, por exemplo. Nesse caso, a empresa paga hoje um recurso que chegará em sua carga total somente daqui a 10 anos.

Utilizaremos um cenário em outros tipos de serviços, porém é preciso que o departamento de TI saiba de todos os recursos que serão utilizados no geral. Nesse momento, precisaremos saber quais são os requisitos de funcionamento dos softwares que irão se utilizar da infraestrutura. Os usuários terão acesso aos aplicativos por meio do recurso de Desktop Remoto.

Vamos criar um cenário comum para empresas de médio porte, que precisam de uma infraestrutura para alocar os seguintes itens:

- » um ERP, com acesso a banco de dados utilizado por 50 usuários;

- » acesso a quarenta estações de trabalho, com sistema operacional Microsoft Windows;
- » acesso a dez laptops com sistema operacional Microsoft Windows;
- » um servidor de rede que autentique os usuários e rode o banco de dados;
- » cinco licenças da ferramenta de Desenvolvimento Visual Studio;
- » 50 licenças do pacote MS Office 365;
- » 50 licenças do Antivírus Kaspersky Endpoint Security Cloud.

Veja a descrição dos requisitos no quadro a seguir:

Quadro 2. Lista de recursos.

Recursos			Requisito mínimo para 50 usuários		
Recurso	Descrição/versão	Recursos extras	Proc.	Mem.	H.D. (SSD)
ERP	Microsiga TOTVS/12	Licenças para 50 usuários	3GHz	16MB	5GB
S.O. de servidor	MS/Win Server 2019	Active Directory, Remote Desktop, DNS	3GHz	32MB	15GB
Gerenciador de banco de dados	PostgreSQL 10		3GHz	16MB	5GB
Office	MS Office 365	Word, Excel, Power Point	3GHz	16MB	5GB
Antivírus			3GHz	16MB	5GB

Fonte: Elaborada pelo autor.

O próximo passo é mensurar a quantidade de recursos necessários para a infraestrutura. Para isso, devemos estimar uma carga atentando-se aos recursos que tendem a ter o seu crescimento conforme o tempo passa, ou seja, quanto mais forem utilizados durante um período, mais serão consumidos. No caso do exemplo anterior, temos o SGDB (Sistema Gerenciador de Banco de Dados), que será alimentado constantemente.

O provisionamento de recursos computacionais, segundo Dominico (2016, p. 5), busca garantir a alocação eficiente de recursos como CPU (Central Processing Unit), armazenamento (memória RAM e disco rígido) e largura de banda de rede. Podemos considerar que uma alocação eficiente dos recursos acontece quando não existem desperdícios de recursos ou quando não há falta de recursos para assegurar a qualidade de serviço.

O provisionamento é subdividido em duas técnicas:

- » Estático: os recursos são provisionados pelo administrador/programador de forma estática e alterada em longos períodos de tempo.
- » Dinâmico: os recursos são alocados, conforme a demanda em tempo de execução.

O provisionamento dinâmico pode ser utilizado sob duas abordagens:

- » Reativo: uma abordagem orientada a recursos, com um conjunto de regras pré-definidas, baseadas em regra-condição-ação. São adotados thresholds superiores e inferiores que indicam quando adicionar e remover recursos (DOMINICO, 2016, p. 5).
- » Segundo Cecchet *et al.* (2011, p. 6), proativo ou preditivo: para o provisionamento de recursos, é realizado um reconhecimento nos padrões da carga de trabalho, antecipando situações negativas para o sistema, para estimar a quantidade de recursos necessária.

A escolha da métrica a ser observada é fator importante para um provisionamento eficiente. Por meio desta, é possível analisar o comportamento do sistema. Segundo Dominico (2016, p. 6), uma métrica define qual é o objetivo do estudo realizado. Essas métricas podem avaliar, por exemplo, a carga de uso de instâncias de VM (Virtual Machine), a carga de uso de CPU, o gasto de energia, a memória, entre outros.

CAPÍTULO 2

Plataforma como Serviço – PaaS

PaaS (Plataforma como serviço) é um ambiente que disponibiliza ferramentas de desenvolvimento e implantação completo na nuvem. Os serviços incluem a possibilidade da instalação de ferramentas de versionamentos, IDEs de desenvolvimentos, serviços de Business Intelligence, sistema gerenciador de banco de dados, contêineres etc. A ideia do PaaS é se dedicar tão somente à arquitetura de desenvolvimento da empresa. Para isso, ela deverá oferecer serviços de IaaS como a possibilidade de escalonar os recursos de hardware, como memória, processamento e armazenamento, e os serviços de Sistemas Operacionais.

O PaaS é muito utilizado para as empresas que estão implantando o conceito de DevOps, pois garante uma flexibilização entre os desenvolvedores e o time de operações, além de implementar a possibilidade de automações necessárias para manter o ciclo de vida do software. Falando em ciclo de vida de desenvolvimento de software, o PaaS garante as etapas de compilação, teste, implantação, gerenciamento, integrações etc.

Os ambientes de desenvolvimento poderão ser os mais variados possíveis, por isso as empresas que forem oferecer essas ferramentas deverão estar prontas para atender a essa demanda diversificada.

Esses serviços prestados por provedores caminharam de encontro com a necessidade das empresas de integrar os serviços de TI e o Desenvolvimento. Nos próximos tópicos, serão demonstrados exemplos de conjuntos de ferramentas de desenvolvimento oferecidas em forma de PaaS, pelas principais grandes empresas, baseando-se nos conceitos do DevOps. Como vimos em estudos anteriores, o DevOps não é um framework, que vem com conceitos fechados ou prontos. Entretanto, mesmo sendo flexível, conceitos genéricos acompanham toda a implantação, nos mais variados segmentos de negócios.

Por não se tratar de um pacote específico, como a IaaS, o PaaS poderá promover uma concorrência entre os provedores e, obviamente, quem sai ganhando é o consumidor. Atrairá mais consumidores aquele provedor que oferecer aos desenvolvedores, engenheiros e arquitetos de software, APIs que forem mais atraentes em sua usabilidade e prover elasticidade, escalabilidade, variedade de ferramentas etc. Os desenvolvedores de determinada empresa poderão precisar

de linguagens de programação específica, de sistema de bancos de dados ou de compiladores somente.

Conjunto de ferramenta PaaS

A seguir, serão listadas algumas das ferramentas, das principais empresas, disponíveis no tipo PaaS:

Google Cloud Platform

O GCP, como o próprio site da Google informa, oferece os seguintes produtos: <https://cloud.google.com/products/#ferramentas-do-desenvolvedor>.

Cloud Code

Inclui ferramentas que ajudam a escrever, implantar e depurar aplicativos nativos da nuvem de maneira rápida e fácil. Fornece extensões para ambientes de desenvolvimento integrado, como Visual Studio Code e IntelliJ, para interagir, depurar e implantar códigos rapidamente no Kubernetes. Segundo a Red Hat (<https://www.redhat.com/pt-br/topics/containers/what-is-kubernetes>), Kubernetes é uma plataforma open source que automatiza as operações dos contêineres Linux. Essa plataforma elimina grande parte dos processos manuais necessários para implantar e escalar as aplicações em contêineres. Em outras palavras, se você deseja agrupar em clusters os hosts executados nos contêineres Linux, o Kubernetes ajudará a gerenciar esses clusters com facilidade e eficiência. Esses clusters podem incluir hosts em nuvens públicas, privadas ou híbridas. Originalmente, o Kubernetes foi criado e desenvolvido pelos engenheiros do Google.

Cloud Build

Permite criar softwares rapidamente em todas as linguagens, controle total sobre as definições de fluxos de trabalho personalizados para criação, teste e implantação em vários ambientes, como VMs, ambientes sem servidor, Kubernetes ou Firebase.

Cloud Source Repositories

Um repositório Git completo, dimensionável e privado. Amplie o fluxo de trabalho do Git conectando-se a outras ferramentas do GCP, como Cloud Build, AppEngine, Stackdriver e Cloud Pub/Sub.

CloudScheduler

É um programador de cronjobs totalmente gerenciado e de nível empresarial que atua como uma interface para que você gerencie todas as tarefas de automação em um só lugar. Com ele, é possível programar praticamente qualquer job, incluindo jobs em lote e de Big Data, operações de infraestrutura na nuvem.

CloudTasks

É um serviço gerenciado que permite administrar a execução, o envio e a entrega de várias tarefas distribuídas. Com ele, é possível executar o trabalho de maneira assíncrona fora da solicitação de um usuário ou de serviço a serviço.

Cloud Tools for PowerShell

Utilizado para criar scripts, automatizar e gerenciar cargas de trabalho do Windows executadas no Cloud Platform.

Tekton

É uma estrutura de software livre nativa do Kubernetes flexível para criar sistemas de integração e entrega contínuas (CI/CD). Permite criar, testar e implantar em vários provedores de nuvem ou sistemas locais abstraindo os detalhes subjacentes da implementação.

Firebase Test Lab

É uma infraestrutura para testar um app para Android ou iOS em uma grande variedade de dispositivos e configurações, e ver os resultados, incluindo registros, vídeos e capturas de tela, no Console do Firebase.

Cloud Tools for Visual Studio

É um ambiente poderoso para criar aplicativos Windows e .NET e implantá-los no Google Cloud Platform diretamente do seu IDE. Execute e teste seu aplicativo localmente e depois implante-o na nuvem a partir do Visual Studio.

A Amazon Web Services (AWS)

É a plataforma de nuvem da Amazon e oferece mais de 165 serviços completos de datacenters em todo o mundo. Clientes de variados tipos, incluindo startups, grandes

corporações e órgãos governamentais, utilizam a AWS para reforçar sua infraestrutura, aumentar sua agilidade e reduzir seus custos. Para desenvolvedores, a AWS oferece alguns serviços, como os citados a seguir.

Amazon Corretto

É uma distribuição gratuita, multiplataforma e pronta para produção do Open Java Development Kit (OpenJDK). O Corretto inclui suporte de longo prazo para fornecer aprimoramentos de performance e correções de segurança, é certificado como compatível com o padrão Java SE. Com ele, é possível desenvolver e executar aplicativos Java em sistemas operacionais populares, incluindo o Linux, o Windows e o macOS.

AWS CloudDevelopment Kit (AWS CDK)

É uma estrutura de desenvolvimento de software de código aberto para modelar e provisionar seus recursos de aplicativos de nuvem usando linguagens de programação conhecidas.

AWS Cloud9

É um Integrated Development Environment (IDE – Ambiente de desenvolvimento integrado) que permite escrever, executar e depurar código usando apenas um navegador; inclui um editor de código, um depurador e um terminal; é fornecido com ferramentas essenciais para linguagens de programação comuns, incluindo JavaScript, Python e PHP, entre outras, para que você não precise instalar arquivos ou configurar a máquina de desenvolvimento para iniciar novos projetos.

AWS CodeBuild

É um serviço de integração contínua totalmente gerenciado que compila o código-fonte, realiza testes e produz pacotes de software prontos para implantação e não é necessário provisionar, gerenciar e escalar seus próprios servidores de compilação.

AWS CodeCommit

É um serviço de controle de origem totalmente gerenciado que hospeda repositórios protegidos baseados em Git. Ele permite que equipes tenham facilidade para colaborar nos códigos em um ecossistema seguro e altamente escalável. O Code Commit elimina a necessidade de operar seu próprio sistema de controle de origem ou a preocupação com a escalabilidade da infraestrutura; funciona perfeitamente com suas ferramentas Git existentes.

AWS CodeDeploy

É um serviço gerenciado de implantação que automatiza implantações de software em diversos serviços de computação como Amazon EC2, AWS Fargate, AWS Lambda e servidores locais. O AWS CodeDeploy facilita o lançamento rápido de novos recursos, ajuda a evitar tempo de inatividade durante a implantação de aplicativos e lida com a complexidade de atualizá-los, automatiza implantações de software e elimina a necessidade de operações manuais propensas a erros.

AWS CodePipeline

É um serviço gerenciado de entrega contínua que ajuda a automatizar pipelines de liberação para oferecer atualizações rápidas e confiáveis de aplicativos e infraestruturas, automatiza as fases de compilação, teste e implantação do processo de liberação sempre que ocorre uma mudança no código, de acordo com o modelo de liberação que você definiu.

A AWS CommandLine Interface (CLI, interface da linha de comando)

É uma ferramenta unificada para o gerenciamento de seus serviços da AWS. Com apenas uma ferramenta para fazer o download e configurar, você poderá controlar vários serviços da AWS pela linha de comando e automatizá-los usando scripts.

AWS CodeStar

Permite que você desenvolva, compile e implante rapidamente aplicativos na AWS, disponibiliza uma interface de usuário unificada, permitindo que você gerencie facilmente suas atividades de desenvolvimento de software em um só lugar. É possível configurar toda a sua cadeia de ferramentas de entrega contínua em questão de minutos, possibilitando começar o lançamento de códigos mais rapidamente.

AWS X-Ray

Ajuda desenvolvedores a analisar e depurar aplicações distribuídas de produção, como as criadas usando uma arquitetura de microsserviços. É possível entender a performance de aplicativos e de seus serviços subjacentes para identificar e solucionar problemas e erros de performance, e disponibiliza uma visualização completa sobre as solicitações, conforme elas percorrem o aplicativo, além de mostrar um mapa dos componentes subjacentes do aplicativo.

AWS DeviceFarm

É um serviço de testes que permite testar e interagir com aplicativos Android, iOS e da web em vários dispositivos ao mesmo tempo ou reproduzir problemas em um dispositivo em tempo real. Permite que você teste seu aplicativo em uma frota compartilhada de mais de 2.500 dispositivos ou em seu próprio laboratório de dispositivos privados na nuvem.

Microsoft Azure

A Microsoft é outra gigante que entrou nos negócios de Cloud Computing. Além de oferecer todos os serviços necessários para um ambiente de TI completo, também conta com uma variedade de ferramentas para desenvolvedores, como veremos a seguir.

Visual Studio

O IDE (ambiente de desenvolvimento integrado) projetado para criar, depurar e monitorar aplicativos de nuvem localmente e implantá-lo no ambiente de produção usando ferramentas de integração avançadas.

SDKs Unificados

Os SDKs Unificados do Azure são coleções de bibliotecas para .NET, Java, JavaScript e Python. Eles são criados em um núcleo comum para facilitar o uso dos serviços do Azure, com foco na consistência, familiaridade e idiomaticidade da linguagem.

Código do Visual Studio

Compile e implante aplicativos para várias plataformas usando qualquer uma das centenas extensões e temas que ajudam a integrar seus aplicativos aos serviços do Azure e crie modelos para o AzureResource.

AzureLab Services

Configure e forneça com facilidade acesso sob demanda às VMs (máquinas virtuais) pré-configuradas para dar suporte aos seus cenários.

Kit de ferramentas do Azure para Eclipse

Use o Azure Toolkit for Eclipse para criar, desenvolver, configurar, testar e implantar facilmente trabalhos do Spark e aplicativos Web Java EE e Spring Boot altamente

disponíveis e escalonáveis no Azure usando o ambiente de desenvolvimento do Eclipse em execução no macOS, Linux e Windows.

Azure DevOps

Planeje de forma mais inteligente, colabore melhor e entregue mais rapidamente com um conjunto de serviços modernos de desenvolvimento.

Azure Pipelines

Crie, teste e implante continuamente em qualquer plataforma e nuvem.

Heroku

O Heroku é uma plataforma como serviço (Paas) totalmente gerenciada e baseada em nuvem para criar, executar e gerenciar aplicativos. A flexibilidade e a experiência projetada da plataforma oferecem suporte ao estilo de desenvolvimento preferido de cada profissional e ajudam a manter o foco e a produtividade. Permite que as empresas criem, entreguem, monitorem e dimensionem aplicativos – somos o caminho mais rápido para ir da ideia ao URL, contornando todas essas dores de cabeça na infraestrutura.

CAPÍTULO 3

Software como Serviço – SaaS

Adquirir software como serviço permite que os usuários utilizem aplicativos em nuvem, por meio da internet. Também conhecido como software como demanda, muitas empresas utilizam desses recursos para software de escritório, mensagens, software de gestão etc. A maioria das soluções SaaS utiliza a Arquitetura Multitenant que é reconhecida por ter apenas uma aplicação e uma base de dados para todos os clientes, separando-os apenas com chaves e relacionamentos no banco de dados. Apesar de parecer menos segura e escalável, a arquitetura Multitenant é o melhor caminho para a grande maioria das aplicações SaaS. Isso porque ela oferece uma maneira fácil de iniciar e manter seu software. Muitas empresas estão optando por distribuir suas aplicações por computação em nuvem por vários motivos, como custo de mídia de instalação, facilidade de atualização, distribuição rápida etc.

Nesse modelo, o provedor oferece aos clientes acesso, em rede única, de uma cópia de aplicativo criado para esse propósito. Normalmente, o código-fonte do aplicativo é o mesmo para todos os clientes e, quando novos recursos ou funcionalidades são lançados, são lançados para todos. Os dados poderão ser armazenados localmente na nuvem ou no cliente, tudo depende do projeto e do SLA.

Os fornecedores de software como serviço estão desenvolvendo e gerenciando seus próprios aplicativos. Atualmente, as soluções desse tipo de software dependem da internet e precisam de um navegador da web para acessá-lo. A solução SaaS utiliza basicamente a arquitetura, na qual o aplicativo atende a vários negócios e usuários e mantém os dados de acordo.

Projeto SaaS

Voltando à lista de recursos que utilizaremos no projeto iniciado no capítulo 1, temos:

- » um ERP, com acesso a banco de dados utilizado por 50 usuários;
- » acesso a quarenta estações de trabalho, com sistema operacional Microsoft Windows;
- » acesso a dez Laptops com sistema operacional Microsoft Windows;
- » um servidor de rede que autentique os usuários e rode o banco de dados;

- » cinco licenças da ferramenta de Desenvolvimento Visual Studio;
- » 50 licenças do pacote MS Office 365;
- » 50 licenças do antivírus Kaspersky Endpoint Security Cloud.

Para projetos locais, primeiramente instalamos o sistema operacional, e o SGDB é instalado sob ele. Em nosso projeto, poderíamos fazer da mesma forma, entretanto não seria o modelo SaaS. Para que pudéssemos utilizar essa arquitetura, precisaríamos ver qual das empresas oferecem o PostgreSQL.

Na Azure, da Microsoft, podemos acessar todas as ferramentas pelo link <https://azure.microsoft.com/pt-br/services/postgresql/>. Além da ferramenta padrão, a Azure oferece alguns benefícios que farão com que o time de desenvolvimento fique focado no negócio. Veja a lista:

- » Integração com recursos importantes do PostgreSQL, incluindo JSONB, compatibilidade geoespacial, a indexação avançada e dezenas de extensões.
- » Escala horizontal de alto desempenho no PostgreSQL usando a Hiperescala (Citus), agora em versão prévia.
- » Recomendações de desempenho inteligente geradas de uma análise personalizada do seu banco de dados.
- » PostgreSQL totalmente gerenciado com o Azure IP Advantage e a Proteção Avançada contra Ameaças do Azure.

Na rede da Amazon, a AWS, podemos acessar todas as ferramentas pelo link https://aws.amazon.com/pt/rds/?nc2=h_ql_prod_db_rds.

O Amazon Relational Database Service (Amazon RDS) facilita a configuração, a operação e a escalabilidade de bancos de dados relacionais na nuvem. O serviço oferece capacidade econômica e redimensionável e automatiza tarefas demoradas de administração, como provisionamento de hardware, configuração de bancos de dados, aplicação de patches e backups. O Amazon RDS está disponível em vários tipos de instância de banco de dados, com otimização para memória, performance ou E/S, bem como oferece seis mecanismos de bancos de dados comuns, incluindo, entre outros, o PostgreSQL.

Já o Google oferece o acesso às ferramentas pelo link <https://cloud.google.com/sql/>.

O Cloud SQL é um serviço de banco de dados totalmente gerenciado que facilita a configuração, a manutenção, o gerenciamento e a administração de bancos de dados relacionais PostgreSQL, MySQL e SQL Server na nuvem. Oferece altos níveis de desempenho, escalabilidade e praticidade. Hospedado no Google Cloud Platform, disponibiliza uma infraestrutura de banco de dados para aplicativos executados em qualquer lugar.

Quanto às licenças dos antivírus, devemos encontrar soluções especializadas e alinhá-las à política de segurança da empresa, analisando a melhor solução. Uma opção é o antivírus Kasperky, que já possui um produto próprio para nuvem.

CAPÍTULO 4

Tecnologia da Informação como Serviço – ItaaS

Para entendermos, de fato, o que significa TI como serviço, temos que primeiramente definir todos os conceitos dos tipos de serviços oferecidos pelas empresas e também o conceito de Tecnologia da Informação. Para isso, juntaremos todos os conceitos aprendidos até agora.

Primeiramente, definimos o que é Tecnologia da Informação e descrevemos, resumidamente, que TI é tudo relacionado ao parque tecnológico, como hardware, redes e software. Além disso, descrevemos o significado de IaaS, PaaS e SaaS e ainda demos exemplo de um cenário. Entretanto, para explicarmos o que é IT as a Services, devemos agrupar todos os conceitos e pesquisar qual serviço caberá no componente que será transportado para a nuvem.

Segundo Veras (2015, p. 8), existem quatro grandes partes na TI: os sistemas de informações, a arquitetura, a infraestrutura e a gestão.

Sistemas de Informações

Segundo Beal (2004, p. 15), um sistema de informações é um conjunto de componentes que interagem entre si para atingir seu objetivo e possuem mecanismo de entrada, processamento e saída. A entrada corresponde aos dados capturados, o processamento transforma os dados de entradas em saídas úteis. Um sistema poderá fazer parte de um sistema maior e ser denominado como subsistema.

Arquitetura de TI

Veras (2015, p. 20) afirma que a arquitetura de TI deve ser pensada baseada na estrutura empresarial. Enquanto a lógica da estrutura empresarial cuida dos processos de negócios, a de TI, entre outras coisas, pode tratar das arquiteturas de processos de negócio, das arquiteturas de dados, das arquiteturas das aplicações e da arquitetura da infraestrutura.

No início, a arquitetura de TI era centralizada em mainframes. Segundo a IBM, um mainframe é o repositório central de dados, ou hub, no centro de processamento

de dados, antigo CPD de uma corporação, ligado aos usuários por meio de dispositivos menos potentes, como estações de trabalho ou terminais burros. A presença de um mainframe geralmente implica uma forma centralizada de computação, e a centralização dos dados em um único repositório de mainframe evita que os clientes precisem gerenciar atualizações para mais de uma cópia dos dados comerciais, o que aumenta a probabilidade de os dados estarem atualizados.

Em seguida, houve a popularização da computação e ela precisaria atingir as empresas de pequeno porte. Como isso ficou inviável com os mainframes, a tecnologia se reinventou com a computação Cliente/Servidor. Nesse tipo de computação, os recursos foram distribuídos, com clientes tendo maior poder de processamento e armazenamento, liberando os servidores desse trabalho e tornando os preços mais acessíveis. Entretanto, nesse período houve um enorme crescimento na capacidade da humanidade gerar informações e seu controle; no formato cliente/servidor, seria impossível. Foi então que um “novo” tipo de computação foi criado: a centralização.

Parece irônico, mas o retorno para a computação centralizada foi necessário, porém em um formato diferente. Na computação centralizada de antigamente, os limites físicos eram uma das barreiras para a expansão das redes. Entretanto, na computação centralizada atual, isso não é problema. Com a Internet cada vez mais veloz, e as empresas cobrando da TI um alinhamento estratégico mais eficaz, surge um formato de computação que está conseguindo suprir essas necessidades: a **computação em nuvem**.

Infraestrutura de TI

Uma infraestrutura de TI é composta por todos os elementos necessários para criar, distribuir e manter a informação disponível, confiável e segura. Essa infraestrutura, normalmente é composta por hardware (computadores, smartphones, tablete, switches, roteadores etc.), software (sistemas de informação, clientes de e-mail, antivírus, pacotes de escritório etc.) e pessoas (usuários operacionais, estratégicos, táticos etc.).

Não há nenhuma dúvida de que o papel mais de uma infraestrutura de TI é a informação. Ela deve estar pronta para gerar corretamente importante as informações por meio dos sistemas gerenciais, ou seja, os equipamentos devem estar com sua capacidade de processamento, armazenamento e acesso à rede em perfeito funcionamento. Para manter a informação, é necessário que o armazenamento, o backup e a restauração sejam testados frequentemente

e estejam em pleno funcionamento, definidos em políticas de segurança da informação (PSI) das empresas.

A disponibilização das informações deverá ser tratada com um cuidado mais rigoroso. Para ser distribuída, a informação deverá estar armazenada em local seguro e confiável – seguro em relação a perdas, acessos restritos e alterações não autorizadas. Seu meio de distribuição deverá garantir que ela esteja disponível de forma segura, rápida e íntegra, sempre que for requisitada. Ao final, os usuários deverão estar preparados para gerar, manipular e entender as diversas opções que as informações podem oferecer, como relatórios gerenciais, tomada de decisões, orçamentos etc.

Gestão de TI

Entende-se por Gestão de TI o monitoramento e a administração dos sistemas de tecnologia da informação de uma organização, relacionado a hardware, software, redes, pessoas e informação. Ela se concentra em como fazer a informação ser útil para a empresas, por meio de seus ativos, sempre em bom estado de funcionamento. Os ativos de TI são os responsáveis por fazer a informação trafegar de forma íntegra, segura, confiável e disponível, desde seu emissor até seu receptor.

Uma imensa variedade de componentes de hardware, software e rede compõe um ambiente de TI, entre eles computadores, roteadores, servidores, aplicativos, sistemas e tecnologias móveis. Essa estrutura de TI pode ser física, estar em determinado local, na nuvem, ou integrada entre as duas opções. Esses ativos podem incluir as pessoas que fazem parte do ambiente de TI, da utilização dos sistemas de gerenciamento, dos administradores de rede, dos usuários de BIs, CRM, ERP etc.

Quem monitora e gerencia o ambiente de TI são os gerentes de TI, que têm como responsabilidades:

- » determinar requisitos de negócios para sistemas de TI;
- » fornecer suporte técnico;
- » gerenciar os sistemas de controle e segurança de rede;
- » gerenciar orçamentos e custos de TI;
- » implementar novos sistemas de software, hardware e dados;
- » monitorar a segurança e conformidade.

Geralmente, quem cuida das estratégias e metas para o negócio das organizações, alinhadas aos departamentos de TI, são os Chief Information Officers (CIOs). Suas responsabilidades fundamentais são “implementar mudanças digitais significativas através da criação de novas ferramentas, soluções e modelos de negócios”. Além disso, assumem um papel de liderança na adoção de novos sistemas para melhorar operações, e tecnologias emergentes, como aprendizado de máquina, análise de chatbots e blockchain. Os CIOs precisaram se tornar mais conhecedor dos negócios para poder usar a tecnologia como aliada na resolução dos problemas e desafios dos negócios.

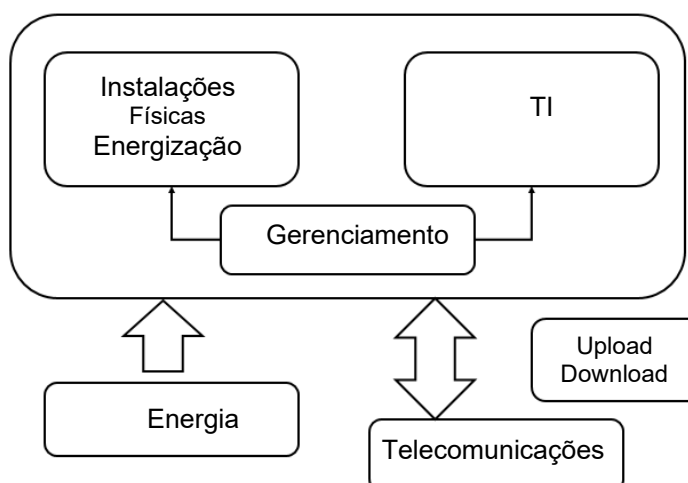
CAPÍTULO 1

Aspectos gerais

Segundo Veras (2015, p. 74), um datacenter é um conjunto integrado de componentes de alta tecnologia que permite fornecer serviços de infraestrutura de TI de valor agregado, processamento e armazenamento de dados, em larga escala, para qualquer tipo de organização.

Como sabemos, há dois tipos de redes utilizadas: a rede pública e a rede privada. Em muitos países, como nos Estados Unidos, por exemplo, grandes corporações utilizam data center internamente. Portanto, é possível existir uma nuvem local. Outras optam por levarem sua estrutura de TI para nuvem pública. Segundo Veras (2015, p. 74), os datacenters podem ser divididos em dois grandes grupos: os eDC, que são os datacenters empresariais; e os iDC, que são os Internet Data Centers, responsáveis pelos serviços de nuvens para terceiros. A estrutura do data center público ou privado será composta por instalações, gerenciamento e TI:

Figura 2. Data center.



Fonte: Veras (2015, p. 75).

As empresas sempre se deparam com a necessidade da demanda de recurso, seja ela uma empresa jovem, que teve seu crescimento exponencialmente, ou uma empresa madura, que se estruturou com eficiência em seu negócio principal e nem tanto na capacidade tecnológica. Em algum momento, todas terão de decidir sobre a reestruturação do parque de TI.

Seja local ou em nuvem, segundo Veras (2015, p. 75), essa mudança envolve a migração de todos os ativos de TI, pessoas e os aplicativos que recebem suporte; portanto, requer um planejamento bem minucioso e abrangente, pois, esse processo é crítico para que o negócio continue a funcionar sem interrupções.

Entretanto, a nova geração de datacenters está surgindo para ser mais flexível para possibilitar a adequação para cada cliente, cada recurso. Seja qual for a necessidade, haverá um data center para ela. Segundo Veras (2015, p. 76), a meta central de um projeto de data center é obter resiliência, que em TI significa atender à demanda de negócios de maneira efetiva, reduzir o custo total de propriedade (TCO) e tornar o negócio mais flexível. Os critérios para isso são:

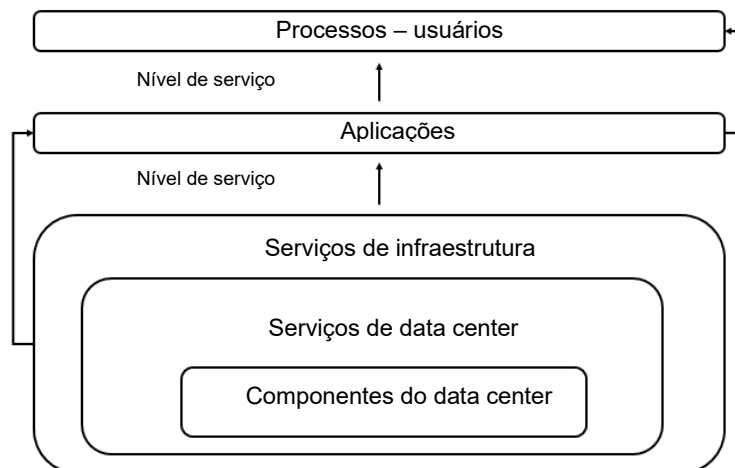
- » desempenho;
- » disponibilidade;
- » escalabilidade;
- » segurança;
- » gerenciabilidade.

Pensar em uma forma de gerar negócios com data center é inevitável. A estrutura que demandou, aliado ao crescimento, fizeram surgir as grandes empresas muito interessadas nesse novo nicho aparente. Foi então que surgiram os novos conceitos de transformar os grandes investimentos nos datacenters em geração de receitas por meio de serviços. Segundo Veras (2015, p. 77), as vantagens são inúmeras, e o custo total de propriedade de uma solução virtualizada é muito fácil de demonstrar devido às diversas opções de ferramentas que surgem. Veras ainda menciona que os principais serviços de TI que podem atingir o nível de serviço são:

- » **Serviços de rede:** são as conexões entre os componentes internos, e deles com o mundo exterior.
- » **Serviços de segurança:** envolvem serviços de firewall, controlando o tráfego e proporcionando monitoramento.

- » **Serviços de processamento:** respondem diretamente pelo desempenho do data center. São os servidores, sistemas operacionais e processadores.
- » **Serviços de armazenamento:** estão relacionados ao armazenamento de dados em unidades de storage.
- » **Serviço de virtualização:** permitem que servidores físicos virtualizem diversos sistemas operacionais.
- » **Serviço de aplicação:** envolve serviço de load-balance, secure socket layer (SSL) offloading e caching.
- » **Serviços de alta disponibilidade (high availability) e disaster recovery:** para obtenção de alta disponibilidade e recuperação de desastres, backup e restore, replicação etc.
- » **Serviço de monitoramento, gerenciamento e automação:** envolve todo o monitoramento de hardware e software, incluindo o NOC (Networking Operation Center).

Figura 3. Serviço de Infraestrutura de TI e Data center.



Fonte: Veras (2015, p. 78).

A Norma TIA 942 define os requisitos para a criação de uma infraestrutura de data center e de telecomunicação e as boas práticas para datacenters de pequenas empresas até datacenters empresariais e de internet. Essa norma é composta de várias outras normas:

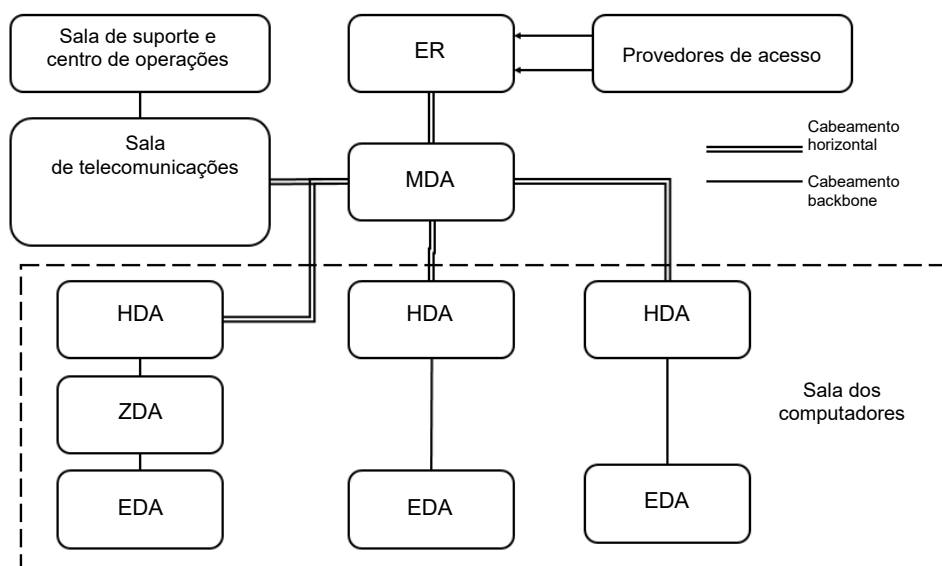
- » **ASHRAE:** classifica padrões de refrigeração.

- » TIA/EIA 568: classifica sistema de cabeamento para aplicações genéricas em edifícios comerciais.
- » TIA/EIA 569: classifica encaminhamento e espaços.
- » TIA/EIA 606: administração uniforme independente das aplicações.
- » TIA/EIA 607: classifica especificação de cabeamentos em prédios comerciais.

Segundo a TIA 942, os principais componentes da arquitetura do data center são:

- » Entrance Room (ER): sala que abriga as interconexões de cabeamento entre as operadoras de telecomunicações e o data center.
- » Main Distribution Area (MDA): conexão central do data center e da distribuição dos cabeamentos, incluindo backbone e roteadores.
- » Horizontal Distribution Area (HDA): inclui os equipamentos horizontais e equipamentos intermediários.
- » Zone Distribution Area (ZDA): interconexão opcional de cabeamento horizontal; fica entre o HDA e EDA.
- » Equipment Distribution Area (EDA): área de servidores, storage, unidades de fita, equipamentos de rede, racks e gabinetes.

Figura 4. Topologia básica de um data center – TIA 942.



Fonte: Veras (2015, p. 81).

CAPÍTULO 2

Eficiência energética

Segundo Veras (2015, p. 100), a eficiência do Data Center, há bem pouco tempo, era medida unicamente em termos de indicadores vinculados à disponibilidade e ao desempenho. Com os aspectos ambientais sendo cada vez mais considerados, o aumento dos custos de energia e a alimentação no fornecimento de energia por parte de alguns provedores, é natural que os gerentes de infraestrutura de TI repensem a estratégia para o Data Center e considerem o aspecto do verde nas diversas escolhas que precisam fazer, incluindo equipamentos e a própria operação.

O Uptime Institute (2005) levantou que os custos de energia representam hoje até 44% do TCO de um Data Center. A eficiência do Data Center, construído há uma década, em geral, não passa de 40%, ou seja, de 100% de energia injetada, só 40% alimenta a carga em TI, os outros 60% são consumidos antes de chegar à carga de TI. Esses Data Centers foram projetados para máxima funcionalidade e desempenho e, na época, não existia tanta preocupação com o consumo de energia. Atualmente, alguns fatores guiam os projetos dos novos Data Centers:

- » aumento na demanda computacional;
- » aumento na densidade dos equipamentos de TI;
- » aumento dos custos de energia;
- » falta de disponibilidade de energia para atender a essas novas demandas.

Por esses fatores, a eficiência energética passou a ser uma das principais preocupações no desenvolvimento de projetos em Data Centers.

Segundo Veras (2015), com o aumento dos custos de energia provocado por servidores cada vez mais densos, é natural que a responsabilidade pelos gastos de energia do data center vá para dentro do departamento de TI. O autor ainda sugere que existem cinco fatores que contribuem para a ineficiência energética:

- » ineficiência dos equipamentos de energia elétrica;
- » ineficiência dos equipamentos de refrigeração;
- » consumo de energia pela iluminação;

- » superdimensionamento dos sistemas de energia e resfriamento;
- » ineficiência devido a configuração.

Equação energética

Calcular o dimensionamento das necessidades de energia de um data center requer compreensão sobre a quantidade de energia para alimentar a carga de TI. Segundo Veras (2015, p. 102), estimar a energia adequada é o aspecto crucial do projeto. Se estimada para baixo, pode conduzir a custos iniciais para baixo; ao contrário, elevaria os custos.

Veras (2015, p. 103) cita duas abordagens para tentar aumentar a eficiência do data center, que podem ser usadas conjuntamente:

- » **Planejar a capacidade instalada:** trata do dimensionamento correto da infraestrutura para a carga de TI esperada.
- » **Utilizar a virtualização:** permite o processamento em um único servidor de carga de trabalho múltipla, aliado na redução de calor, considerando que um servidor a plena carga ou a 15% consome a mesma energia.

Segundo a APC, citada por Veras (2015, p. 104), para vencer os desafios impostos pela virtualização ao projeto de data center, deve-se adotar as seguintes medidas:

- » para carga de altas densidade, utilizar resfriamento baseado em colunas;
- » para sistema de energia e refrigeração superdimensionados, utilizar equipamentos escaláveis;
- » utilizar ferramentas adequadas para saber se a capacidade atende à demanda no nível da fila, rack e servidor.

Em 2007, a Green Grid Association criou duas métricas para verificar a eficiência energética do data center. A Green Grid Association trabalha para melhorar a eficiência de recursos da tecnologia da informação (TI) e datacenters em todo o mundo: PUE e DCiE.

PUE – Power Usage Effectiveness

As métricas de escalabilidade e análise estatística da PUE fornecem métricas de escalabilidade associadas a alterações de cargas, normalmente vistas no data

center. A escalabilidade do PUE é usada para mostrar quão bem a energia total de um data center afeta o consumo de alterações nas cargas de equipamentos de TI.

DCiE – Data center Efficiency

É o inverso do PUE: é a energia dos equipamentos de TI dividida pelo total de energia da instalação. Desde a introdução dessas duas métricas pela The Green Grid, a PUE superou o DCiE na indústria adoção. No futuro, o The Green Grid usará o PUE e o verá como a métrica preferida pelo setor para medir a eficiência energética da infraestrutura nos datacenters.

Das duas métricas, a PUE é a mais utilizada, e sua fórmula é:

$$PUE = \frac{\text{Energia total da instalação}}{\text{Energia dos equipamentos de TI}}$$

A energia total da instalação é a energia em medidores diretamente nos datacenters, e a energia dos equipamentos de TI é o consumo de todos os equipamentos que envolvem o funcionamento do TI. Quanto mais minucioso for para encontrar essa variável, mais exato ficará o resultado. Segundo Veras (2015, p. 106), a medição de energia total pode ser feita logo depois do switch de transferência. Já a APC, citada por Veras (2015, p. 107), sugere que a medição da energia da carga de TI seja feita logo depois do No-break, mas considera perdas se houver as Power Distribution Unit (PDU) entre o No-break e a carga de TI.

Estudo de caso

O Google fez um trabalho para reduzir o uso de energia e, ao mesmo tempo, atender ao crescimento explosivo da Internet. Os detalhes podem ser acessados pelo link: <https://www.google.com/about/datacenters/efficiency/how/>.

Eles justificaram que a maioria dos datacenters usa quase tanta energia não computacional ou “indireta” (como refrigeração e conversão de energia) quanto usa para alimentar seus servidores. No Google, foi reduzida essa sobrecarga para apenas 11%. Dessa forma, a maior parte da energia usada alimenta as máquinas que atendem diretamente a pesquisas e produtos do Google. Foram tomadas medidas detalhadas para avançar continuamente para fazer mais com menos, atendendo a mais usuários e gastando menos energia, e o cálculo adotado foi o PUE.

Os cálculos incluem o desempenho de toda a frota de datacenters em todo o mundo, não apenas das melhores e mais novas instalações – o desempenho foi medido continuamente ao longo do ano, e não apenas nas estações mais frias. Além disso, foram incluídas todas as fontes de custos indiretos na métrica de eficiência.

Segundo o Google, é possível relatar números muito mais baixos se for adotada a interpretação mais vaga dos padrões de medição PUE da Green Grid. De fato, o melhor site pode apresentar uma PUE menor que 1,06 se usar uma interpretação comumente usada no setor. No entanto, estão adotando um padrão mais alto porque acreditam que é melhor medir e otimizar tudo em todos os sites, não apenas em parte dele. Portanto, foi reportada uma PUE abrangente de doze meses (TTM) de 1,11 em todos os nossos datacenters de grande escala (quando atingem operações estáveis), em todas as estações, incluindo todas as fontes de custos indiretos. Foram usados vários medidores de energia on-line nos datacenters para medir o consumo de energia ao longo do tempo. Foi rastreada toda a energia usada por toda a infraestrutura de refrigeração e equipamentos de TI em medidores separados, fornecendo cálculos de PUE muito precisos, inclusive todos os nossos elementos que consomem energia foram incluídos na PUE, usando dezenas ou mesmo centenas de medidores de energia nas instalações.

Desempenho de PUE do Google Data Center

O PUE, em toda a frota, caiu significativamente desde que começaram a reportar os números em 2008. O PUE médio ponderado em energia TTM para todos os datacenters do Google é de 1,11, tornando-os os mais eficientes do mundo, e o PUE da média global de datacenters, segundo pesquisa do Uptime Institute, é de 1,67.

CAPÍTULO 3

Arquitetura e infraestrutura

Segundo Veras (2015, pp. 116 e 117), existem dois modelos que as empresas mais utilizam para construir um data center: o padrão TIA-942 e o padrão sugerido pela Cisco. Ambos são modelos hierárquicos, e o da Cisco foi uma alteração feita no modelo da TIA-942, na qual foi alterada a camada de distribuição, pela camada de agregação. Esse modelo, segundo a empresa, está sendo testado por vários anos em grandes datacenters.

As camadas da norma TIA-942 são formadas com a seguinte estrutura:

- » MDA: Main Distribution Area.
- » HDA: Horizontal Distribution Area.
- » EDA: Equipment Distribution Area.

Já as camadas do modelo da Cisco são estruturadas da seguinte maneira:

- » Core: formada por switches de alta velocidade para fluxos que entram e saem do data center. Ela fornece conectividade a vários módulos de agregação e uma malha roteada, sem um único ponto de falha. Também executa um protocolo de roteamento interno, como OSPF ou EIGRP, e equilibra a carga de tráfego entre o núcleo e as camadas de agregação usando algoritmos de hash baseados no Cisco Express Forwarding.
- » Aggregation: formada por switches inteligentes que integram os principais serviços, como balanceamento de carga, detecção de intrusão, firewalls, SSL etc.
- » Access: formada por switches conectados nos servidores à rede em que estão configuradas as políticas de rede (ACLs, QoS, VLANs), servidores blade com comutadores integrais, servidores blade com cabeamento de passagem, servidores em cluster e mainframes com adaptadores OSA. A infraestrutura de rede da camada de acesso consiste em comutadores modulares, comutadores de configuração fixa e comutadores integrais de servidor blade.

A estrutura de rede, segundo Veras (2015, p. 117), é formada por grandes switches modulares, que podem ficar localizados nas colunas dos datacenters (End-of-Row – EoR) ou fixo no topo do rack (Top of Rack – ToR). Essa estrutura poderá utilizar

cabeamento de cobre, categoria 6 e 6a para redes de 10Gbps. Para redes com velocidade maior, deve-se utilizar fibra ótica.

Os principais recursos utilizados em datacenters, como recurso de processamento e memória, servidores, clusters, armazenamento e conectividade, estão sendo alterados conforme são disponibilizados. O armazenamento, a refrigeração e o espaço físico estão recebendo forte influência de arquiteturas baseadas em *blades*. Outra influência muito grande, que chega a inviabilizar as arquiteturas antigas, é a virtualização e a clusterização. A virtualização foi mencionada no capítulo 3 da Unidade I. Agora, falaremos sobre a clusterização e a virtualização.

Segundo Veras (2015, p. 122), a virtualização resolve problemas de sobra de recursos em servidores físicos, e a clusterização resolve o problema da falta de recursos em servidores físicos. Ele deve permitir que uma aplicação rode em mais de uma máquina física, incrementando a performance ao mesmo tempo que aumenta a disponibilidade.

O clustering, alternativamente, é uma estratégia para computação de alto desempenho e balanceamento de carga. O armazenamento em cluster permite agregar recursos, oferecendo desempenho virtual de grande importância. Todos os clusters têm o objetivo comum de combinar várias CPUs para aparecer como um sistema unificado de alto desempenho usando software especial e interconexões de rede de alta velocidade.

Os clusters podem ser classificados em:

- » **Cluster de alta disponibilidade (high availability – HA):** endereça redundância com capacidade de recuperação de falhas automática.
- » **Cluster de balanceamento de carga (Load Balancing – LB):** melhora da capacidade para a execução da carga de trabalho.
- » **Cluster de alta performance (HPC e HTC):** endereça aumento da performance da aplicação.
- » **Cluster em grid:** engloba alta disponibilidade e alto desempenho.

Cluster de Alta performance

Com base na documentação da Cisco, podemos fazer uma análise sobre clusters de alta performance.

Embora os clusters de alto desempenho (HPCs) tenham suas configurações de arquiteturas variadas, existem três classificações:

HPC tipo 1 – Passagem paralela de mensagens (acoplamento rígido):

- » Os aplicativos são executados em todos os nós de computação simultaneamente em paralelo.
- » Um nó mestre determina o processamento de entrada para cada nó de computação.
- » Pode ser um cluster grande ou pequeno, dividido em seções (por exemplo, 1.000 servidores com mais de 20 seções) com comunicação IPC entre nós de computação/seções.

HPC tipo 2 – Processamento de E/S distribuído:

- » A solicitação do cliente é balanceada entre nós principais e, em seguida, pulverizada para calcular nós para processamento paralelo.
- » Esse tipo obtém a resposta mais rápida, aplica inserção de conteúdo (publicidade) e envia ao cliente.

HPC Tipo 3 – Processamento de arquivo paralelo (acoplamento fraco):

- » O arquivo de dados de origem é dividido e distribuído pelo pool de computação para manipulação em paralelo. Os componentes processados são reunidos novamente após a conclusão e gravados no armazenamento.
- » O middleware controla o processo de gerenciamento de tarefas (por exemplo, sistema de arquivos linear da plataforma [LFS]).

Cluster de Balanceamento de carga e Alta Disponibilidade

Segundo Veras (2015, p. 123), um cluster de alta disponibilidade (HA) destina-se a aplicativos que se mantêm muito tempo em memória, denominados aplicativos de monitoramento de estado, como banco de dados. Os clusters de balanceamento de carga destinam-se para aplicativos que ficam por menos tempo em memória, que normalmente são de somente leitura ou dados frequentemente alterados, como servidor web, VPNs, servidores FTP, firewalls etc.

Top 500

A top500.com é uma lista da classificação dos 500 supercomputadores. Ela é atualizada semestralmente e possui vários tópicos de classificação, como top Green, voltada para a eficiência energética, HPC voltada para o alto desempenho.

Por exemplo, no dia 15 de outubro de 2019, a IBM possui dois computadores na top 2, Summit e Sierra, instalados no Laboratório Nacional Oak Ridge (ORNL) do Departamento de Energia no Tennessee e no Laboratório Nacional Lawrence Livermore na Califórnia, respectivamente. Ambos derivam seu poder computacional das CPUs Power 9 e das GPUs NVIDIA V100. O sistema Summit possui um recorde de 148,6 petaflops, enquanto o sistema Sierra 94,6 petaflops.

Flops, acrônimo de floating point operations, é uma unidade usada para medir a capacidade de processamento ou o cálculo de pontos flutuantes que um computador pode realizar por segundo.

CAPÍTULO 4

Computação em nuvem como negócio

Vimos que, em relação à computação em nuvem, as empresas poderão conseguir muitas oportunidades e bons negócios, e conhecemos os tipos de serviços que grandes empresas, como Amazon, Microsoft, Google e IBM, oferecem para que as empresas menores possam migrar seu parque de TI para a nuvem.

Como podemos conseguir oportunidades tendo essas empresas como concorrentes? Mesmo que tenhamos um capital muito alto para investir e concorrer com os grandes, o tempo de maturidade conta muito quando estamos iniciando um negócio novo. Porém podemos ver oportunidades e ter a nuvem como um negócio, aliando-se às grandes e prestando serviços de TI, por meio dos três modelos que existem: IaaS, SaaS e PaaS. Os profissionais com esse perfil poderão ser um arquiteto de serviços, especialista em migração, especialista em segurança etc.

Primeiramente poderíamos aproveitar as eventuais lacunas que estão entre as empresas que ainda não migraram sua infraestrutura de TI para a nuvem. Vamos criar um cenário hipotético para estudo, porém existem muitas empresas nessa situação, em que precisam migrar sua infraestrutura de 80 estações de trabalho e utilizar os produtos da Microsoft, como Windows 10 e Pacote Office, em uma estrutura de rede única, sem nada instalado ou armazenado localmente.

Podemos disponibilizar esses produtos por meio de virtualização de servidores, instalando sistemas operacionais e aplicativos. Nesse caso, poderíamos utilizar qualquer uma das grandes empresas que oferecessem IaaS e SaaS. Entretanto, por se tratar de produtos Microsoft, é mais vantajoso, de fácil implementação, atualização e monitoramento se contratássemos um SaaS diretamente no Azure, pois todos os sub-serviços ficariam sob responsabilidade do provedor desses serviços.

Em uma rápida pesquisa no serviço de “Área de Trabalho Virtual do Windows” (<https://azure.microsoft.com/pt-br/services/virtual-desktop/>). Acesso em: 1º out. 2019), podemos fazer uma lista de opções interessantes para se oferecer às empresas. Lembre-se de que você é o prestador de serviços de implantação, configuração e instalação. Seu trabalho será deixar tudo pronto para o cliente utilizar, treinar se for preciso e/ou visitá-lo periodicamente. Portanto, para ele caberá dois Contratos de Serviços: um com a Microsoft, relacionado ao SaaS, e outro com você, referente aos serviços de implantação.

O serviço de Área de Trabalho Virtual do Windows disponibiliza as seguintes opções:

- » implantação e escala das áreas de trabalho e dos aplicativos virtualizados do Windows no Azure;
- » entrega de gerenciamento simplificado, Windows 10 multissessão, otimizações para o Office 365 ProPlus, é compatível com ambientes de RDS (Serviços de Área de Trabalho Remota), obtendo a segurança interna e funcionalidades de conformidade;
- » acesso a áreas de trabalho e aplicativos do Windows 10 Enterprise e do Windows 7 Enterprise sem custos adicionais se tiver uma licença elegível do Windows ou do Microsoft 365.
- » obtenção gratuita das atualizações de segurança estendidas até janeiro de 2023 para a área de trabalho virtual do Windows 7, com mais opções de suporte para aplicativos herdados ao fazer a transição para o Windows 10.

No link <https://azure.microsoft.com/pt-br/pricing/calculator/?service=virtual-deskto>, é possível fazer o cálculo estimado de quanto o cliente irá dimensionar e efetivamente pagar. No cenário proposto acima, foram simulados os seguintes recursos:

- » Região: leste dos EUA.
- » 100 usuários.
- » 220 horas de uso por mês (estimadas).
- » Máquina virtual que será disponibilizada, porém não será necessário interferência: 4 virtual CPU, 16 GB de RAM, 32 de armazenamento temporário.
- » 6 instâncias.
- » 6 discos de 128 GiB.

Pelo simulador, o valor foi de R\$ 1.519,20 por mês, incluindo as licenças dos softwares. Além disso, é necessário calcular o valor de implantação. Normalmente, esse valor é dado pela quantidade de horas que foram e serão disponibilizadas. Depois de funcionar, o cliente terá toda sua estrutura de TI em nuvem.

Esse foi um exemplo de um cenário, porém o site <https://www.profitableventure.com/cloud-computing-based-business-ideas/> oferece as melhores ideias de negócios baseados em computação em nuvem. Veja alguns:

- » **Crie software de apresentação baseado em nuvem:** o software de apresentação online é uma plataforma baseada na nuvem que, ao contrário do PowerPoint e de outros aplicativos de desktop, não há necessidade de download e você pode trabalhar nas suas apresentações on-line.
- » **Call center hospedado na nuvem:** um call center ou cloud contact center hospedado na nuvem é um ponto central de uma empresa, hospedada em um servidor da Internet, a partir do qual todas as comunicações de entrada e saída do cliente são tratadas.
- » **Serviço de armazenamento em nuvem:** o uso de um servidor em nuvem permite que o usuário baixe arquivos para outros dispositivos quando necessário. Você pode iniciar um serviço de armazenamento em nuvem e fornecer seus serviços a empresas que lidam com grandes dados.
- » **Ferramentas de nuvem de gerenciamento de log:** um log é a documentação produzida automaticamente e com registro de data e hora dos eventos relevantes para um sistema específico. Você pode começar a produzir e vender suas ferramentas de gerenciamento de logs baseadas na nuvem para empresas que precisam delas.
- » **Hospedagem de e-mail baseado em nuvem:** a nuvem é a mais recente técnica de hospedagem e é usada para vários serviços comerciais e pessoais, entre os quais a hospedagem de e-mail.
- » **Servidores em nuvem privados:** os servidores em nuvem privada são criados para oferecer alto desempenho, acesso e upload de dados mais rápidos.
- » **Hospedagem de servidores dedicada:** a hospedagem na Web é um serviço que permite que organizações e indivíduos publiquem um site ou página da Web na Internet.

Criptomoedas

Agora que o Bitcoin atingiu o auge e se tornou um fenômeno mundial, mais pessoas estão tentando entrar no ramo de criptomoedas.

Outra oportunidade está na Legislação. A maioria das empresas que decidem contratar um data center fora do país não está familiarizada com a Tributação dos Serviços.

A EVEO Enterprise Cloud é uma empresa especializada em oferecer soluções de infraestrutura para data centers, principalmente os hospedados na nuvem. Ela criou um guia que demonstra a complexidade sobre esse assunto, disponível em: <https://www.eveo.com.br/ebooks/Guia-Tributacao-Data-Center-Exterior.pdf>. Segundo o guia, a Receita Federal deixou bem claro que, ao contratar um provedor de data center ou cloud no exterior, a pessoa física ou empresa sediada no Brasil deverá recolher todos os impostos sobre importação de serviço. Além disso, a instrução normativa deixa claro quais são esses impostos (somente os federais):

Sobre os valores de que trata o caput devem incidir o Imposto sobre a Renda Retido na Fonte (IRRF), a Contribuição de Intervenção no Domínio Econômico destinada a financiar o Programa de Estímulo à Interação Universidade-Empresa para o Apoio à Inovação (Cide-Royalties), a Contribuição para o PIS/Pasep-Importação e a Cofins-Importação

Como podemos observar, existem inúmeras oportunidades de criarmos um negócio baseado em computação em nuvem.

CAPÍTULO 1

Web API

API significa Interface de Programação de Aplicações (Application Programming Interface). Por meio dela, desenvolvedores constroem aplicações que interagem entre si sem precisar ter a mesma compatibilidade de linguagem de programação, sistema operacional ou equipamentos. Existem APIs para todos os tipos de sistemas, incluindo sistemas operacionais, bibliotecas e Web.

Uma API é composta por vários elementos, como métodos, protocolos e ferramentas que permitem aos desenvolvedores criar aplicativos. Ela também está presente em vários níveis do sistema operacional. Por exemplo, um sistema operacional é composto por centenas de software, executando e realizando diversos tipos de operações. Muitos deles foram feitos por equipes ou empresas diferentes, porém foram programados para se comunicarem e ser compatíveis entre si.

Hoje, com a massiva popularização da Internet, a necessidade de comunicado entre dispositivos está tomando proporções gigantescas. Antigamente, quando falávamos em dispositivos, a referência era a comunicação entre computadores. Nos dias atuais, a palavra dispositivo abre espaço para smartphones, impressoras, carros, televisores, fechaduras eletrônicas, câmeras de vigilâncias etc.

Por isso, antes de estudarmos uma Web API, vamos entender cada elemento necessário para seu funcionamento.

Protocolos de rede

Protocolos de rede são um conjunto de normas para garantir uma padronização de como as máquinas podem se conectar e se comunicar. Eles estabelecem todos os processos, requisitos e restrições para iniciar e realizar a comunicação

entre todos os dispositivos que desejam conectar em uma rede e que possa ser instalado ou implantado, como computadores, servidores, roteadores e outros dispositivos. Existem vários protocolos, para vários determinados fins. Por exemplo:

- » **TCP** (Transmission Control Protocol): é um protocolo usado para comunicação em rede. Sua função é controlar os pacotes que forem transmitidos e recebidos. Cada pacote transmitido chama-se datagrama e é composto pelos endereços de origem e destino, uma marcação para saber quanto ainda falta do arquivo a ser transmitidos, entre outros. É um protocolo que vai garantir a entrega dos arquivos de comunicação.
- » **IP** (Internet Protocol): é um protocolo de endereçamento que orienta o endereçamento ao protocolo TCP. Por isso, esse conjunto, para ser completo, é chamado de TPC/IP. Os endereços IP nos pacotes ajudam no roteamento por meio de diferentes nós em uma rede até atingir o sistema de destino.
- » **UDP** (User Datagram Protocol): é um protocolo de comunicação usado principalmente para estabelecer conexões de baixa latência e tolerância a perdas entre aplicativos na Internet. Foi criado como alternativa para o TCP. Enquanto o TCP garante as entregas dos pacotes, com o UDP os pacotes podem seguir caminhos diferentes entre remetente e destinatário e, como resultado, alguns pacotes podem ser perdidos ou recebidos fora de ordem, porém é mais rápido para certas implementações de redes e transmissões, com largura de banda mais baixa, por exemplo.
- » **POP e SMTP** (Post Office Protocol) é um protocolo criado para orientar o recebimento de e-mails e o SMTP (Simple Mail Transfer Protocol), para enviar e distribuir e-mails.
- » **FTP**: é um protocolo projetado para gerenciar a transmissão de arquivos, ou seja, é por meio dele que os usuários enviam arquivos aos servidores. Os arquivos podem ser de vários tipos, como arquivo de texto, arquivo de planilhas eletrônicas, músicas, vídeos etc.

Por fim, falaremos sobre o HTTP, que é o protocolo utilizado em Web API.

Protocolo HTTP

O HTTP (Hyper Text Transfer Protocol) é um protocolo utilizado para envio e recebimento de documentos pela web. Para que essa transmissão seja possível, é necessário um cliente fazendo uma requisição a um servidor, que executa alguma operação e retorna o resultado para o cliente. Essa comunicação é feita por meio de mensagens de textos ou bits de textos, pois nelas podem conter mídias de áudio, vídeo etc.

Para que o cliente encontre o destino que ele deverá fazer a requisição, é necessário existir uma URL (Uniform Resource Locator). As URLs são o destino final de uma requisição HTTP; são chamadas de end point ou endereço final.

Uma URL completa é composta do endereço do destinatário e de segmentos que poderão ter ou não parâmetros de consulta. Por exemplo, vamos imaginar que precisamos acessar um serviço do IBGE que irá nos retornar os dados dos municípios do Brasil. O endereço do serviço que o IBGE disponibiliza é:

» <https://servicodados.ibge.gov.br/api/v1/localidades/municipios>.

Decompondo essa URL, conforme a explicação, temos:

https://servicodados.ibge.gov.br	/	api/v1/localidades/municipios
Endereço do destinatário		Segmentos ou parâmetros

Os segmentos poderão ser parâmetros ou não e dependerá do tipo de programação que foi feita no servidor. Como exemplo, a URL a seguir, especificamente na parte do segmento:

» <https://servicodados.ibge.gov.br/api/v1/localidades/estados/{UF}/municipios>.

O site do IBGE dispõe da seguinte documentação para o parâmetro UF que está no segmento da URL:

Quadro 3. Documentação da URL de serviços do IBGE.

Municípios por UF	
Obtém o conjunto de municípios do Brasil a partir dos identificadores das Unidades da Federação	
UF	<p><i>string</i></p> <p><i>Um ou mais identificadores de Unidades da Federação delimitados pelo caracter (pipe)</i></p> <p><i>Exemplos:</i></p> <p><i>http://servicodados.ibge.gov.br/api/v1/localidades/estados/33/municipios</i></p> <p><i>Obtém os dados referentes aos municípios do Rio de Janeiro (33)</i></p> <p><i>http://servicodados.ibge.gov.br/api/v1/localidades/estados/33 35/municipios</i></p> <p><i>Obtém os dados referentes aos municípios do Rio de Janeiro (33) e São Paulo (35)</i></p> <p><i>Required</i></p>

Fonte: IBGE <https://servicodados.ibge.gov.br/api/docs/localidades?versao=1#api-Municipios-estadosUFMunicipiosGet>. Acesso em: 27/2/2020.

As requisições mais comuns são as requisições GET e POST, entretanto, com o surgimento de novos conceitos, como o REST, novos verbos foram adicionados no HTTP para que ficasse mais seguro e flexível.

GET

É o verbo HTTP mais simples e mais utilizado. Ele é um verbo somente para recuperar informações, e não para alterá-las, ou seja, ele não modifica nenhum dado no sistema. Um exemplo foi a consulta do IBGE que fizemos anteriormente.

POST

Esse verbo é utilizado quando precisamos processar algo no servidor, como adicionar informações. Podemos utilizar o método POST toda vez que enviamos um formulário ao servidor, pois seu corpo carrega todas as informações de cadastro.

Exemplo: cadastro de usuários, produtos, clientes etc.

PUT

Esse verbo é utilizado quando queremos criar ou atualizar algo novo no servidor. Em princípio, pode parecer com o POST, entretanto a grande diferença está nas operações

sequenciais. Se fizer atualização de recursos usando o PUT e, em seguida, fizer a mesma requisição, o recurso ainda estará no mesmo estado.

DELETE

Será utilizado toda vez que quisermos excluir algum recurso passado na URL.

Exemplo: deletar um usuário.

Quadro 4. Exemplo de endpoints, seus verbos REST e suas ações.

Endpoint	Método	Ação
/users	GET	Retorna lista de usuários
/users	POST	Insere um novo usuário
/users/{id}	GET	Retorna os dados do usuário id = {id}
/users/{id}	PUT	Substitui os dados do usuário id = {id}
/users/{id}	PATCH	Altera itens dos dados do usuário com id = {id}
/users/{id}	DELETE	Remove o usuário com id = {id}

Fonte: Elaborada pelo autor.

Sempre que efetuamos uma operação com os verbos, o protocolo HTTP nos responde com um status code ou código de status, contido no header das requisições, que também contém encode, content type etc.

Os principais status code são:

- » **200 OK** – Solicitação do cliente foi recebida, entendida e aceita com êxito pelo pela API.
- » **201 Created** – Solicitação recebida, entendida, aceita com êxito e resultou na criação de um novo recurso.
- » **202 Accepted** – Solicitação recebida, entendida, aceita com êxito para ser processada, porém o processamento não foi concluído.
- » **203 No Content** – Solicitação recebida, entendida, aceita com êxito, porém a API não precisa retornar um corpo de entidade e pode querer retornar informações atualizadas.
- » **301 Moved Permanently** – O recurso foi atribuído a uma nova URL e deve ser redirecionado para tal.
- » **302 Found** – O recurso foi encontrado e será efetuado um redirecionamento.

- » **401 Unauthorized** – Significa que o usuário precisa de uma autorização, que, no caso, foi negada.
- » **403 Forbidden** – Indica que a solicitação do cliente foi formada corretamente, mas a API se recusa a cumpri-la, ou seja, o usuário não tem as permissões necessárias para o recurso.
- » **404 NotFound** – Não foi encontrada a URL ou o recurso no servidor.

Vamos agora praticar um pouco. Vale lembrar que toda simulação será feita localmente, por dois motivos: pelo fato de ser local, e não em uma nuvem, não iremos perder nenhum tipo de aprendizado, pois o conceito será o mesmo; para fazermos a simulação em uma nuvem comercial, teremos que nos cadastrar, e o processo é mais demorado, então teríamos que demonstrar na maioria delas. Portanto, localmente será nossa melhor opção para estudo de WebApi.

Primeiramente baixe e instale as ferramentas a seguir:

- » **NodeJS** – <https://nodejs.org/en/>.
- » NodeJS é uma plataforma que oferece microserviços, porém teremos um capítulo só para esse assunto. O que precisamos da instalação do NodeJS, nesse momento, é uma ferramenta chamada NPM.
- » **Postman** – <https://www.getpostman.com/>.
- » Postman é uma plataforma de colaboração para o desenvolvimento de API. Os recursos do Postman simplificam cada etapa da criação de uma API e agilizam a colaboração para que você possa criar APIs melhores, entretanto utilizaremos somente o cliente de API do Postman.
- » **JSON Server** – <https://www.npmjs.com/package/json-server>.

JSON Server é um servidor de API, de fácil implementação, que roda localmente. Ideal para confeccionarmos clientes de APIs para serem publicadas, posteriormente, em produção.

Depois do NodeJS instalado, abra um terminal e digite:

- » `npm install -g json-server`.

Se a instalação foi bem-sucedida, aparecerá a seguinte mensagem, podendo variar as versões:

- » + json-server@0.15.1.
- » added 237 packages from 128 contributors in 60.528s.

Em seguida, abra um editor de texto puro, como o bloco de notas, e um arquivo chamado db.json:

```
{
  "postagens": [
    { "id": 1, "title": "Primeira Postagem", "autor": "aluno" },
  ],
  "comentarios": [
    { "id": 1, "body": "Meu primeiro comentario", "postId": 1 },
  ],
  "profile": { "name": "alunos" }
}
```

Em seguida, inicie o servidor:

- » json-server --watchdb.json

Se tudo funcionou bem, a seguinte mensagem será exibida:

```
\{^_^\}/hi!
```

```
Loadingdb.json
```

```
Done
```

```
Resources
```

```
http://localhost:3000/postagens
```

```
http://localhost:3000/comentarios
```

`http://localhost:3000/perfil`

Home

`http://localhost:3000`

Type s + enter at any time to create a snapshot of the database

Watching.

Agora, abra o Postman e, na tela que aparecerá, escolha o método GET, digite o endereço e clique em “Send”. O resultado deverá ser conforme a figura:

Figura 5. Testando a web api pelo método GET.



Fonte: Elaborada pelo autor.

Para testar o método POST, altere a opção no Postman:

- » Método de envio: POST.
- » Body: opção x-www-form-urlencoded.
- » Campos (key):
 - › Key= body, value= Meu segundo comentário.
 - › Key= postId, value= 1.

Veja a mensagem que aparecerá na área de visualização:

```
{  
  "body": "Meu segundo comentário",  
  "postId": "1",  
  "id": 2  
}
```

Para testar o método DELETE, altere a opção no Postman:

- » Método de envio: DELETE.
- » Body: opção x-www-form-urlencoded.
- » Campos (key):
 - › Key= postId, value= 1.

Executado esse comando, o registro de postId, value= 1, será excluído.

CAPÍTULO 2

Mobile

Existem algumas definições existentes para o termo computação em nuvem móvel. Isso acontece pelo fato de as possibilidades de se trabalhar com a computação móvel serem muito extensas. De modo geral, o termo computação em nuvem móvel significa executar um aplicativo remotamente, em um servidor bem equipado de recursos de processamento, memória e armazenamento, enquanto o dispositivo móvel age como um cliente conectando no servidor remoto por meio de 3G. Alguns outros exemplos desse tipo são os serviços com reconhecimento de local do Facebook e do Twitter para widgets meteorológicos móveis etc. Neste material, vamos adotar dois conceitos: o desenvolvimento de aplicações web e sua disponibilização em nuvem.

Mobile cloud computing

Segundo Fernando (2012, p. 87), uma abordagem é considerar outros dispositivos móveis também como provedores de recursos da nuvem que compõem um dispositivo móvel rede ponto a ponto, ou seja, disponibilizar os recursos coletivos dos vários dispositivos móveis nas proximidades e outros dispositivos fixos, se acessíveis. Essa abordagem suporta a mobilidade do usuário e reconhece o potencial das nuvens móveis para realizar o sensor coletivo também.

Outra abordagem da computação em nuvem móvel ou MCC (Mobile Cloud Computing), na sua forma mais simples, refere-se a uma infraestrutura em que o armazenamento e o processamento de dados ocorrem fora do dispositivo móvel. Os aplicativos de nuvem móvel movem o poder da computação e o armazenamento de dados dos telefones celulares para a nuvem, levando os aplicativos e o MCC não apenas aos usuários de smartphones, mas a uma gama muito mais ampla de assinantes móveis.

A segurança em relação aos dados é muito confiável, e o backup é feito enviando todos os dados na nuvem. Esse backup pode ser recuperado a qualquer momento de maneira segura. Os dados armazenados com a ajuda do aplicativo móvel para nuvem podem fazer backup facilmente e recuperar quando necessário. A recuperação de desastres na nuvem é um plano que consiste em armazenar e manter cópias de dados em vários locais, mantendo as medidas de segurança em seu pico.

Vantagens da computação em nuvem móvel

Flexibilidade

- » O Mobile Cloud Computing é flexível, pois permite acessar dados de qualquer lugar e a qualquer momento. O cliente exige apenas uma conexão com a Internet e um dispositivo com o qual eles possam acessar dados na nuvem.
- » Disponibilidade de várias plataformas.
- » O aplicativo de computação em nuvem introduz pela empresa o uso em várias plataformas, como Android, IOS e muito mais. A nuvem pode acessar e modificar facilmente, independentemente da plataforma.

Financeiro

O Mobile Cloud Computing elimina o custo do hardware e é um dos métodos mais econômicos de usar e manter. A computação em nuvem móvel tem um custo inicial muito menor, pois o cliente precisa pagar apenas pelo que usou.

Armazenamento

O armazenamento remoto faz parte do Cloud Computing móvel, no qual os dados podem armazenar e recuperar com a ajuda do telefone móvel. O armazenamento de dados na nuvem remotamente garante que as informações desejadas estejam no lugar certo e possam ser recuperadas a qualquer momento, assumindo a disponibilidade de conectividade confiável.

Desenvolvimento de aplicações

As empresas desenvolvedoras de software estão produzindo cada vez mais aplicações para dispositivos móveis. Vimos também a facilidade de criarmos um ambiente de banco de dados e uma plataforma de desenvolvimento, e essa produtividade permite que os desenvolvedores foquem no processo de criação. Com isso, elas estão desenvolvendo aplicativos móveis que ajudam os clientes diariamente, oferecendo atualizações de melhorias e segurança, conseguindo, assim, entregas mais rápidas e confiáveis. Um aplicativo mobile pode ser desenvolvido de várias maneiras, entre elas a forma Nativa e Cross-Plataforma, e todas elas poderão se utilizar de WebApis.

Plataforma cruzada

As arquiteturas de desenvolvimento de aplicativos, de plataforma cruzada, permitem que os desenvolvedores criem aplicativos móveis compatíveis com mais de um sistema operacional – os mais utilizados são o Android e o iOS –, que fornece a capacidade de escrever o código uma vez e executá-lo em qualquer lugar para outras plataformas.

Plataforma nativa

Aplicativos nativos são escritos no código usado preliminarmente para o dispositivo e seu sistema operacional. Os aplicativos nativos funcionam com o sistema operacional do dispositivo de maneira a permitir um desempenho mais rápido e flexível do que os tipos de aplicativos alternativos. Se o aplicativo for comercializado para usuários de vários tipos de dispositivos, os desenvolvedores criarão uma versão do aplicativo separada para cada um.

As comparações entre plataformas podem ser feitas baseando-se em algumas características peculiares de ambas, como, por exemplo, no momento da codificação e no momento da execução da aplicação. No artigo denominado “Deciding between native and cross-platform mobile frontend programming frameworks”, Giovanni Maggini mostra as principais vantagens e desvantagens na execução e na codificação.

Codificação

A codificação para aplicativos nativos é desenvolvida com linguagens de programação e ferramentas oferecidas pela empresa que desenvolve a plataforma e o SO em que são executados. Por exemplo, aplicativos iOS nativos seriam escritos principalmente em Objective-C e Swift, enquanto aplicativos Android nativos seriam escritos em Java ou Kotlin.

Já na plataforma cruzada, o desenvolvimento é feito em linguagens e ferramentas de programação não incluídas nas ferramentas de desenvolvimento oferecidas pela empresa que desenvolve a plataforma e o SO em que são executados. Essa é uma ampla categorização, incluindo pilhas e estruturas como Ionic, Xamarin, React Native, Apache Cordova, Flutter e muito mais.

Execução

Aplicativos nativos são escritos em linguagens de programação nativas ou de plataforma cruzada e são compilados para o código de montagem antecipadamente (AOT) ou just-

in-time (JIT), que ainda aproveitam os widgets da interface do usuário nativos. Por exemplo, os aplicativos escritos em Xamarin, React Native e Flutter pertencem a essa categoria. Esse é um conceito recente, apoiado no fato de que essas estruturas foram lançadas recentemente.

Aplicativos híbridos são escritos em linguagens de programação baseadas na Web de plataforma cruzada (HTML, CSS e JavaScript / ECMAScript ou SCSS e Typescript) e são executados em um shell nativo, geralmente um WebView ou equivalente. Aplicativos escritos em Cordova e Ionic são exemplos dessa categoria.

Vantagens e desvantagens de app nativos

Vantagens

- » Desempenho e funcionamento mais rápidos.
- » Facilita acesso aos dispositivos de hardware do aparelho como câmera, bússola, microfone etc.

Desvantagens

- » Será necessário gerar codificação para várias plataformas e sistemas operacionais.
- » É um aplicativo demorado e difícil de desenvolver.
- » Os aplicativos construídos são de maior alcance e flexíveis. Há uma variedade de abordagens e dispositivos de desenvolvimento compatíveis com a computação em nuvem móvel. Na MCC, o cliente pode selecionar os serviços necessários para seus negócios, o que os torna mais flexíveis.

Plataforma disponíveis

Google Cloud

A Google oferece vários serviços para dispositivos móveis que vão desde o desenvolvimento até o funcionamento e a distribuição.

Serviços de back-end de app para dispositivos móveis

A maioria dos jogos e apps para dispositivos móveis precisa de um serviço de back-end para tarefas que não podem ser realizadas exclusivamente no dispositivo, como compartilhamento e processamento de dados de vários usuários ou armazenamento de arquivos grandes.

Compute Engine e o REST

A execução do back-end móvel no Compute Engine e o uso do REST como protocolo de comunicação entre o app para dispositivos móveis e o servidor de back-end tem várias vantagens:

- » é a forma mais rápida de mover um serviço atual em execução em uma máquina local ou virtual, para o Google Cloud Platform (GCP);
- » oferece controle total sobre a configuração da máquina virtual e do servidor;
- » possibilita o uso de bibliotecas de terceiros;
- » permite configurar um autoescalador para escalonar o número de máquinas virtuais para atender à demanda.

A desvantagem de executar o serviço no Compute Engine é que você fica responsável por manter e atualizar o servidor.

Google Compute Engine e o gRPC

A biblioteca gRPC permite que um aplicativo para dispositivos móveis chame diretamente métodos em um serviço de back-end como se fosse um objeto local. O gRPC pode ser usado para tornar o aplicativo para dispositivos móveis mais eficiente em termos de largura de banda e reduzir a latência entre o aplicativo e o serviço de back-end executado no Google Cloud Platform.

Plataforma para desenvolvimento de jogos

É possível criar ferramentas para analisar eventos de jogos para dispositivos móveis usando dois padrões principais de arquitetura:

- » processamento em tempo real de eventos individuais com um padrão de streaming;
- » processamento em massa de eventos agregados com um padrão de lote;
- » cloud memorystore para Redis como uma tabela de classificação em jogos.

Microserviço

O microserviço é implementado no Cloud Functions para Firebase e usa os seguintes produtos do Cloud AI para traduzir as mensagens:

- » Cloud Speech-to-Text;
- » Cloud Translation;
- » Cloud Text-to-Speech.
- » O microserviço armazena mensagens de áudio traduzidas em um intervalo no Cloud Storage para Firebase.

Aplicativo cliente

O componente cliente é um aplicativo para Android que registra mensagens de áudio e faz o download das mensagens traduzidas do intervalo do Cloud Storage.

CAPÍTULO 3

Microsserviços

Segundo a RedHat, microsserviços são uma abordagem de arquitetura para a criação de aplicações que se diferencia das aplicações de abordagens monolíticas tradicionais. Ela é composta por várias funções, denominadas de serviços, e pode ser criada e implantada de maneira independente. Isso significa que cada serviço individual pode funcionar ou falhar sem comprometer os demais. Uma aplicação monolítica é criada como uma única unidade de execução.

Executáveis

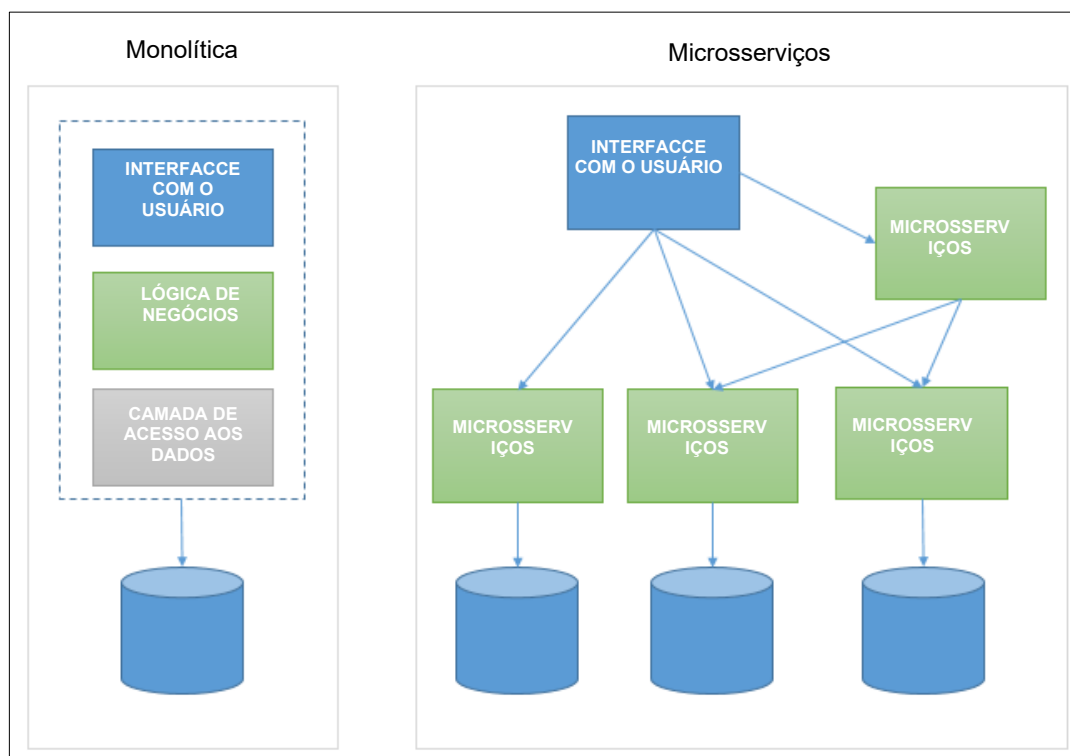
Os aplicativos executáveis passam por vários processos de transformação até serem efetivamente um aplicativo funcional. Existem várias maneiras de se gerar um aplicativo executável, porém, em sua grande maioria, o processo é a compilação. O processo de compilação transforma todas as instruções criadas em uma linguagem de programação em um arquivo que possa ser executado e passa por várias etapas: análises léxicas, sintáticas e semânticas e gerador de código-alvo.

A **Análise Léxica** é a primeira fase e sua tarefa é ler os caracteres de entrada, remover espaços em brancos, correlacionar com mensagens de erros e produzir uma sequência de símbolos básicos que compõem uma linguagem, para a próxima fase.

Na próxima fase, que é a **Análise Sintática**, analisa-se a forma com que os símbolos são combinados e reconhecidos numa sentença. Sua função é verificar se as construções usadas no programa estão gramaticalmente corretas.

A **Análise Semântica** verifica as particularidades de cada linguagem de programação, sob o código enviado da análise sintática. Após a fase da Análise Sintática, o compilador tem como objetivo básico a criação de um módulo executável ou uma linguagem intermediária.

Figura 6. Aplicação monolítica versus microsserviços.



Fonte: <https://www.redhat.com/cms/managed-files/monolithic-vs-microservices.png>. Acesso em: 27/2/2020.

Aplicativos Web

Os aplicativos web, em sua maioria, são compostos por uma interface com o usuário, normalmente desenvolvido em linguagens de programação Javascript e linguagem de marcação HTML, que são interpretadas e executadas no navegador web, ou seja, no lado do cliente. Para responder a requisições dos clientes, são criados scripts em linguagens de programação reconhecidas pelos servidores web, como PHP, ASPX, Python, Ruby etc. Essas linguagens poderão acessar os bancos de dados, na maioria das vezes banco de dados relacionais, compostos por tabelas. O aplicativo do lado do servidor, um monólito, ou seja, único executável lógico, manipula solicitações HTTP, executa a lógica do domínio, recupera e atualiza dados do banco de dados e seleciona e preenche as visualizações HTML a serem enviadas ao navegador. Qualquer alteração no sistema representará a construção e implantação de uma versão nova do aplicativo ao lado do servidor.

Essa é a forma mais comum de se abordar aplicações monolíticas, seja ele desenvolvido ou executado em diversas plataformas, o importante é analisar o conceito. Numa aplicação monolítica, toda solicitação será executada ou processada em um único processo. Eles até podem ser bem-sucedidos quando criados seguindo boas práticas

de engenharia e desenvolvimento de software ou seguindo desenvolvimento modular, porém, à medida que surge a necessidade de implementação de novos processos ou a necessidade de serem executados em nuvens, aparecem algumas frustrações.

Os microsserviços permitem que um aplicativo monolítico e complexo seja desenvolvido como um grupo de pequenos serviços, agrupados por áreas de negócios específicas. Entretanto, quando estão em funcionamento, essa divisão em pequenos serviços é transparente aos usuários, sendo apresentada como uma única aplicação. Seus principais benefícios são:

- » código menos complexos;
- » fácil gerenciamento de aplicações;
- » melhoria em processos de testes;
- » escalabilidade otimizada;
- » melhora na disponibilização;
- » ciclos de operações otimizados;
- » entendimento entre a equipe de negócios e a equipe de desenvolvimento.

A Groupon possui um exemplo interessante de desmanche de uma aplicação monolítica para recursos de microsserviços. Segundo artigo o “Dismantling the Monolith”, disponível em: <https://engineering.groupon.com/2013/misc/i-tier-dismantling-the-monoliths/>, tudo começou quando o código de front-end cresceu rapidamente, dificultando a manutenção e o fornecimento de novos recursos. A solução foi dividir a aplicação em partes menores, aproveitando ao máximo do conceito de microsserviços.

Em exemplos relacionados à codificação, equipes poderão trabalhar em códigos de domínios diferentes. Com isso, não existe interferência direta na codificação, e as operações de testes poderão ser efetuadas em pequenos blocos. Para iniciar a criação de um aplicativo baseado em microsserviços, é necessário ter noção da separação das funcionalidades baseadas em domínios, verificar se cada aplicativo é implementável, testável individualmente, se os próprios microsserviços podem ser melhorados e “quebrados” em partes menores. Após verificar se os conceitos para o desenvolvimento de aplicativos em microsserviços estão bem claros, é hora de definir o conceito de arquitetura.

A Open Group, que é um consórcio global que permite alcançar os objetivos de negócios por meio de padrões de tecnologia, define a MSA (Micro Services Architecture) como:

um estilo de arquitetura que define e cria sistemas através do uso de pequenos serviços independentes e independentes, alinhados estreitamente com as atividades comerciais. Um microserviço individual é um serviço implementado com uma única finalidade, independente e independente de outras instâncias e serviços.

Quando uma aplicação não segue o padrão MSA, o ciclo de alterações e correções exige um esforço muito grande de mudanças. Se essa aplicação for modularizada, porém ainda não seguir o padrão MSA, ela tende a se enfraquecer com o tempo. Quando esse aplicativo estiver fortemente acoplado, vai requerer todo o dimensionamento da solução do aplicativo.

A cultura DevOps ressalta a natureza colaborativa do desenvolvimento e manutenção de aplicativos. Concentra-se na eliminação de silos entre desenvolvimento e operações, ou seja, o uso de equipes autônomas que detém todo o ciclo de vida de um aplicativo. A natureza independente e distribuída dos microserviços adequa-se naturalmente à implementação nessa cultura.

A integração contínua (CI) e a implantação contínua (CD) precisam da integração do software com a implantação automatizada do código integrado. A modularidade e a independência inerentes a um MSA e seus serviços servem muito bem aos paradigmas de CI e CD executados por diversas equipes distribuídas que trabalham com uma base de código controlada.

As principais características propostas pelo Open Group, que definem um MSA, são:

- » **Independência de Serviço:** ao identificar e isolar os serviços que apresentam constante rotatividade, a independência minimiza o impacto na infraestrutura de serviços. Assim que identificados, esses serviços devem ser atualizados ou substituídos sem nenhuma alteração adicional no cenário do software.
- » **Responsabilidade Única:** para responsabilidade única, deve ocorrer o alinhamento direto de um serviço a uma atividade única. O serviço deve mapear completamente a atividade e entregar a lógica necessária para cumprir a atividade. Quando cada serviço é associado a uma única atividade de negócios, torna-se possível o rastreamento do impacto das mudanças nos negócios por meio do cenário de serviços de software.
- » **Autocontenção:** a contenção determina que um serviço deve abranger todos os recursos externos de TI (por exemplo, fontes de dados, regras de negócios) necessários para apoiar a atividade de negócios. Além

disso, a autocontenção exige que as dependências de serviço que estejam fora do escopo da equipe de desenvolvimento sejam minimizadas ou preferencialmente eliminadas. Ao aderir à autocontenção, os serviços são mais facilmente substituíveis e atualizáveis. Juntamente com a responsabilidade única, essa característica também promove a propriedade única do serviço, uma vez que engloba não apenas requisitos de negócios específicos (responsabilidade única), mas também dependências de software específicas (autocontenção). Os microsserviços são fornecidos com contêineres e componentes como unidades de implantação única. Portanto, para alcançar a autocontenção, é essencial uma granularidade adequada na decomposição.

- » **Altamente dissociado:** para manter dependências mínimas de serviço, os microsserviços devem ser altamente dissociados.. Para conseguir esse desacoplamento, a função comercial deve ser capaz de ser decomposta até o nível em que um microsserviço está implementando uma única função comercial atômica. É essa decomposição e a consequente remoção de dependências entre as funções de negócios atômicas que permitem a independência do serviço e a autocontenção necessária em um MSA.
- » Ao confiar fortemente em protocolos existentes, cujo único objetivo é rotear mensagens (HTTP, por exemplo), as necessidades de um serviço podem ser reduzidas para receber a solicitação de mensagem, aplicar a lógica de negócios apropriada e gerar uma resposta de mensagem. A MSA desencoraja o uso de serviços de transformação de dados externos ou de ponte de protocolo, pois eles introduzem dependências de serviço adicionais fortemente acopladas que precisam ser gerenciadas durante as atualizações. A MSA endossa o conceito de terminais inteligentes, em que toda a lógica necessária para gerenciar a solicitação recebida e produzir uma resposta apropriada permanece encapsulada no serviço.
- » **Altamente resiliente:** um MSA deve ser uma arquitetura altamente resiliente. Seus microsserviços devem ser projetados para possíveis falhas, porque as falhas individuais de serviço não devem afetar negativamente a experiência do usuário. Como um microsserviço representa uma única responsabilidade e é independente, uma falha no serviço pode significar que determinada função ou processo comercial não pode ser concluído com êxito. O tempo de inatividade prolongado do serviço também pode ter um impacto significativo em todo o negócio. Portanto, mecanismos devem estar em vigor para garantir a recuperação oportuna do serviço.

O monitoramento de serviços em tempo real pode fornecer um meio proativo para identificar serviços que estão sofrendo com muita carga ou incapazes de satisfazer os SLAs (acordos de nível de serviço) existentes. O monitoramento em tempo real das métricas de negócios também fornece informações sobre alterações futuras nas atividades de negócios que se estendem aos serviços de TI.

CAPÍTULO 4

Segurança

Após o projeto de migração para a nuvem ter sido elaborado, todo o parque de TI desenhado e definido, os orçamentos, os tipos de serviços como SaaS, IaaS e PaaS já definidos, imediatamente a próxima pessoa a demonstrar esse cenário será o responsável pela segurança da informação. É primordial que ele acompanhe o projeto e a implementação dos serviços que serão executados.

Conceitos básicos de segurança da informação

A segurança em nuvem tem seu propósito definido nos conceitos básicos, do alicerce da segurança da informação, seja ela local ou em nuvem. A Segurança da Informação trata do conjunto de controles e processos que visam preservar os dados que trafegam ou são armazenados em qualquer meio e seu propósito **é** proteger as informações registradas, sem importar onde estejam situadas. Seus pilares são sustentados por:

Integridade

Uma informação íntegra é uma informação que não foi alterada de forma indevida ou não autorizada. A quebra de integridade ocorre quando a informação é corrompida, falsificada ou indevidamente alterada.

Confidencialidade

Garante que somente pessoas autorizadas tenham acesso às informações; perda de confidencialidade significa perda de segredo.

As informações têm diferentes graus de confidencialidade, normalmente relacionadas aos seus valores:

- » confidencial;
- » restrito;
- » sigiloso;
- » público.

Disponibilidade

Além de cuidar para que a informação chegue aos destinatários certos e de forma íntegra, a SI deve fazer com que ela esteja disponível quando for requisitada.

Assim, o ambiente tecnológico e os suportes da informação deverão estar funcionando corretamente para que a informação armazenada neles e que por eles transita possa ser utilizada pelos usuários.

O principal objetivo da Segurança da Informação é proteger os Ativos de TI, que é todo elemento que manipula a informação, inclusive ela mesma, passando pelo seu emissor, o meio pelo qual ela é transmitida ou armazenada, até chegar a seu receptor: as informações, os equipamentos e os sistemas que oferecem suporte às informações, às pessoas que as utilizam.

Sempre que o pessoal envolvido na segurança da informação analisa um parque tecnológico, é necessário identificar os riscos aos quais estão submetidos os ativos de uma organização, ou seja, saber qual é a probabilidade de que as ameaças se concretizem e o impacto que elas causarão ao negócio. As ameaças podem se tornar realidade por meio de vulnerabilidades. A relação ameaça X vulnerabilidade X impacto é a principal condição que deve ser levada em conta no momento de priorizar ações de segurança para a proteção dos ativos mais importantes para empresa.

Para colocar em prática as diretrizes de segurança da informação, é necessário implantar uma Política de Segurança da Informação. Uma política de segurança é um conjunto de diretrizes, normas, procedimentos e instruções que comanda as atuações de trabalho e define os critérios de segurança para que sejam adotados com o objetivo de estabelecer, padronizar e normatizar a segurança e seus processos tanto no escopo humano quanto no tecnológico. A partir desses princípios, é possível fazer da segurança das informações um esforço comum.

Para elaborar uma política de segurança das informações, é importante levar em consideração as exigências básicas e as etapas necessárias para a sua produção, ou seja, primeiramente é necessário formar uma equipe multidisciplinar que represente grande parte dos aspectos culturais, técnicos e administrativos da empresa e que se reúna periodicamente dentro de um cronograma pré-estabelecido. Esse comitê é formado por um grupo de pessoas responsáveis por atividades referentes à criação e aprovação de requisitos e demandas de segurança na empresa.

Segurança como serviço

A segurança como serviço (SecaaS ou SaaS) é um modelo de computação em nuvem que fornece serviços de segurança gerenciados pela Internet. O SecaaS é baseado no modelo de software como serviço (SaaS), mas limitado a serviços especializados de segurança da informação.

A segurança como serviço abrange o software de segurança entregue na nuvem, bem como o gerenciamento de segurança interno oferecido por terceiros. Algumas das soluções disponíveis podem ser encontradas em várias categorias, conforme descrito pela Cloud Security Alliance em <https://downloads.cloudsecurityalliance.org/assets/research/security-as-a-service/csa-categories-securities-prep.pdf>, acesso em 1º out. 2019:

- » **Recuperação de desastres e continuidade dos negócios:** ferramentas que ajudam você a garantir que sua TI e operações estejam de volta sem que as pessoas percebam que ocorreu um desastre.
- » **Monitoramento contínuo:** ferramentas que permitem gerenciar riscos continuamente, monitorando os processos de segurança existentes.
- » **Prevenção de perda de dados:** ferramentas que protegem, monitoram e verificam a segurança de todos os seus dados, estejam eles em armazenamento ou em uso.
- » **Segurança de e-mail:** protege sua empresa contra phishing, spam e anexos maliciosos.
- » **Criptografia:** torna seus dados ilegíveis, a menos que sejam decodificados usando as cifras numéricas e criptográficas corretas.
- » **Gerenciamento de identidade e acesso:** fornece autenticação, inteligência de acesso e identifica ferramentas de verificação e gerenciamento de usuários.
- » **Gerenciamento de intrusão:** detecta eventos e comportamentos incomuns usando a tecnologia de reconhecimento de padrões. Essas ferramentas não apenas detectam invasões, mas também ajudam a gerenciá-las.
- » **Segurança de rede:** ferramentas e serviços que ajudam a gerenciar o acesso à rede e distribuir, proteger e monitorar serviços de rede.

- » **Avaliação de segurança:** audita as medidas de segurança atuais em vigor para verificar se elas são compatíveis com os padrões do setor.
- » **Informações de segurança e gerenciamento de eventos:** ferramentas que agregam informações de log e eventos, que podem ser analisadas em tempo real para ajudar a detectar possíveis anomalias e invasões.
- » **Verificação de vulnerabilidades:** detecta qualquer vulnerabilidade na sua rede ou infraestrutura de TI.
- » **Segurança na Web:** oferece proteção para aplicativos on-line acessados pelo público em tempo real.

Secure Development Lifecycle

Um SDL não é um processo específico, com regras rígidas, qualquer desenvolvedor pode seguir algum já preestabelecido ou criar o seu. A vantagem de seguir algo pronto é que muitas pessoas testam, dão seu feedback e melhoram. Um exemplo de um SDL que acabou se tornando algo que o mercado utiliza, devido à sua maturidade, é o SDL da Microsoft.

O SDL da Microsoft foi lançado em resposta ao famoso memorando de Bill Gates de janeiro de 2002, no qual Gates estabeleceu o requisito de aumentar a segurança dos produtos da Microsoft, admitindo que, devido a vários surtos de vírus e malwares, a Microsoft precisava incorporar segurança para que fosse levada a sério no mercado. O SDL pode ser acessado pelo link: www.microsoft.com/sdl.

Segundo a Microsoft:

O objetivo do SDL é fornecer uma estrutura *simples* para a inclusão pragmática de práticas de segurança no processo de desenvolvimento de software. Ele descreve uma série de atividades distintas e não proprietárias de desenvolvimento de segurança que, quando combinadas com o processo de automação eficiente e uma orientação de política sólida, representam as etapas necessárias para haver conformidade objetiva de uma organização com o Microsoft SDL, conforme definido pelo nível “avançado” no Modelo de otimização do SDL.

O Microsoft SDL é uma coleção de atividades de segurança obrigatórias, apresentadas na ordem em que devem ocorrer e agrupadas por fases do ciclo de vida do desenvolvimento de software tradicional. As atividades opcionais de

segurança podem ser adicionadas a critério da equipe do projeto ou do consultor de segurança para atingir os objetivos desejados de segurança e de privacidade. Para ser breve, a discussão detalhada de cada atividade de segurança será omitida. Suas fases são:

- » **Requisitos de treinamento:** todos os membros de uma equipe de desenvolvimento devem receber um treinamento apropriado para se manterem informados sobre os fundamentos básicos de segurança e sobre as tendências de segurança e de privacidade.
- » **Requisitos de segurança:** a análise de requisitos de segurança e de privacidade é realizada no primeiro esboço do projeto e inclui a especificação dos requisitos de segurança mínimos para o aplicativo, conforme ele é criado para execução em seu ambiente operacional planejado, e a especificação e implantação de um sistema de monitoramento de vulnerabilidade de segurança/itens de trabalho.
- » **Quality Gates/Bug Bars:** são usados para estabelecer níveis mínimos aceitáveis de qualidade de segurança e de privacidade. A definição desses critérios no início de um projeto melhora o entendimento dos riscos associados aos problemas de segurança e permite que as equipes identifiquem e corrijam os erros de segurança durante o desenvolvimento.
- » **Avaliação de riscos de segurança e de privacidade:** as SRAS (avaliações de riscos de segurança) e as PRAS (avaliações de riscos de privacidade) são processos obrigatórios que identificam os aspectos funcionais do software que requerem uma análise profunda.
- » **Requisitos de design:** o momento ideal para influenciar a confiabilidade do design de um projeto é o início de seu ciclo de vida. A mitigação dos problemas de segurança e de privacidade é muito mais barata quando realizada durante os estágios iniciais do ciclo de vida de um projeto.
- » **Redução de superfície de ataque:** a redução da superfície de ataque está intimamente alinhada com a modelagem de ameaças, embora ela aborde as questões de segurança sob uma perspectiva um pouco diferente.
- » **Modelagem de ameaças:** a modelagem de ameaças é usada em ambientes nos quais há um risco de segurança significativo. É uma prática que permite que as equipes de desenvolvimento considerem, documentem

e discutam as implicações de segurança de designs no contexto de seu ambiente operacional planejado e de uma maneira estruturada.

- » **Utilizar ferramentas aprovadas:** todas as equipes de desenvolvimento devem definir e publicar uma lista de ferramentas aprovadas e suas verificações de segurança associadas, como as opções e os avisos de compilador/vinculador. Essa lista deve ser aprovada pelo consultor de segurança da equipe do projeto.
- » **Desaprovar funções não seguras:** muitas funções e APIs comumente usadas não são seguras ante o ambiente atual de ameaças. As equipes de projeto devem analisar todas as funções e APIs que serão usadas em conjunto com um projeto de desenvolvimento de software e proibir aquelas que estão determinadas como não seguras.
- » **Análise estática:** as equipes de projeto devem realizar análises estáticas de código-fonte. Essa análise fornece uma capacidade escalável de revisão de código de segurança e pode ajudar a assegurar que as políticas de codificação estejam seguras
- » **Análise de programa dinâmica:** a verificação de tempo de execução de programas de software é necessária para garantir que a funcionalidade de um programa funcione como planejado. Essa tarefa de verificação deve especificar as ferramentas que monitoram o comportamento do aplicativo quanto à corrupção da memória, problemas de privilégio do usuário e outros problemas de segurança críticos.
- » **Teste de fuzzing:** o teste de fuzzing é uma forma especializada de análise dinâmica usada para induzir a falha do programa ao introduzir deliberadamente dados defeituosos ou aleatórios a um aplicativo.
- » **Modelo de ameaças e revisão da superfície de ataque:** é comum para um aplicativo desviar significativamente das especificações funcionais e de design criadas durante as fases de requisitos e de design de um projeto de desenvolvimento de software.
- » **Plano de resposta de incidentes:** cada liberação de software sujeita aos requisitos do SDL deve incluir um plano de resposta de incidentes. Mesmo os programas sem vulnerabilidades conhecidas no momento da liberação podem estar sujeitos a novas ameaças que surgem com o tempo.

- » **Revisão final de segurança:** a FSR (Revisão final de segurança) é um exame deliberado de todas as atividades de segurança realizadas em um aplicativo de software antes da liberação.
- » **Liberar/Arquivar:** a liberação do software para fabricação (RTM) ou liberação para a web (RTW) depende da conclusão do processo do SDL.

Referências

BEAL 2004, A. **Gestão Estratégica da Informação**: como transformar a informação e a tecnologia da informação em fatores de crescimento e de alto desempenho nas organizações. São Paulo: Atlas, 2004.

CECCHET, E.; SINGH, R.; SHARMA, U.; SHENOY, P. (2011). **Dolly**: virtualization-driven database provisioning for the cloud. *In*: ACM Sigplan/Sigops International Conference on Virtual Execution Environments, VEE. https://www.researchgate.net/publication/221137821_Dolly_Virtualization-driven_Database_Provisioning_for_the_Cloud_-_UMass.

DOMINICO, S. **Provisionamento de Recursos Computacionais baseado em Redes de Petri para Bancos de Dados Orientados a Leitura**. Curitiba: Universidade Federal do Paraná, 2016.

FERNANDO, N.; LOKE SENG, W.; RAHAYU, W. **Mobile cloud computing**: a survey. Australia: Department of Computer Science and Computer Engineering, La Trobe University, 2012.

ISACA 2012. **Cálculo do ROI da nuvem**: a perspectiva do cliente. Disponível em: http://www.isaca.org/Knowledge-Center/Research/Documents/Calculating-Cloud-ROI_whp_Por_0712.pdf. USA: 2012. Acesso em: 1º out. 2019.

MELL, P.; GRANCE, T. **The NIST Definition of Cloud Computing**, Gaithersburg, MD: National Institute of Standards and Technology, 2011. Disponível em: <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>. Acesso em: 7 out. 2019.

SCHUBERT, F.; ROLIM, C. O.; WESTPHALL, C. B. (2011). Aplicação de algoritmos de provisionamento baseados em contratos de nível de serviço para computação em nuvem. *In*: **IX Workshop em Clouds, Grids e Aplicações**. Disponível em: https://www.researchgate.net/publication/247935902_Aplicacao_de_Algoritmos_de_Provisionamento_Baseados_em_Contratos_de_Nivel_de_Servico_para_Computacao_em_Nuvem. Acesso em: 1º out. 2019.

SLA Management Handbook – Volume 4: Enterprise Perspective. ISBN: 1-931624-51-8 Document Number: G045 This document is equivalent to TMF document reference GB917, Version 2.0, Volume 4. Published by The Open Group, October 2004.

SOUZA, Jano Moreira de. **Strategic Planning for Information Technology: a collaborative model of information technology strategic plan for the government sector.** Rio de Janeiro: Universidade Federal, 2018.

The Green Grid, PUE: a comprehensive examination of the metric White paper 49. (2012) Disponível em: <https://www.thegreengrid.org/file/458/download?token=R6cW2uzD>. Acesso em: 28 out. 2019.

VERAS, Manuel. **Computação em Nuvem.** Rio de Janeiro: Brasport, 2015.

Sites

<https://www.eveo.com.br/ebooks/Guia-Tributacao-Data-Center-Exterior.pdf>.

<http://www.inf.ufpr.br/sbbd-sbsc2014/sbbd/proceedings/artigos/pdfs/27.pdf>.

http://www.repositorio.ufc.br/bitstream/riufc/16374/1/2016_dis_vaefarias.pdf.

<https://azure.microsoft.com/pt-br/>.

<https://blog.rocketseat.com.br/saas-single-tenant-multi-tenant/>.

<https://blogs.technet.microsoft.com/lpalma/2008/10/01/paravirtualizacao-emulao-bare-metal/>.

<https://cloudsecurityalliance.org/research/guidance/>.

<https://developers.redhat.com/blog/2017/05/04/the-truth-about-microservices/>.

<https://downloads.cloudsecurityalliance.org/assets/research/security-as-a-service/csa-categories-securities-prep.pdf>.

<https://restfulapi.net/>.

<http-status-codes/>.

<https://searchcio.techtarget.com/feature/Cloud-migration-failures-and-how-to-prevent-them>.

<https://searchcloudcomputing.techtarget.com/definition/Software-as-a-Service>.

<https://servicodados.ibge.gov.br/api/docs/localidades?versao=1>.

https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Data_Center/DC_Infra2_5/DCInfra_1.html.

<https://www.infoworld.com/article/3226386/what-is-saas-software-as-a-service-defined.html>.

https://www.isaca.org/Journal/archives/2014/Volume-3/Documents/Securing-an-Evolving-Cloud-Environment_joa_Eng_0514.pdf.

<https://www.prppg.ufpr.br/siga/visitante/trabalhoConclusaoWS?idpessoal=28318&idprograma=40001016034P5&anobase=2016&idtc=1305>.

https://www.researchgate.net/publication/247935902_Aplicacao_de_Algoritmos_de_Provisionamento_Baseados_em_Contratos_de_Nivel_de_Servico_para_Computacao_em_Nuvem.

<https://www.techradar.com/news/what-is-saas>.

<https://www.thegreengrid.org/file/458/download?token=R6cW2uzD>.

<https://www.w3.org/Protocols/rfc2616/rfc2616.html>.

<https://www.wired.com/2002/01/bill-gates-trustworthy-computing/>.

<https://www.redhat.com>.

<https://azurecomcdn.azureedge.net>.

<https://azurecomcdn.azureedge.net/cvt-add6f4a23186e176ede2ce0357a4c26bb9352332c93c37a18e80bd3245a301b9/images/page/overview/what-is-paas/what-is-paas.png>. Acesso em: 7 out. 2019.

<https://www.redhat.com/cms/managed-files/monolithic-vs-microservices.png>.

<https://nsl.cs.sfu.ca/teaching/13/880/mobileCloudSurvey.pdf>.

<https://www.fedramp.gov/>.

<https://www.ipm.com.br/blog/historia-da-computacao-em-nuvem-como-surgiu-a-cloud-computing>.

<https://www.pmi.org/learning/library/strategic-alignment-connecting-between-strategy-execution-5962>.

<https://pubs.opengroup.org/onlinepubs/9299959499/toc.pdf>.

REFERÊNCIAS

<https://blogs.technet.microsoft.com/lpalma/2008/10/01/paravirtualizacao-emulao-bare-metal>.

<https://azure.microsoft.com/pt-br/services/postgresql/>.

https://aws.amazon.com/pt/rds/?nc2=h_ql_prod_db_rds.

https://www.ibm.com/support/knowledgecenter/zosbasics/com.ibm.zos.zmainframe/zconc_whatismainframe.htm.

<https://www.google.com/about/datacenters/efficiency/how/>.

<https://uptimeinstitute.com/resources/asset/2019-data-center-industry-survey>.

https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Data_Center/DC_Infra2_5/DCInfra_1.html.

<https://developer.ibm.com/articles/deciding-between-native-and-cross-platform-mobile-frontend-programming-frameworks/>.

<https://engineering.groupon.com/2013/misc/i-tier-dismantling-the-monoliths/>.