

Computação em Nuvem e Provisionamento na *Amazon Web Services* *Cloud Computing and Provisioning on Amazon Web Services*

Gustavo Oliveira Mota¹
gustavo.mota@fatec.sp.gov.br

Cristina Corrêa de Oliveira¹
cristina.oliveira@fatec.sp.gov.br

1 – Faculdade de Tecnologia da Zona Leste | Fatec Zona Leste

Resumo: A computação em nuvem tem chamado a atenção por ter como propósito solucionar grandes problemas enfrentado por empresas, como manter suas aplicações disponíveis em cenários de demanda alta, armazenar seus dados de forma segura e a redução de custos com infraestrutura. O presente artigo tem como finalidade explorar o conceito de computação em nuvem, abordando a base da computação em nuvem, suas principais características, modelos de serviço, modelos de implantação, benefícios e limitações. Após o provisionamento de uma interface de programação de aplicação na nuvem utilizando a *Amazon Web Services* como provedor, seguindo uma arquitetura *serverless* com *API Gateway*, uma função *Lambda* e banco de dados não relacional *Amazon DynamoDB*, foi possível realizar testes de carga utilizando *K6*, e análises qualitativas e quantitativas a fim de verificar, de forma prática, alguns benefícios e desafios ao optar pela computação em nuvem. A computação em nuvem é uma ferramenta poderosa, mas requer atenção a critérios como segurança, conformidade e privacidade. Além disso, deve-se escolher o modelo de serviço e o modelo de implantação que melhor atenda às necessidades específicas de uma organização.

Palavras-chave: Computação em nuvem, Nuvem pública, Função como serviço.

Abstract: *Cloud computing has drawn attention for its purpose of addressing significant challenges faced by businesses, such as keeping their applications available in scenarios of high demand, securely storing their data, and reducing costs associated with infrastructure. This article aims to explore the concept of cloud computing, covering the basics of cloud computing, its key characteristics, service models, deployment models, benefits, and limitations. After provisioning a cloud-based application programming interface using Amazon Web Services as the provider, following a serverless architecture with API Gateway, a Lambda function, and the Amazon DynamoDB NoSQL database, load tests were conducted using K6, along with qualitative and quantitative analyses to practically assess some benefits and challenges of opting for cloud computing. Cloud computing is a powerful tool but requires attention to criteria such as security, compliance, and privacy. Additionally, choosing the service model and deployment model that best meets the specific needs of an organization is crucial.*

Keywords: *Cloud computing, public cloud, function as a service.*

Recebido em
20 nov. 2023

Aceito em
15 fev. 2024

Publicado em
27 mar. 2024

<https://git.fateczl.edu.br>
e_ISSN
2965-3339
DOI
10.29327/2384439.2.2-1

@_GIT
Advances in Global
Innovation & Technology
Volume 2
Número 2
São Paulo
Março
2024



1. INTRODUÇÃO

A cada dia as empresas vem se deparando com problemas como falta de escalabilidade dos seus sistemas pois os servidores não atendem o aumento da demanda, levando a empresa a adquirir mais servidores que geram mais consumo de energia e necessitam ser protegidos contra interrupções elétricas. Além disso, a aquisição e configuração desses servidores podem ser processos demorados, atrasando o início de projetos e a disponibilização de um produto ou serviço no mercado.

Vários nichos de mercado apresentam aumento de demanda devido a sazonalidade do seu negócio, fazendo que a infraestrutura de servidores seja subutilizada, gerando alto custo de propriedade com espaço físico, manutenções variadas e segurança física e lógica. A influência da atividade humana tem contribuído para um aumento na frequência e na intensidade de desastres naturais, como tempestades, enchentes, desmoronamentos, queda de árvores, deslizamentos de terra e incêndios. Essa tendência representa uma ameaça significativa para as empresas, pois há o risco de perder os servidores que abrigam informações essenciais para suas operações. Diante dos desafios mencionados, a computação em nuvem (CN) surge como resposta às necessidades das empresas em um ambiente em constante evolução, proporcionando uma abordagem mais flexível, econômica e eficaz para a gestão de recursos de Tecnologia da Informação (TI) e para atender às crescentes exigências do mundo dos negócios.

Este artigo tem como objetivo explorar o conceito de CN, abordando suas principais características, benefícios e limitações.

Além disso, será provisionado uma aplicação na *Amazon Web Services* no modelo *serverless* e a realização de testes de carga para avaliar seu desempenho. A análise do provisionamento, do funcionamento da aplicação e a apresentação de métricas destacarão as vantagens práticas da adoção da CN.

2. FUNDAMENTAÇÃO TEÓRICA

O termo *cloud computing* ou CN começou a ser amplamente utilizado no final da década de 2000, embora as ideias e conceitos que o fundamentam tenham raízes em décadas anteriores. A seguir serão apresentadas as bases da CN, a definição, as características, os modelos de serviços e de implantação, bem como benefícios e limitações.

2.1. Bases do *Cloud Computing*

A CN é uma evolução de conceitos que surgiram nas décadas de 1950, até chegar no estado atual. A seguir são exemplificadas a evolução do conceito:

- Década de 1950: surgem os primeiros estudos em simulações de tempo real (RUBINOFF, 1955)
- Década de 1960: Os conceitos iniciais de compartilhamento de recursos computacionais remotos começaram a surgir (O'SULLIVAN, 1967). O modelo inicial foi conhecido como *time sharing* ou tempo compartilhado que envolvia a alocação de recursos computacionais compartilhados entre usuários em terminais remotos (MOTOBAYASHI, MASUDA e TAKAHASHI, 1969). O termo também se refere a sistemas multiusuários, em computadores de grande porte, ou mainframe, com a computação de propósito geral

executada e os usuários operam independentemente de um outro (O'SULLIVAN, 1967), geralmente em locais remotos do computador em si (CORBATÓ, 2003). Na década seguinte surgem os minicomputadores com o surgimento do Unix que apresenta uma diminuição do custo do hardware e do sistema operacional (RITCHIE e THOMPSON, 1973).

- Década de 1980 surge o microcomputador, com a diminuição dos custos do *hardware*, *software* e de processamento, e as redes de computadores (VAN TILBORG e WITTIE, 1981) (SOLNTSEFF, 1980). O conceito computação distribuída é utilizada academicamente.
- Década de 1990: o conceito de aplicação cliente/servidor se expande para o mundo corporativo. Este modelo consiste na separação do processo cliente e servidor executados em máquinas diferentes conectadas por meio de uma rede, podendo ser uma rede local ou uma rede remota. O processo cliente é ativo e envia uma solicitação ao servidor, que normalmente é dedicado a sessão do usuário, começando e terminando com a sessão. Um cliente pode interagir com um ou muitos servidores. Obrigatoriamente há a presença de um servidor. O servidor é um ente reativo, que é disparado pela chegada de pedidos dos seus clientes (RENAUD, 1994). O uso da expressão "*cloud computing*" para descrever esse conceito específico começou a ganhar popularidade no final dos anos 90. No entanto, ainda era uma ideia emergente e não tão amplamente adotada como hoje.
- Início dos anos 2000: Grandes empresas de tecnologia, como

Amazon, começaram a oferecer serviços de computação e armazenamento pela internet. A ideia de fornecer recursos de TI como serviço começou a se consolidar.

- Meados da Década de 2000: O termo "*cloud computing*" começou a ser mais amplamente adotado pela indústria de tecnologia e por empresas que ofereciam serviços baseados na nuvem.
- Final da Década de 2000: Empresas como *Amazon Web Services* (AWS) e Google começaram a oferecer uma ampla gama de serviços em nuvem. O conceito de "infraestrutura como serviço" (IaaS), "plataforma como serviço" (PaaS) e "software como serviço" (SaaS) se tornaram mais definidos.

Desde então, a computação em nuvem tornou-se uma parte fundamental da infraestrutura de TI em muitas organizações, e o termo "*cloud computing*" é agora amplamente reconhecido e adotado globalmente. O crescimento contínuo da computação em nuvem está ligado à sua capacidade de oferecer flexibilidade, escalabilidade e eficiência em termos de custo para indivíduos, empresas e organizações. Estes termos serão explorados nos tópicos a seguir.

2.2. Computação em Nuvem

A inovação que a CN trouxe resultou em várias interpretações do conceito. A seguir, será apresentado as definições fornecidas pelos três principais provedores de serviços em nuvem.

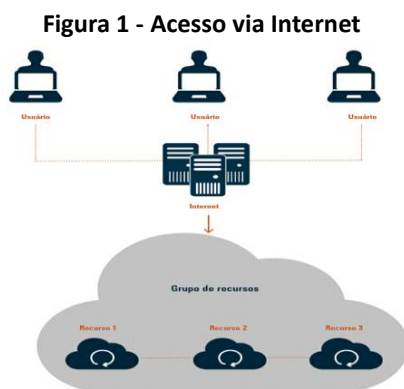
De acordo com o *Google Cloud*, a CN é a disponibilidade sob demanda dos recursos de computação como serviços na Internet. Ela elimina a necessidade de as empresas

adquirirem, configurarem ou gerenciarem a infraestrutura, assim elas pagarão apenas pelo que usarem (GOOGLE CLOUD, 2023b).

Para a *Microsoft Azure*, ela é o fornecimento de serviços de computação, incluindo servidores, armazenamento, bancos de dados, rede, *software*, análise e inteligência, pela Internet (“a nuvem”) para oferecer inovações mais rápidas, recursos flexíveis e economias de escala. Paga-se apenas pelos serviços de nuvem que usa, ajudando a reduzir os custos operacionais, a executar sua infraestrutura com mais eficiência e a escalonar conforme as necessidades da sua empresa mudam (MICROSOFT AZURE, 2023).

A AWS define a CN como a entrega de recursos de TI sob demanda por meio da Internet com definição de preço de pagamento conforme o uso. Em vez de comprar, ter e manter datacenters e servidores físicos, o acesso a serviços de tecnologia, como capacidade computacional, armazenamento e bancos de dados, é feito conforme a necessidade, usando um provedor de nuvem (AWS, 2023a).

Ambas descrevem a CN como um paradigma que permite acesso via internet a um grupo de recursos físicos ou virtuais, com provisionamento e administração sob demanda. A Figura 1 representa essa definição visualmente.



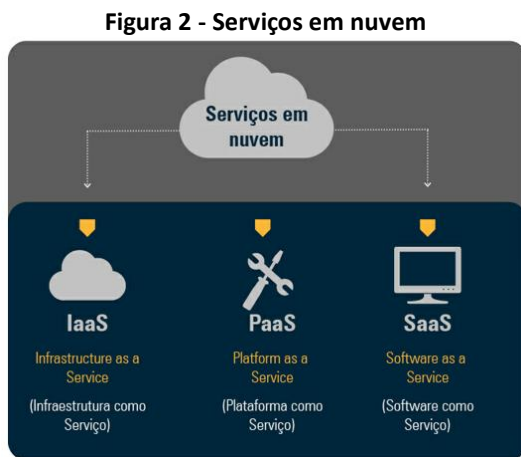
Fonte: SANTOS (2018)

- **Ampla acesso à rede:** essa característica se refere a uma condição em que os recursos devem estar disponíveis via internet e de maneira padronizada, garantindo, assim, a possibilidade de uso por meio dos mais diversos tipos de dispositivos (computadores, *smartphones*, *tablets*, etc.). É um acesso onipresente e multiplataforma.
- **Grupo de recursos:** nessa condição, os provedores de CN agrupam recursos, possibilitando servir múltiplos clientes dentro de um modelo compartilhado. Isso significa que uma única plataforma pode servir vários clientes de acordo com a demanda, sem que um usuário afete o outro.
- **Rápida elasticidade:** essa característica coloca que os recursos computacionais da nuvem, de maneira geral, devem ser provisionados e disponibilizados para uso de maneira rápida e elástica. Trata-se de uma condição em que há percepção da existência ilimitada dos recursos que podem ser adquiridos no momento e na quantidade desejada, com ampliação ou diminuição rápida, na medida da solicitação.
- **Serviços mensuráveis:** é uma condição em que os serviços e recursos fornecidos em nuvem são controlados e monitorados de maneira automática. Com essa espécie de visualização, otimizam-se os recursos, tornando possível aferir métricas. Ao ser mensurável, é possível estabelecer os limites de consumo e realizar o provisionamento automático dos recursos. Tal possibilidade permite, por exemplo, solicitar uma maior capacidade de processamento, armazenamento ou acessos, dentro de um período específico. Em geral, esses serviços são adquiridos na forma de

“pague por uso”. Nessa situação, o acréscimo será pago quando a cota previamente acordada for excedida.

2.4 Modelos de Serviço

Existem basicamente três tipos de modelos de serviço em nuvem para atender a diversas necessidades, conforme Figura 2.



Fonte: Santos (2018)

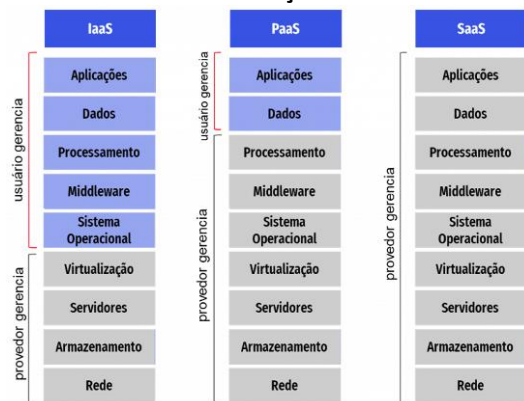
- *Infrastructure as a Service (IaaS)*: significa Infraestrutura como Serviço. Trata-se do serviço fornecido ao cliente, por meio da Internet, para processamento, armazenamento e outros recursos computacionais fundamentais. O consumidor não precisa comprar o equipamento, não se preocupa com o espaço físico, não gerencia ou controla a infraestrutura, mas pode atuar no sistema operacional, no armazenamento e na implementação das aplicações.
- *Platform as a Service (PaaS)*: significa Plataforma como Serviço. Trata-se de uma capacidade em que se pode, por exemplo, realizar a implementação e o desenvolvimento de um sistema próprio, dentro da infraestrutura do provedor de serviços. Nessa condição, o consumidor não gerencia ou controla

nenhum item da infraestrutura, tal como servidores, rede, sistemas operacionais ou armazenamento. Além disso, o consumidor não se preocupa com a escalabilidade, pois é gerenciada pelo provedor.

- *Software as a Service (SaaS)*: significa Software como Serviço. É um serviço que disponibiliza um programa que fica em um servidor remoto. Nesse caso, as aplicações são acessíveis por meio de dispositivos do cliente, com uso de interfaces simples. O usuário não gerencia ou controla nenhum item da infraestrutura que hospeda *software*. Somente é possível ajustar configurações específicas a nível de usuário.

A Figura 3 detalha a responsabilidade do usuário e do provedor em cada tipo de serviço de nuvem.

Figura 3 - Responsabilidades em cada modelo de serviço



Fonte: Autores

A criação de novos modelos está em constante crescimento. Nesse contexto, surgiu o termo "XaaS" ou *Everything as a Service*, que se refere à ideia de que praticamente qualquer tipo de serviço pode ser oferecido como um serviço (TOTVS, 2023). É importante destacar que esses modelos evoluíram a partir das categorias mencionadas anteriormente.

Abaixo, apresentamos alguns exemplos desses serviços, juntamente com suas características.

- *Function as a Service (FaaS)*: significa Função como Serviço. É um serviço de CN que permite que os clientes executem códigos em resposta a eventos, sem gerenciar a infraestrutura complexa normalmente associada à criação e ao lançamento de aplicativos de microsserviços (IBM, 2023).
- *Database as a Service (DBaaS)*: significa Bando de Dados como Serviço. É uma oferta de banco de dados em nuvem que fornece aos clientes acesso a um banco de dados sem ter que implantar e gerenciar a infraestrutura subjacente. O DBaaS é entregue como um serviço de banco de dados gerenciado, o que significa que o provedor cuida de corrigir, atualizar e fazer backup do banco de dados (ORACLE, 2023).

2.5 Modelos de Implantação

Existem três tipos de modelo de implantação: nuvem privada, nuvem pública e nuvem híbrida. Cada opção apresenta características únicas e vantagens específicas. Assim, a escolha do modelo de implantação deve se basear no tipo de aplicação que o cliente necessita. Conheça, a seguir, as características das formas de implementação de nuvem. A Figura 4 representa a definição visualmente.

Figura 4 - Modelos de implantação



Fonte: Autores

- **Nuvem privada**: é uma infraestrutura em nuvem desenvolvida, operada e gerenciada por uma única organização. Eventualmente, essa tarefa é compartilhada com uma empresa terceirizada. Pode ser instalada no ambiente da organização ou em um data center externo. Outros nomes de nuvem privada são nuvem interna ou nuvem corporativa.
- **Nuvem pública**: é uma infraestrutura em nuvem fornecida abertamente para o uso do público em geral. Pode ser desenvolvida, operada e gerenciada por uma única organização empresarial, acadêmica ou governamental, de modo separado ou combinado. Fica instalada no provedor de nuvem.
- **Nuvem híbrida**: é uma infraestrutura em nuvem composta pela combinação da nuvem pública e privada. A CN híbrida estende a infraestrutura e as operações de forma consistente para oferecer um único modelo operacional que gerencia as cargas de trabalho dos aplicativos nos dois ambientes, permitindo a migração contínua das cargas de trabalho entre as nuvens privada e pública, conforme as necessidades dos negócios.

2.6 Benefícios

A CN oferece uma série de benefícios para organizações e usuários (GOOGLE CLOUD, 2023c), incluindo:

- **Maior rapidez no tempo de lançamento**: É possível ativar novas instâncias ou desativá-las em segundos, permitindo que os desenvolvedores acelerem o desenvolvimento com implantações rápidas. A CN oferece suporte a inovações, facilitando o teste de novas ideias e o desenvolvimento de novos

aplicativos sem limitações de hardware ou processos de aquisição lentos.

- Escalabilidade e flexibilidade: Os recursos de CN podem ser escalados para cima ou para baixo conforme a demanda, permitindo que as organizações atendam a picos de tráfego e necessidades sazonais sem investimentos antecipados em infraestrutura.
- Economia de custos: A CN elimina a necessidade de comprar, manter e atualizar hardware e software, reduzindo os custos de capital. Os modelos de pagamento por uso permitem que as organizações paguem apenas pelos recursos que consomem, o que pode resultar em economias significativas. Um exemplo disso é a Natura, que conseguiu reduzir seus custos operacionais em impressionantes 40% ao adotar a CN (AWS, 2023b). O Mercado Livre também obteve economias notáveis de 31% ao otimizar o uso dos serviços fornecidos pela AWS, tudo isso enquanto mantinha a performance (AWS, 2023c).
- Melhor colaboração: O armazenamento na nuvem permite que a disponibilização de dados onde estiver, sempre que precisar. Em vez de estarem vinculados a um local ou dispositivo específico, as pessoas podem acessar dados de qualquer lugar do mundo e de qualquer dispositivo, desde que tenham uma conexão de Internet.
- Segurança avançada: A CN pode fortalecer sua postura de segurança devido à profundidade e amplitude de recursos de segurança, à manutenção automática e ao gerenciamento centralizado. Provedores de nuvem

confiáveis também contratam os melhores especialistas em segurança e empregam as soluções mais avançadas, oferecendo uma proteção mais robusta.

- Prevenção contra perda de dados: Os provedores de nuvem oferecem recursos de backup e recuperação de desastres. Armazenar dados na nuvem em vez de localmente pode ajudar a evitar a perda de dados em caso de emergência, como mau funcionamento do hardware, ameaças mal-intencionadas ou até mesmo um simples erro do usuário.

Esses benefícios tornaram a CN uma escolha popular para empresas de todos os tamanhos e setores, permitindo que elas se adaptem rapidamente às mudanças nas necessidades de TI e aproveitem as vantagens da tecnologia de forma eficaz e econômica.

2.7 Limitações

Apesar de todos os benefícios, é preciso considerar as limitações da CN antes de aderir (GOOGLE CLOUD, 2023a). A seguir, veja uma descrição mais detalhada a respeito dos pontos críticos para adoção da CN:

- Segurança: é um grande desafio da CN. A segurança e a privacidade dos dados estão no topo das preocupações relacionadas à TI. Isso é confirmado pela maioria das pesquisas realizadas no segmento. Tal receio do mercado se deve pelo fato de a CN colocar os riscos em outro nível, porque os serviços essenciais são, muitas vezes, subcontratados de terceiros. Nessa condição, pode ser mais difícil manter a integridade e a privacidade dos dados.

- Resultados e benefícios: uma parte dos clientes ainda não está convencida sobre as vantagens do modelo de serviço em nuvem. Há muita insegurança sobre como tornar os recursos da nuvem uma parte do ambiente principal conciliado ao retorno sobre investimento (ROI) e ao uso pleno do potencial tecnológico. Para essa situação, existe a necessidade de comparar as métricas financeiras da TI tradicional com a utilização da nuvem. Sem essa base, há grande dificuldade em compreender as vantagens econômicas das soluções disponíveis nesse novo modelo.
- Performance e tempo de resposta no acesso via rede: entregar um serviço complexo via rede é completamente impossível se a performance do acesso for ineficiente. Muitas empresas esperam a melhora na oferta e na qualidade dos links de acesso antes de moverem suas aplicações para a nuvem.
- Integração: a integração entre sistemas, aplicativos e nuvem pode ser muito complexa. Existem condições de incompatibilidade que geram conflitos técnicos de difícil solução. Em alguns casos, os sistemas já residentes da organização são incompatíveis para a migração ou associação com a nuvem. A evolução da padronização dos protocolos e procedimentos entre os tipos de nuvens ainda está em desenvolvimento, o que agrega dificuldades de integração.

3. MATERIAIS E MÉTODOS

Esta seção apresenta os procedimentos metodológicos utilizados no estudo com o objetivo de entender as vantagens do uso da CN em relação a provisionar nuvem

privada.

Foram estabelecidos no enquadramento metodológico: a natureza da pesquisa, o objetivo da pesquisa; a abordagem; o processo da implementação e, por fim, a análise dos resultados.

A natureza deste estudo é aplicada, pois objetiva gerar conhecimento para escolha do tipo de nuvem que mais se adequa à situação em questão. A pesquisa aplicada na área empresarial busca facilitar a tomada de decisões (ZIKMUND e BABIN, 2010).

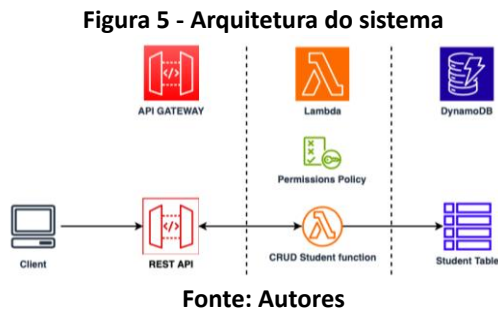
O objetivo, do presente artigo, é exploratório, pois se pretende construir um conhecimento sobre CN, abordando suas características, benefícios e funcionamento, facilitando a decisão de qual nuvem escolher para determinado contexto de negócio.

A abordagem deste estudo é qualitativa, pois pretende-se apresentar as vantagens do uso da nuvem pública em relação à nuvem privada.

Para apresentação dos resultados, será implementado uma aplicação capaz de realizar cadastros de alunos numa situação hipotética como a realização do Exame Nacional do Ensino Médio.

Após implementada e implantada na AWS, será realizado requisições utilizando uma ferramenta de testes de carga, denominada K6, e análise do funcionamento da aplicação, aferindo a taxa de sucesso das requisições. A aplicação foi criada a partir de um exemplo disponibilizado na AWS e seguirá a arquitetura apresentada visualmente na Figura 5 (AWS, 2023d).

Abaixo segue uma breve descrição de cada serviço utilizado na implementação e em qual modelo de serviço eles se encaixam.



3.1 Amazon API Gateway

O *Amazon API Gateway* é um serviço da AWS que atua como um ponto de entrada para criar, publicar, gerenciar e proteger APIs (Interfaces de Programação de Aplicativos) de maneira escalável e segura na nuvem. É uma ferramenta poderosa para desenvolvedores e organizações que desejam expor funcionalidades e dados por meio de APIs para acesso público ou privado (AWS, 2023e). Aqui estão alguns dos principais aspectos e funcionalidades do *Amazon API Gateway*:

- Criação de APIs: Permite a criação de APIs RESTful (*Representational State Transfer*) e *WebSocket* para expor serviços e recursos.
- Gerenciamento de Tráfego: Oferece recursos para rotear solicitações, transformar dados, fazer cache de respostas e melhorar o desempenho e a segurança.
- Integrações Flexíveis: Suporta integrações com vários serviços da AWS, bem como com serviços externos, permitindo que conecte APIs a diversas fontes de dados e funcionalidades.
- Segurança Avançada: Oferece opções de autenticação e autorização personalizáveis, bem como proteção contra ameaças.
- Documentação Automática: Gera documentação automática para suas APIs, tornando mais fácil para os

desenvolvedores entenderem como usá-las.

- Escalabilidade Automática: Escala automaticamente para lidar com o tráfego, garantindo que as APIs permaneçam disponíveis e responsivas.
- Monitoramento e Registro: Fornece métricas e registros integrados para monitorar o tráfego e o desempenho das APIs em tempo real.
- Gerenciamento de Versões: Permite criar e gerenciar várias versões de suas APIs, permitindo a evolução e o teste de novas funcionalidades.
- Uso Comum: O *Amazon API Gateway* é amplamente usado para criar aplicativos da web, móveis e sistemas distribuídos que dependem da exposição de APIs para se comunicar e compartilhar dados.

Em resumo, o *Amazon API Gateway* é uma solução completa para expor e gerenciar APIs na nuvem, tornando mais fácil para as organizações compartilharem funcionalidades e dados com segurança e eficiência. É um componente importante na criação de sistemas modernos baseados em microsserviços e é frequentemente usado em conjunto com outros serviços da AWS para desenvolver aplicativos escaláveis e seguros.

3.2 AWS Lambda

AWS Lambda é um serviço de computação *serverless* que se enquadra no modelo função como serviço (FaaS) oferecido pela AWS. Ele permite que os desenvolvedores executem código em resposta a eventos sem a necessidade de provisionar ou gerenciar servidores. O modelo *serverless* elimina a complexidade de gerenciar a infraestrutura e permite que os

desenvolvedores se concentrem apenas no código e na lógica do aplicativo (AWS, 2023f).

Aqui estão alguns dos principais aspectos do AWS Lambda:

- **Execução de Código Sem Servidor:** O código é executado automaticamente em resposta a eventos, como solicitações HTTP, alterações em bancos de dados, uploads de arquivos e muito mais.
- **Escalabilidade Automática:** O AWS Lambda dimensiona automaticamente a capacidade de execução de acordo com a demanda, garantindo que o código seja executado eficientemente, independentemente da carga.
- **Pagamento por Uso:** Pago apenas pelo tempo de execução do código, o que torna os custos previsíveis e eficientes.
- **Integração com Outros Serviços AWS:** *AWS Lambda* pode ser facilmente integrado com outros serviços da AWS, permitindo a criação de aplicativos e automações altamente escaláveis.
- **Suporte a Múltiplas Linguagens de Programação:** A escrita as funções Lambda pode ser em várias linguagens de programação, como Python, Node.js, Java, C#, Ruby e muito mais.
- **Tempo de Execução Personalizado:** Pode ser criado funções personalizadas com base em tempo de execução personalizado, o que permite a execução de código em contêineres.
- **Monitoramento e Registro:** *AWS Lambda* oferece métricas integradas e registros para monitorar o desempenho e o comportamento de suas funções.
- **Segurança:** As funções Lambda podem ser configuradas com políticas de segurança, controlando quem pode invocá-las e acessar os recursos

associados.

O *AWS Lambda* é amplamente utilizado em cenários que requerem respostas rápidas a eventos, automações, microsserviços, processamento de fluxo de dados em tempo real e muito mais. Ele simplifica o desenvolvimento e a implantação de código e é uma parte essencial da arquitetura de aplicativos modernos baseados em microsserviços e em uma abordagem *serverless*.

3.3 Amazon DynamoDB

O *Amazon DynamoDB* é um serviço de banco de dados NoSQL que se enquadra no modelo banco de dados como serviço (DBaaS) totalmente gerenciado fornecido pela *Amazon Web Services (AWS)*. Ele é projetado para fornecer armazenamento de dados altamente escalável e de alto desempenho, com acesso rápido e flexível a informações estruturadas e semiestruturadas (AWS, 2023g).

Aqui estão algumas das principais características e benefícios do *Amazon DynamoDB*:

- **Totalmente Gerenciado:** O *DynamoDB* é um serviço totalmente gerenciado, o que significa que a AWS cuida de todas as tarefas de administração, como provisionamento de recursos, escalabilidade, backups e manutenção.
- **Escalabilidade Automática:** O serviço dimensiona automaticamente para atender às demandas de tráfego, permitindo que os aplicativos mantenham um desempenho consistente, independentemente do volume de dados ou tráfego.
- **Modelo de Dados Flexível:** O *DynamoDB* suporta uma variedade de modelos de dados, incluindo documentos, colunas largas e tabelas

de chave-valor, o que permite flexibilidade na modelagem de dados.

- **Latência Baixa:** Oferece acesso a dados de baixa latência, tornando-o adequado para aplicativos que exigem respostas rápidas, como aplicativos da web e móveis em tempo real.
- **Segurança Avançada:** Oferece recursos de segurança avançados, como controle de acesso, criptografia de dados em repouso e em trânsito, além de auditoria.
- **Integração com Outros Serviços AWS:** É facilmente integrado com outros serviços da AWS, como o *AWS Lambda*, o Amazon S3 e o Amazon Kinesis, permitindo a construção de aplicativos e sistemas completos.
- **Backup e Restauração:** Suporta *backup* e recuperação de dados, ajudando a proteger contra perda de dados acidental.
- **Modelo de Preços Flexível:** O *DynamoDB* oferece opções de preços flexíveis, permitindo que os usuários paguem apenas pelo armazenamento e capacidade provisionada.

O Amazon *DynamoDB* é amplamente utilizado em uma variedade de cenários, desde aplicativos da web e móveis até jogos, análise de dados em tempo real e aplicativos que requerem escalabilidade e desempenho excepcionais. Ele é uma escolha popular para organizações que desejam escalabilidade e desempenho de banco de dados sem a sobrecarga de gerenciar a infraestrutura subjacente.

3.4 K6

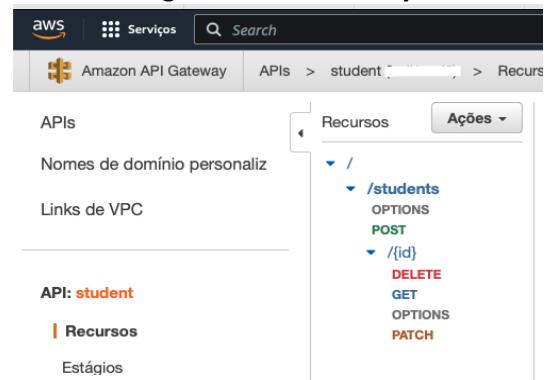
O k6 é uma ferramenta de código aberto de teste de carga e desempenho projetada para ajudar desenvolvedores e equipes de operações a avaliar o desempenho de

aplicativos e sistemas. Ele permite automatizar testes de carga, medir o desempenho e identificar problemas antes que afetem os usuários em produção (K6, 2023). O k6 é amplamente utilizado para verificar o desempenho de aplicativos web, APIs e serviços, garantindo que eles operem de forma eficaz e escalável, independentemente das condições de carga. É uma ferramenta valiosa para manter o desempenho de sistemas críticos e identificar possíveis problemas antes que eles afetem os usuários finais

4. RESULTADOS E DISCUSSÃO

Após a criação de uma conta na AWS e utilizando os recursos e limites gratuitos, através do Amazon *API Gateway*, foi criado um recurso com quatro métodos, sendo eles POST, PATCH, GET e DELETE, para a criação, consulta, deleção e atualização dos dados de cada aluno, conforme apresentado visualmente Figura 6.

Figura 6 - API Gateway



Fonte: Autores

O código fonte para a Lambda foi desenvolvido na linguagem de programação Python, conforme Figura 7.

Vale lembrar que a Lambda é um serviço FaaS, totalmente *serverless* e orientado a eventos. Nesse caso, o gatilho ou evento

que dispara o código da Lambda são as requisições feitas para o API Gateway, conforme exemplificado na Figura 8.

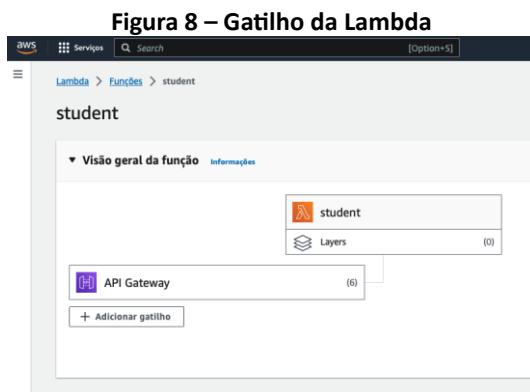
Figura 7 – Código lambda

```

1 from pprint import pprint
2
3 import boto3
4 import json
5 import logging
6 import sys
7
8 # define the DynamoDB table that Lambda will connect to
9 tableName = "student"
10
11 # create the DynamoDB resource
12 dynamo = boto3.resource("dynamodb").table(tableName)
13
14 class Student:
15     def __init__(self, id, name, cpf):
16         self.id = id
17         self.name = name
18         self.cpf = cpf
19
20 def Lambda_handler(event, context):
21     body = ""
22     statusCode = 200
23
24     # Chamada via API Gateway com proxy para Lambda
25     # =====
26     # 'requestContext' in event and 'resource' in event:
27     if 'requestContext' in event and 'resource' in event:
28         rota = str(event['requestContext']['resource']) + str(event['resource'])
29     else:
30         statusCode = 400
31
32     """
33     """

```

Fonte: Autores



Fonte: Autores

A tabela “student” armazena os dados do aluno. Visto que o DynamoDB é um banco não relacional, é apenas informado a chave de partição, nesse caso, o id do tipo String, conforme Figura 9.



Fonte: Autores

A fim de verificar se a implementação obteve sucesso, foi realizado uma

requisição através do Postman, uma ferramenta para testes de API, conforme Figura 10.



Fonte: Autores

O cadastro foi realizado e salvo no banco de dados, conforme Figura 11.



Fonte: Autores

Para verificar a taxa de sucesso das requisições, foi desenvolvido um cenário de teste do método POST através do k6, conforme Figura 12.

Figura 12 – Cenário de teste método POST

```

1 export default function () {
2
3     const url = 'https://execute-api.us-east-2.amazonaws.com/v1/students';
4     const payload = JSON.stringify(createStudent());
5
6     const response = http.post(url, payload);
7
8     let passed = check(response, { 'statusCode was 201': (r) => r.status === 201 });
9
10    if (!passed) {
11        console.log('Request to $(url) failed with status $(response.status) and body: ');
12        console.log(response.body);
13    }
14
15    sleep(1);
16
17 }

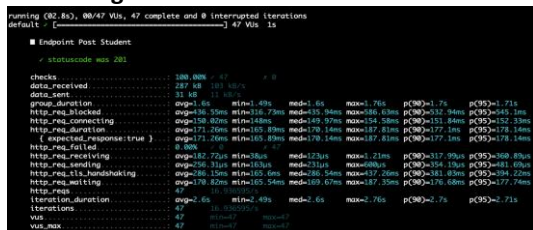
```

Fonte: Autores

Após todo o cenário construído e configurado, os testes foram realizados. Para o cenário com 47 usuários virtuais

simultâneos com duração de 1s, a taxa de sucesso do cadastro de alunos foi de 100%, conforme demonstrado na Figura 13.

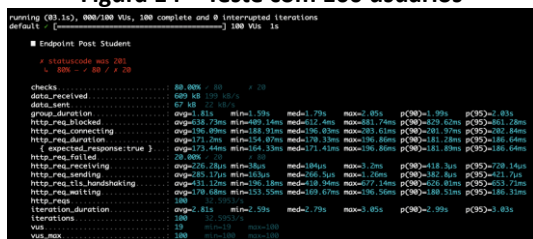
Figura 13 – Teste com 47 usuário



Fonte: Autores

No cenário onde há 100 usuários virtuais simultâneos com duração de 1s, a taxa de sucesso do cadastro de alunos chegou a 80%, conforme demonstrado na Figura 14.

Figura 14 – Teste com 100 usuários



Fonte: Autores

5. CONCLUSÃO

Este artigo objetivava explorar o conceito de CN, abordando suas principais características, benefícios e limitações, pois os serviços em nuvem explodiram nos últimos anos, especialmente em aplicações empresariais baseadas na web. O conceito de pagamento conforme o uso permite uma maneira mais versátil e valiosa de acessar o poder da computação que agora é relativamente fácil de obter. Os aplicativos baseados em nuvem permitem ao usuário comprar facilmente um produto online e ativar e desativar imagens virtuais. Compartilhar e criar imagens virtuais com outros usuários é um dos recursos mais populares dos serviços baseados em nuvem.

Foi desenvolvida e implantada uma aplicação na *Amazon Web Services* no modelo *serverless*, com a realização de testes de carga para avaliar seu desempenho. Pode-se observar que a CN apresenta diversas vantagens, porém também traz consigo desafios significativos. Com base no conhecimento prévio em CN, foram dedicadas oito horas à implementação da aplicação proposta neste artigo, tendo em consideração os desafios enfrentados. Dada a complexidade da tarefa, é notável a facilidade de provisionamento e o curto tempo necessário para o lançamento.

Neste estudo, foi utilizado os recursos fornecidos gratuitamente pela AWS. No entanto, em caso de necessidade de pagamento, apenas os recursos efetivamente utilizados seriam cobrados, levando em conta o número de execuções da função *Lambda* e o armazenamento utilizado.

Apesar das vantagens evidentes, é fundamental contar com pessoal ou uma equipe especializada no assunto para gerenciar de forma eficaz todos os serviços utilizados. Isso é crucial para otimizar a utilização, considerando a natureza do problema a ser solucionado e o valor a ser investido. Além disso, embora alguns serviços sejam apresentados como auto escaláveis, em grande parte dos casos é necessário configurar parâmetros como quantidade de CPU e memória para garantir que a aplicação suporte um maior volume de requisições.

A CN é uma tecnologia poderosa oferecem flexibilidade, escalabilidade e eficiência, porém requer atenção especial à segurança, privacidade e conformidade. A sua adoção exige uma avaliação cuidadosa das necessidades específicas e a capacidade de se adaptar às constantes

mudanças no cenário da tecnologia da informação. À medida que a CN continua a evoluir, é essencial acompanhar as tendências e as melhores práticas para aproveitar ao máximo os seus benefícios.

REFERÊNCIAS

AWS, Amazon Web Services. **O que é computação em nuvem?** Disponível em: <<https://aws.amazon.com/pt/what-is-cloud-computing/>>. Acesso em: 07 de jun. de 2023a.

AWS, Amazon Web Services. **Natura utiliza cloud para lançar nova plataforma de vendas para suas consultoras com suporta da AWS.** Disponível em: <<https://aws.amazon.com/pt/solutions/case-studies/natura-waf/>>. Acesso em: 4 de out. de 2023b.

AWS, Amazon Web Services. **Mercado Livre na AWS.** Disponível em: <<https://aws.amazon.com/pt/solutions/case-studies/innovators/mercado-libre/>>. Acesso em: 4 de out. de 2023c.

AWS, Amazon Web Services. **Tutorial: Uso do Lambda com API Gateway.** Disponível em: <https://docs.aws.amazon.com/pt_br/lambda/latest/dg/services-apigateway-tutorial.html>. Acesso em: 4 de out. de 2023d.

AWS, Amazon Web Services. **Amazon API Gateway.** Disponível em: <<https://aws.amazon.com/pt/api-gateway/>>. Acesso em: 4 de out. de 2023e.

AWS, Amazon Web Services. **Amazon DynamoDB.** Disponível em: <<https://aws.amazon.com/pt/lambda/>>. Acesso em: 4 de out. de 2023g.

AWS, Amazon Web Services. **AWS Lambda.** Disponível em: <<https://aws.amazon.com/pt/lambda/>>. Acesso em: 4 de out. de 2023f.

CORBATÓ, F. J. Time Sharing. In: Encyclopedia of Computer Science. GBR: John Wiley and Sons Ltd., 2003. p. 1778–1782. ISBN: 0470864125.

FIFE, Dennis W. **An Optimization Model for**

Time-Sharing. Proceedings of the April 26-28, 1966, Spring Joint Computer Conference. New York, NY, USA: Association for Computing Machinery. 1966. p. 97–104.

GOOGLE CLOUD. **Limitações da computação em nuvem.** Disponível em: <<https://cloud.google.com/learn/advantages-of-cloud-computing?hl=pt-br#section-4>>. Acesso em: 1 de out. de 2023a.

GOOGLE CLOUD. **O que é computação em nuvem?** Disponível em: <<https://cloud.google.com/learn/what-is-cloud-computing?hl=pt-br>>. Acesso em: 01 de out. de 2023b.

GOOGLE CLOUD. **Benefícios da computação em nuvem.** Disponível em: <<https://cloud.google.com/learn/advantages-of-cloud-computing?hl=pt-br#section-3>>. Acesso em: 01 de out. de 2023c.

IBM, International Business Machines. **O que é FaaS (Função como Serviço)?** Disponível em: <<https://www.ibm.com/br-pt/topics/faas>>. Acesso em: 4 de out. de 2023.

K6. **Welcome to the k6 documentation.** Disponível em: <<https://k6.io/docs/>>. Acesso em: 4 de out. de 2023.

KINSLOW, H. A. **The Time-Sharing Monitor System.** Proceedings of the October 27-29, 1964, Fall Joint Computer Conference, Part I. New York, NY, USA: Association for Computing Machinery. 1964. p. 443–454.

MELL, Peter; GRANCE, Timothy. **The NIST definition of cloud computing.** Gaithersburg: NIST, 2011. Disponível em: <<http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>>. Acesso em: 07 de jun. de 2023.

AZURE, Microsoft Azure. **O que é computação em nuvem?** Disponível em: <<https://azure.microsoft.com/pt-br/resources/cloud-computing-dictionary/what-is-cloud-computing>>. Acesso em: 01 de out. de 2023.

MOTOBAYASHI, Shigeru; MASUDA, Takashi; TAKAHASHI, Nobumasa. **The HITAC5020 Time**

Sharing System. Proceedings of the 1969 24th National Conference. New York, NY, USA: Association for Computing Machinery. 1969. p. 419–429.

MOTTA, Gianmario; SFONDRINI, Nicola; SACCO, Daniele. **Cloud Computing: An architectural and technological overview.** 2012 International Joint Conference on Service Sciences.

NIELSEN, Norman R. **The Simulation of Time Sharing Systems.** Commun. ACM, New York, NY, USA, v. 10, p. 397–412, July 1967. ISSN: 0001-0782. Disponível em: <<https://doi-org.ez338.periodicos.capes.gov.br/10.1145/363427.363436>>. Acesso em: 16 de nov. de 2023.

ORACLE. **What is DBaaS?** Disponível em: <<https://www.oracle.com/database/what-is-a-cloud-database/dbaas/>>. Acesso em: 4 de out. de 2023.

O'SULLIVAN, Thomas C. **Exploiting the Time-Sharing Environment.** Proceedings of the 1967 22nd National Conference. New York, NY, USA: Association for Computing Machinery. 1967. p. 169–175.

RENAUD, Paul E. **Introdução aos sistemas cliente/servidor: um guia prático para profissionais de sistemas.** Rio de Janeiro: John Wiley & Sons, 1994.

RITCHIE, Dennis M.; THOMPSON, Ken. **The UNIX Time-Sharing System.** Proceedings of the Fourth ACM Symposium on Operating System Principles. New York, NY, USA: Association for Computing Machinery. 1973. p. 27.

RUBINOFF, Morris. **Digital Computers for Real-Time Simulation.** J. ACM, New York, NY, USA, v. 2, p. 186–204, July 1955. ISSN: 0004-5411. Disponível em: <<https://doi-org.ez338.periodicos.capes.gov.br/10.1145/320802.320806>>. Acesso em: 16 de nov. de 2023.

SANTOS, Tiago. **Fundamentos da Computação nas Nuven.** Brasil, Editora Senac São Paulo, 2018.

SINGLA, Neeraj; Nisha; Chahat; Harnoor. **A Review Paper on Cloud Computing.** 2022 2nd International Conference on Innovative

Sustainable Computational Technologies (CISCT).

SOLNTSEFF, N. A **Distributed Operating System for an Educational Microcomputer Network.** Proceedings of the 3rd ACM SIGSMALL Symposium and the First SIGPC Symposium on Small Systems. New York, NY, USA: Association for Computing Machinery. 1980. p. 67–71.

TOTVS. **O que é XaaS (Everything as a Service)?** Disponível em: <<https://www.totvs.com/blog/negocios/xaas/#:~:text=XaaS%2C%20ou%20Everything%20as%20a%20Service%2C%20nada%20mais%20é%20do,%2C%20%20como%20um%20serviço>>. Acesso em: 3 de out. de 2023.

VAN TILBORG, André M.; WITTIE, Larry D. **Distributed Task Force Scheduling in Multi-Microcomputer Networks.** Proceedings of the May 4-7, 1981, National Computer Conference. New York, NY, USA: Association for Computing Machinery. 1981. p. 283–289.

WEISS, Aaron. **Computing in the Clouds.** NetWorker, New York, NY, USA, v. 11, p. 16–25, December 2007. ISSN: 1091-3556. Disponível em: <<https://doi-org.ez338.periodicos.capes.gov.br/10.1145/1327512.1327513>>. Acesso em: 16 de nov. de 2023.

ZIKMUND, W. G. et al. **Business Research Methods.** 8. ed. [S.l.]: South-Western Cengage Learning, Boston, 2010. ISBN: 9781439080672LCCN: 2009926309. Disponível em: <<https://books.google.com.br/books?id=pexMPgAACAAJ>>. Acesso em: 16 de nov. de 2023.