



ARQUITETURA DE SOFTWARE EM NUVEM

Manoel Veras – Aula 02

Professores

MANOEL VERAS

Professor Convidado

Graduado em Engenharia Elétrica pela Universidade Federal do Rio Grande do Norte (UFRN), é mestre em Engenharia Elétrica pela Universidade Estadual de Campinas (UNICAMP) e doutor em Administração na Universidade de São Paulo (USP). Atualmente, é professor e consultor vinculado ao Departamento de Administração da Universidade Federal do Rio Grande do Norte (UFRN). Atua também no Programa de Pós-Graduação em Administração como coordenador, com vínculo com a linha de pesquisa Estratégia, Gestão e Tecnologia da Informação. É certificado como PMP pelo PMI desde 2001, ITIL pelo itSMF e COBIT pelo ISACA.

TIAGO COELHO FERRETO

Professor PUCRS

Professor adjunto da Pontifícia Universidade Católica do Rio Grande do Sul, possui doutorado em Ciência da Computação pela PUCRS (2010), com doutorado-sanduíche na Technische Universität Berlin, na Alemanha (2007-2008). Tem experiência na área de Ciência da Computação, com ênfase em Redes de Computadores, atuando principalmente nos seguintes temas: computação em nuvem, grades computacionais, virtualização, processamento de alto desempenho e gerência de infraestrutura de TI.

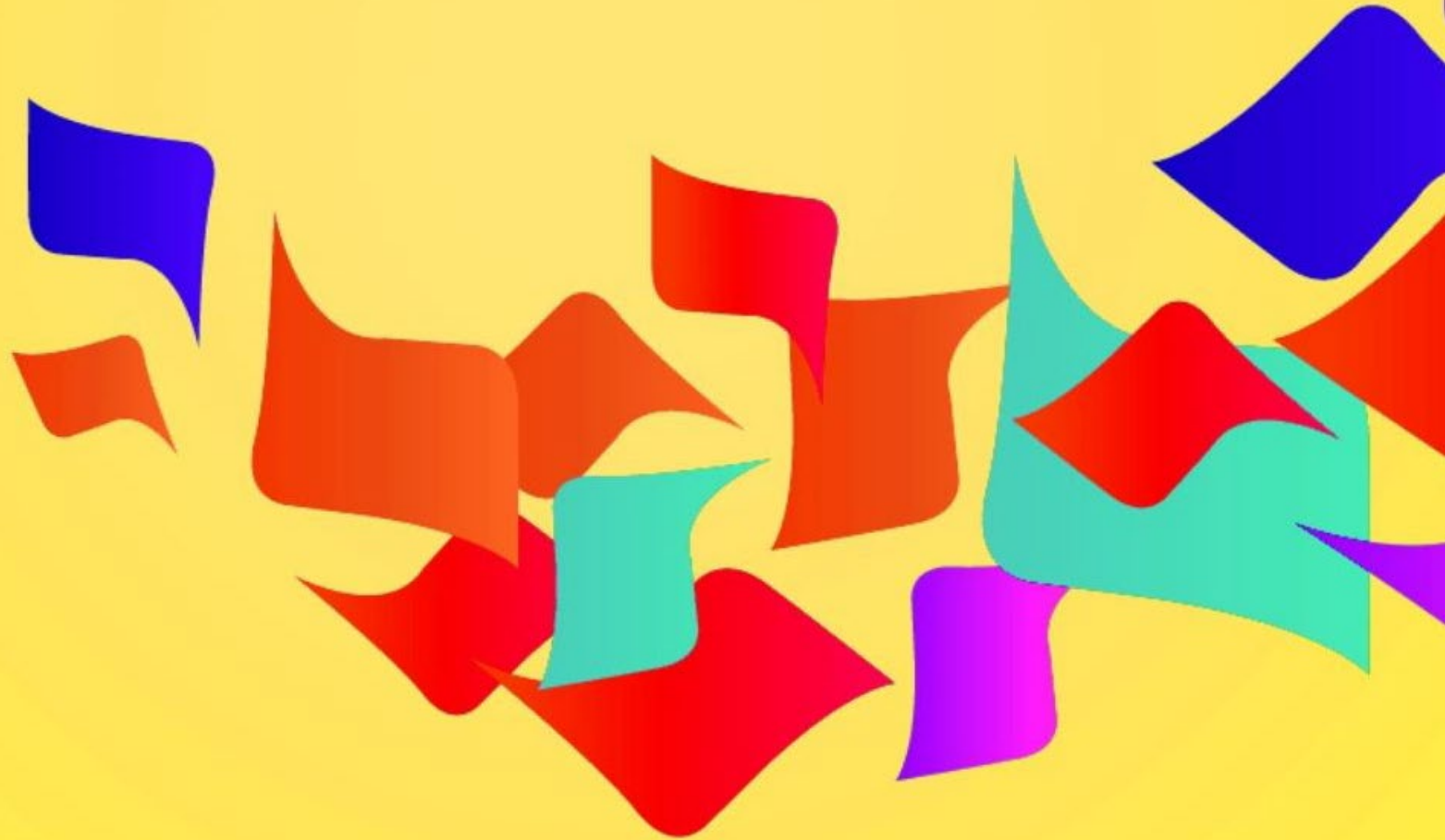
Ementa da disciplina

Características do paradigma de computação em nuvem (self-service, elasticidade, pay-as-you-go). Estudo dos modelos de computação em nuvem: SaaS, PaaS e IaaS. Análise dos principais provedores de nuvem pública (AWS, Google e Azure) e seus serviços. Estudo sobre arquiteturas de software centradas na nuvem (Cloud-Native Architecture). Estudo de casos de aplicações que utilizam computação em nuvem. Desafios e tendências em computação em nuvem.

Desenvolvimento Full Stack

Arquitetura de Software em Nuvem

Prof. Manoel Veras



Aula 2 - Aspectos essenciais da Arquitetura de Nuvem (Princípios)

- Otimização de Custos
- Excelência Operacional
- Eficiência de Performance
- Confiabilidade
- Segurança



I - Otimização de Custos

Uma carga de trabalho com custo otimizado utiliza integralmente todos os recursos, alcança um resultado ao menor ponto de preço possível e atende a seus requisitos funcionais.

A otimização de custos e o gerenciamento financeiro na nuvem são um esforço contínuo. Você deve trabalhar regularmente com suas equipes de finanças e tecnologia, analisar sua abordagem arquitetônica e atualizar sua seleção de componentes

Princípios

- Implementar o gerenciamento financeiro;
- Adotar um modelo de consumo;
- Medir a eficiência geral;
- Parar de gastar dinheiro com tarefas pesadas genéricas;
- Analisar e atribuir gastos.

- Implementar o gerenciamento financeiro. Para obter sucesso financeiro e acelerar a realização de valor empresarial na nuvem, você deve investir em gerenciamento financeiro na nuvem. Sua organização deve dedicar o tempo e os recursos necessários para criar aptidão nesse novo domínio de gerenciamento de tecnologia e uso.
- Adotar um modelo de consumo . Pague apenas pelos recursos de computação que você consome e aumente ou diminua o uso dependendo dos requisitos da empresa. Por exemplo, ambientes de desenvolvimento e teste normalmente são usados apenas por oito horas ao dia durante a semana de trabalho. Você pode parar esses recursos quando eles não estiverem em uso para obter uma economia de custos potencial de 75% (40 horas versus 168 horas).
- Medir a eficiência geral: meça o resultado comercial da carga de trabalho e os custos associados à sua entrega. Use esses dados para entender os ganhos obtidos com o aumento da saída, o aumento da funcionalidade e a redução de custos.
- Parar de gastar dinheiro em tarefas pesadas genéricas: a nuvem faz o trabalho pesado das operações de datacenter, como o armazenamento em rack, o empilhamento e a alimentação de servidores. Ele também elimina a sobrecarga operacional do gerenciamento de sistemas operacionais e aplicativos com serviços gerenciados. Isso permite que você mantenha o foco em seus clientes e projetos de negócios e não na infraestrutura de TI. Analisar e atribuir gastos: a nuvem facilita a identificação precisa do custo e uso das cargas de trabalho, o que permite a atribuição transparente de custos de TI para fluxos de receita e proprietários de cargas de trabalho individuais. Isso ajuda a medir o retorno sobre o investimento (ROI) e oferece aos proprietários de cargas de trabalho a oportunidade de otimizar seus recursos e reduzir custos.
- Analisar e atribuir gastos: a nuvem facilita a identificação precisa do custo e uso das cargas de trabalho, o que permite a atribuição transparente de custos de TI para fluxos de receita e proprietários de cargas de trabalho individuais. Isso ajuda a medir o retorno sobre o investimento (ROI) e oferece aos proprietários de cargas de trabalho a oportunidade de otimizar seus recursos e reduzir custos.

II - Excelência Operacional

O pilar de excelência operacional inclui como sua organização apoia seus objetivos empresariais e sua capacidade de executar cargas de trabalho com eficácia, de obter insights sobre operações e de aprimorar continuamente processos e procedimentos de apoio para oferecer valor empresarial.

Todo evento e toda falha operacional devem ser tratados como uma oportunidade para melhorar as operações de sua arquitetura. Ao compreender as necessidades de suas cargas de trabalho, predefinir runbooks para atividades rotineiras e manuais para orientar a resolução de problemas, usar as operações como recursos de código na nuvem e manter a consciência da situação, suas operações estarão prontas e capazes de responder com mais eficiência quando ocorrerem incidentes.

Princípios

- Executar operações como código
- Fazer alterações frequentes, pequenas e reversíveis
- Refinar procedimentos de operações com frequência
- Antecipar falhas
- Aprender com todas as falhas operacionais

- Executar operações como código: na nuvem, você pode aplicar a todo o seu ambiente a mesma disciplina de engenharia usada para o código do aplicativo. É possível definir toda a sua carga de trabalho (aplicativos, infraestrutura etc.) como código e atualizá-la com código. Você fazer um script dos procedimentos de operações e automatizar sua execução por meio de acionamento deles em resposta a eventos. Ao executar operações como código, você limita o erro humano e permite respostas consistentes aos eventos.
- Fazer alterações frequentes, pequenas e reversíveis: projete cargas de trabalho para permitir que os componentes sejam atualizados regularmente para aumentar o fluxo de alterações benéficas em sua carga de trabalho. Faça alterações em pequenos incrementos que possam ser revertidos se não auxiliarem na identificação e resolução de problemas apresentados em seu ambiente (sem afetar os clientes quando possível).
- Refinar procedimentos de operações com frequência: à medida que você usar procedimentos de operações, procure oportunidades para melhorá-los. Ao evoluir sua carga de trabalho, aprimore procedimentos adequadamente. Organize dias de jogo periódicos (simulações) para analisar e validar se todos os procedimentos são eficazes e se as equipes estão familiarizadas com eles.
- Antecipar falhas: execute exercícios pre mortem para identificar possíveis origens de falhas, para que elas possam ser removidas ou mitigadas. Testar cenários de falha e validar sua compreensão do impacto deles. Teste seus procedimentos de resposta para garantir que sejam eficazes e que as equipes estejam familiarizadas com a execução deles. Organize dias de jogo periódicos para testar cargas de trabalho e respostas da equipe a eventos simulados.
- Aprender com todas as falhas operacionais: promova melhorias por meio de lições aprendidas com todos os eventos e falhas operacionais. Compartilhe o que foi aprendido com as equipes e a organização inteira.

III - Eficiência de Performance

- O pilar de eficiência de performance tem como foco o uso eficiente de recursos de computação para atender a requisitos e manter a eficiência conforme as demandas mudam e as tecnologias evoluem.
- Atingir e manter a eficiência de performance requer uma abordagem conduzida por dados. Você deve avaliar ativamente os padrões de acesso e as concessões que viabilizarão a otimização para uma maior performance. O uso de um processo de análise baseado em benchmarks e testes de carga permite que você selecione os tipos de recursos e as configurações adequados. Tratar sua infraestrutura como código permite que você promova avanços em sua arquitetura de modo rápido e seguro, enquanto usa dados para tomar decisões baseadas em fatos sobre sua arquitetura. O estabelecimento de uma combinação de monitoramentos ativo e passivo garante que a performance de sua arquitetura não apresente degradação ao longo do tempo.

Princípios

- Democratize tecnologias avançadas.
- Tenha alcance global em poucos minutos.
- Use arquiteturas sem servidor.
- Experimente com mais frequência.
- Leve as particularidades em consideração.

- Democratize tecnologias avançadas: facilite a implementação de tecnologia avançada para a sua equipe delegando tarefas complexas ao seu fornecedor de nuvem. Em vez de solicitar que sua equipe de TI aprenda sobre como hospedar e executar uma nova tecnologia, avalie a possibilidade de consumir a tecnologia como um serviço. Por exemplo, bancos de dados NoSQL, transcodificação de mídia e machine learning são tecnologias que exigem altos níveis de especialização. Na nuvem, essas tecnologias se tornam serviços que sua equipe pode consumir, permitindo que a equipe se concentre no desenvolvimento de produtos, em vez de provisionamento e gerenciamento de recursos.
- Tenha alcance global em poucos minutos: a implantação de sua carga de trabalho em várias regiões da AWS em todo o mundo permite oferecer menor latência e uma melhor experiência para seus clientes a um custo mínimo.
- Use arquiteturas sem servidor: as arquiteturas sem servidor eliminam a necessidade de executar e manter servidores físicos para realizar atividades tradicionais de computação. Os serviços de armazenamento sem servidor, por exemplo, podem atuar como sites estáticos (eliminando a necessidade de servidores da web) e os serviços de eventos podem hospedar o código. Isso elimina o fardo operacional do gerenciamento de servidores físicos e pode reduzir os custos transacionais, pois os serviços gerenciados operam em escala de nuvem.
- Experimente com mais frequência: com recursos virtuais e automatizáveis, você pode executar rapidamente testes comparativos usando diferentes tipos de instâncias, armazenamento ou configurações.
- Leve as particularidades em consideração: use a abordagem tecnológica que esteja mais bem alinhada aos seus objetivos. Por exemplo, avalie padrões de acesso a dados ao selecionar abordagens de banco de dados ou armazenamento.

IV - Confiabilidade

- O pilar Confiabilidade abrange a capacidade de uma carga de trabalho de executar a função pretendida correta e consistentemente quando esperado. Isso inclui a capacidade de operar e testar a carga de trabalho durante o ciclo de vida total dela.
- A confiabilidade de uma carga de trabalho na nuvem depende de vários fatores, o principal dos quais é a Resiliência: Resiliência é a capacidade de uma carga de trabalho se recuperar de interrupções de infraestrutura ou serviço, adquirir dinamicamente recursos de computação para atender à demanda e mitigar interrupções, como configurações incorretas ou problemas transitórios de rede.

Princípios do Projeto

- Recuperar de falhas automaticamente.
- Testar os procedimentos de recuperação.
- Escalar horizontalmente para aumentar a disponibilidade agregada da carga de trabalho.
- Parar de adivinhar a capacidade.
- Gerenciar alterações na automação.

- Recuperar de falhas automaticamente: ao monitorar os principais indicadores de performance (KPIs) de uma carga de trabalho, você pode acionar a automação quando um limite é violado. Esses KPIs devem ser uma medida do valor empresarial, não dos aspectos técnicos da operação do serviço. Isso permite a notificação automática e o rastreamento de falhas, além de processos de recuperação automatizados que solucionam ou reparam a falha. Com uma automação mais sofisticada, é possível antecipar e corrigir falhas antes que elas ocorram.
- Testar os procedimentos de recuperação: em um ambiente no local, muitas vezes os testes são realizados para provar que a carga de trabalho funciona em um cenário específico. O teste normalmente não é usado para validar estratégias de recuperação. Na nuvem, você pode testar como a carga de trabalho falha e validar os procedimentos de recuperação. É possível usar a automação para simular falhas diferentes ou para recriar cenários que levaram a falhas antes. Essa abordagem expõe caminhos de falha que você pode testar e corrigir antes que um cenário de falha real ocorra, reduzindo assim o risco.
- Escalar horizontalmente para aumentar a disponibilidade agregada da carga de trabalho: substitua um recurso grande por vários recursos pequenos para reduzir o impacto de uma única falha na carga de trabalho geral. Distribua solicitações em vários recursos menores para garantir que elas não compartilhem uma falha em comum.
- Parar de adivinhar a capacidade: uma causa comum de falha nas cargas de trabalho no local é a saturação de recursos, quando as demandas impostas a uma carga de trabalho excedem a capacidade dela (esse geralmente é o objetivo dos ataques de negação de serviço). Na nuvem, você pode monitorar a carga de trabalho e a utilização do sistema e automatizar a adição ou a remoção de recursos para manter o nível ideal para satisfazer a demanda sem provisionamento em excesso ou subprovisionamento. Ainda há limites, mas algumas cotas podem ser controladas e outras podem ser gerenciadas (consulte Gerenciar cotas e restrições de serviço).
- Gerenciar alterações na automação: as alterações à sua infraestrutura devem ser feitas usando a automação. As alterações que precisam ser gerenciadas incluem alterações na automação, que podem ser rastreadas e analisadas.

Disponibilidade

- Disponibilidade É uma métrica comumente usada para medir quantitativamente a confiabilidade.
 - Disponibilidade é a porcentagem de tempo em que uma carga de trabalho está disponível para uso.
 - Esse percentual é calculado em períodos como um mês, um ano ou os últimos três anos. Aplicando a interpretação mais estrita possível, a disponibilidade diminui sempre que o aplicativo não está operando normalmente, incluindo interrupções programadas e não programadas.
 - Definimos disponibilidade usando os seguintes critérios:
 - $Availability = Available\ for\ Use\ Time / Total\ Time$
 - Uma porcentagem do tempo de atividade (como 99,9%) em um período (geralmente, um ano)
 - A abreviação comum refere-se apenas ao “número de noves”. Por exemplo, “cinco noves” significa 99,999% disponível
 - Alguns clientes preferem excluir o tempo de inatividade do serviço programado (por exemplo, manutenção planejada) do tempo total na fórmula no primeiro marcador. Porém, essa costuma ser uma opção falsa, pois os usuários podem querer usar seu serviço durante esses períodos.

Disponibilidade	Indisponibilidade máxima (por ano)	Categorias de aplicativos
<u>99%</u>	3 dias e 15 horas	Trabalhos de processamento em lote, extração de dados, transferência e carregamento
<u>99,9%</u>	8 horas e 45 minutos	Ferramentas internas como gerenciamento de conhecimento e acompanhamento de projeto
<u>99,95%</u>	4 horas e 22 minutos	Comércio online, ponto de vendas
<u>99,99%</u>	52 minutos	Entrega de vídeo, cargas de trabalho de difusão
<u>99,999%</u>	5 minutos	Transações em caixas eletrônicos, cargas de trabalho de telecomunicações

Cálculo de disponibilidade com dependências rígidas

- Muitos sistemas têm dependências rígidas de outros sistemas, em que uma interrupção de um sistema dependente leva diretamente a uma interrupção do sistema que o invocou. Isso é o oposto de uma dependência flexível, em que uma falha do sistema dependente é compensada no aplicativo. Quando ocorrem essas dependências rígidas, a disponibilidade do sistema invocador é o produto das disponibilidades dos sistemas dependentes. Por exemplo, se você tiver um sistema projetado para uma disponibilidade de 99,99% que tenha uma dependência rígida de outros dois sistemas independentes, cada um projetado para uma disponibilidade de 99,99%, a carga de trabalho teoricamente poderá atingir uma disponibilidade de 99,97%.

$$Avail_{workload} = Avail_{invok} \times Avail_{dep1} \times Avail_{dep2}$$

$$99.99\% \times 99.99\% \times 99.99\% = 99.97\%$$

Cálculo de disponibilidade com componentes redundantes


- Cálculo de disponibilidade com componentes redundantes. Quando um sistema envolve o uso de componentes independentes redundantes (por exemplo, zonas de disponibilidade redundantes), a disponibilidade teórica é calculada como 100% menos o produto das taxas de falha de componente. Por exemplo, se um sistema usar dois componentes independentes, cada um com uma disponibilidade de 99,9%, a disponibilidade do sistema resultante será >99,9999%:
- $$Availworkload = AvailMAX - ((100\% - Availdependency) \times (100\% - Availdependency))$$
$$100\% - (0.1\% \times 0.1\%) = 99.9999\%$$

V - Segurança

- O pilar Segurança descreve como aproveitar as tecnologias de nuvem para proteger dados, sistemas e ativos de uma maneira que possa melhorar sua condição de segurança.
- A segurança é um esforço contínuo. Quando ocorrem incidentes, eles devem ser tratados como oportunidades de melhorar a segurança da arquitetura. Ter controles fortes de identidade, automatizar respostas a eventos de segurança, proteger a infraestrutura em vários níveis e usar criptografia para gerenciar dados bem classificados proporcionam a defesa profunda que todas as empresas devem implementar

Princípios

- Implementar uma base sólida de identidade.
- Habilitar a capacidade de monitoramento.
- Aplicar segurança em todas as camadas.
- Automatizar as melhores práticas de segurança.
- Proteger dados em trânsito e em repouso.
- Manter as pessoas afastadas dos dados.
- Preparar para eventos de segurança.

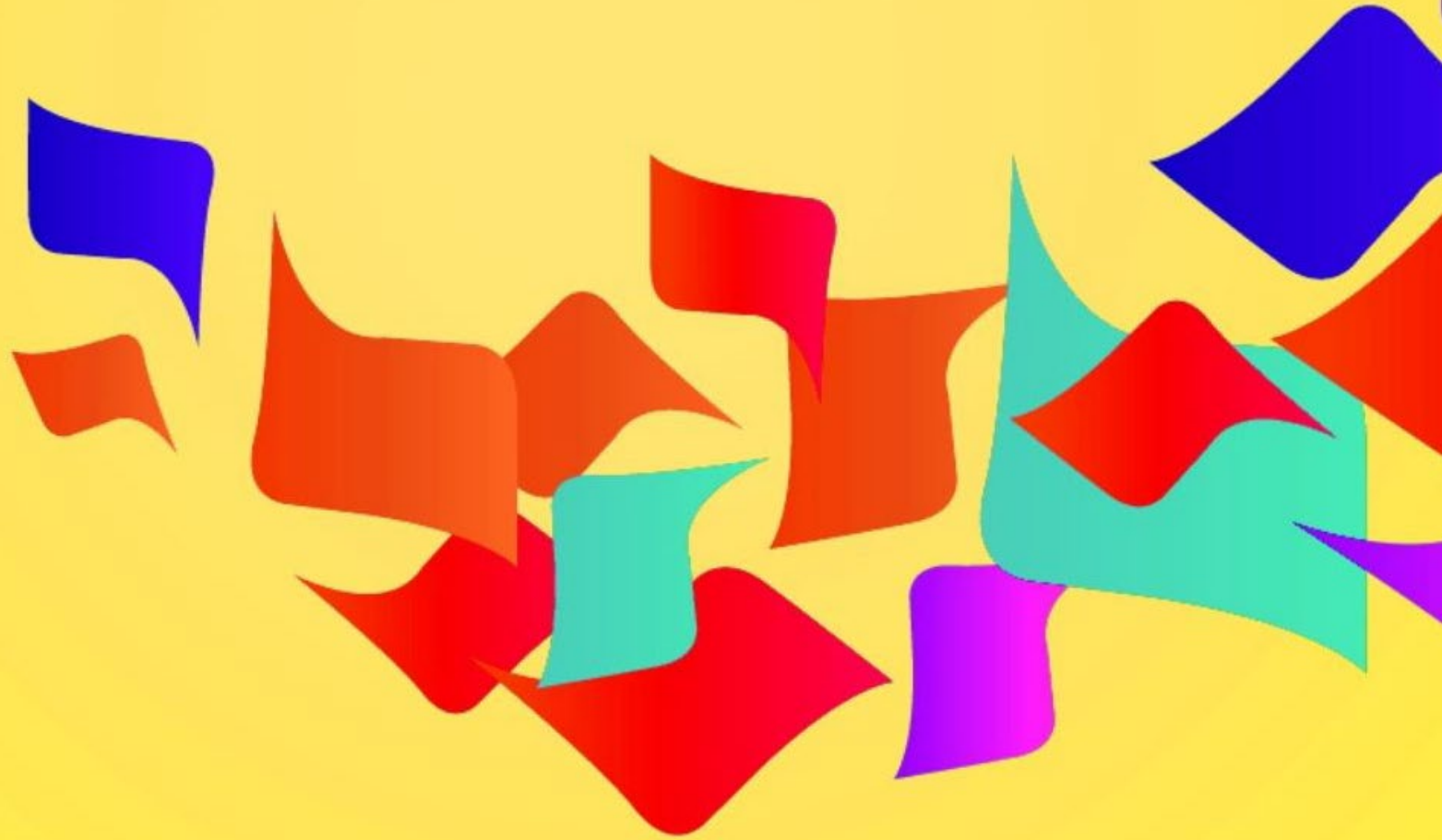
- Implementar uma base sólida de identidade: implemente o princípio do privilégio mínimo e separe as tarefas com autorização apropriada para cada interação com os recursos da nuvem. Centralize o gerenciamento de identidades e procure eliminar a dependência de credenciais estáticas de longo prazo.  Habilitar a capacidade de monitoramento: monitore, alerte e examine ações e alterações em seu ambiente em tempo real. Integre a coleta de logs e métricas aos sistemas para investigar e executar ações automaticamente.
- Aplicar segurança em todas as camadas: aplique uma abordagem de defesa detalhada com vários controles de segurança. Aplique a todas as camadas (por exemplo, borda da rede, VPC, balanceamento de carga, cada instância e serviço de computação, sistema operacional, aplicação e código).
- Automatizar as melhores práticas de segurança: os mecanismos de segurança automatizados baseados em software aprimoram sua capacidade de dimensionar com segurança e de forma mais rápida e econômica. Crie arquiteturas seguras, incluindo a implementação de controles definidos e gerenciados como código em modelos controlados por versão.
- Proteger dados em trânsito e em repouso: classifique seus dados em níveis de confidencialidade e use mecanismos, como criptografia, tokenização e controle de acesso, quando apropriado.
- Manter as pessoas afastadas dos dados: crie mecanismos e ferramentas para reduzir ou eliminar a necessidade de acesso direto ou processamento manual de dados. Isso reduz o risco de erros de processamento ou modificação e erro humano ao manipular dados confidenciais.
- Preparar para eventos de segurança: prepare-se para um incidente com políticas e processos de gerenciamento e investigação de incidentes alinhados aos requisitos organizacionais. Execute simulações de resposta a incidentes e use ferramentas com automação para aumentar sua velocidade de identificação, investigação e recuperação.

Desenvolvimento Full
Stack

Arquitetura de
Software em Nuvem

Considerações
importantes !

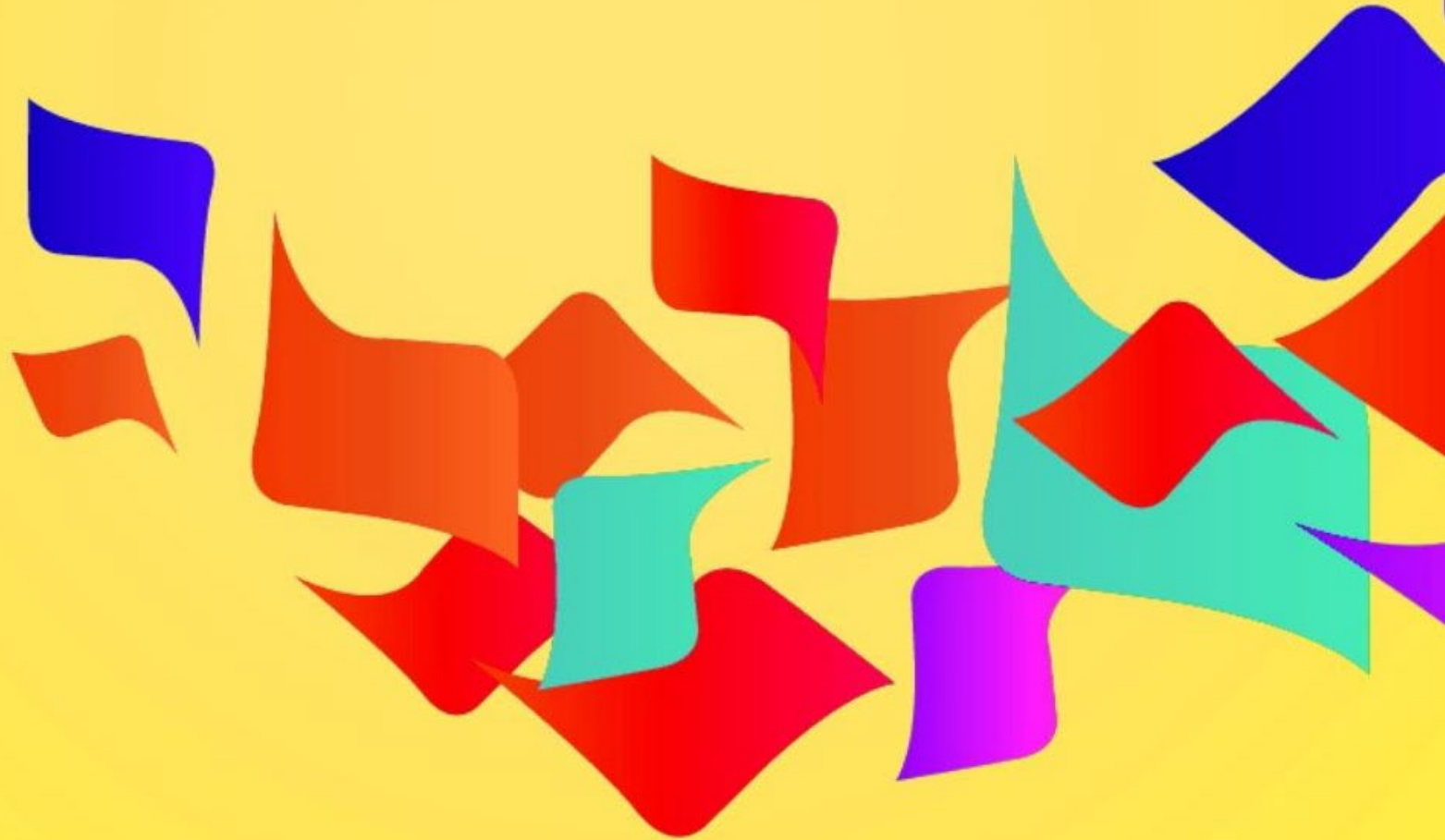
Prof. Manoel Veras



Desenvolvimento Full Stack

Arquitetura de Software em Nuvem

Prof. Manoel Veras



Aula 2 – Comparação de Serviços – AWS – AZURE - GCP

- Cada provedor possui recursos e vantagens que atendem às necessidades específicas de cada cliente de uma forma diferente.
- Da perspectiva do cliente, com a evolução desses serviços, também se abre um caminho que começa a ser adotado por algumas empresas: a nuvem híbrida.

Amazon AWS (AWS)

- Foi o primeiro provedor de nuvem (2006)
- A AWS tem seu foco voltado a ser um provedor amplo de serviços de TI, abrangendo nativos da nuvem e borda até ERP e cargas de trabalho essenciais.
- A AWS possui operações geograficamente diversas e atende clientes de todas as demandas, desde startups em seu estágio inicial até grandes empresas consolidadas no mercado.




Prós e Contras da AWS

- A maior força da Amazon é o domínio do mercado de nuvem pública.
- Em seu Quadrante Mágico de Infraestrutura de Nuvem como Serviço, em todo o mundo, o Gartner observou: “A AWS é líder em participação de mercado em IaaS na nuvem há mais de 10 anos”.
- Parte da razão de sua popularidade é, sem dúvida, o enorme escopo de sua oferta.
- A AWS possui uma enorme e crescente variedade de serviços disponíveis, bem como a rede mais abrangente de datacenters em todo o mundo.
- O relatório do Gartner resumiu, dizendo: “A AWS é o provedor mais maduro e pronto para empresas, com os recursos mais profundos para administrar um grande número de usuários e recursos”.
- Uma possível fraqueza da Amazon está relacionada ao custo. Muitas empresas acham difícil entender a estrutura de custos da empresa e gerenciar esses custos efetivamente ao executar um alto volume de cargas de trabalho para um serviço.

Quando escolher AWS

- A AWS é uma ótima opção para cargas de trabalho analíticas e web, até migrações de data center em grande escala, a AWS fornece uma série de serviços.
- Quando se trata de computação, a AWS fornece a maior variedade de tipos de VM. Atualmente, a AWS também possui diversas opções de computação e armazenamento disponíveis no mercado.
- Sua ampla variedade de tipos de VM (136 tipos de VM e mais de 26 famílias de VM) permite que os clientes executem desde pequenas cargas de trabalho na web até as maiores cargas de trabalho.
- Para aprendizado de máquina e cargas de trabalho de IA, a AWS também fornece as configurações dos tipos de VM habilitados para GPU.
- Para cargas de trabalho que exigem locação única por motivos de conformidade e regulamentação, a AWS agora também fornece Bare-Metal-as-a-Service.
- O armazenamento em bloco vem com uma variedade de opções, como redimensionamento dinâmico, diferentes tipos de disco (magnético e SSD).
- Ao contrário de outros CSPs, a AWS não restringe IOPS por tamanho de volume. Você pode provisionar IOPS por um custo extra até para discos pequenos.

- 
- Na frente do banco de dados relacional gerenciado, a AWS oferece suporte a bancos de dados gerenciados para MySQL, PostgreSQL, MariaDB, Oracle (SE e EE) e MS SQL (edições Web e Enterprise).
 - Além disso, eles têm seu próprio banco de dados compatível com MySQL e PostgreSQL, que oferece desempenho semelhante ao Oracle por um investimento baixo.
 - A AWS fornece uma variedade de bancos de dados NoSQL. Isso inclui DynamoDB, Neptune e ElastiCache.
 - Para segurança de rede, a AWS lançou serviços gerenciados para proteção contra DDoS (AWS Shield) e Web Application Firewall (WAF), juntamente com o AWS Inspector, o AWS Config e o CloudTrail para gerenciamento e auditoria de inventário e políticas.
 - O GuardDuty fornece detecção de ameaças.
 - A AWS atende a cargas de trabalho do governo dos EUA em regiões separadas do GovCloud nos EUA (CIA e FBI).

Microsoft Azure (AZURE)

- O Azure é uma opção para todos os casos de uso, que incluem a computação estendida em nuvem e de borda.
- Com foco de investimento em melhorias na plataforma, fornecendo um grande leque de serviços, suas operações são geograficamente diversificadas e seus clientes tendem a ser empresas de médio e grande porte.
- A Azure é a concorrente que mais conseguiu se aproximar da líder AWS.

Prós e contras do Microsoft Azure

- A Microsoft chegou atrasada ao mercado de nuvem, mas deu um passo à frente, adotando essencialmente o software local – Windows Server, Office, SQL Server, Sharepoint, Dynamics Active Directory, .Net e outros – e adaptando-os para a nuvem.
- Um grande razão para o sucesso do Azure é a integração com as aplicações/softwarewares da Microsoft.
- Como o Azure está totalmente integrado a esses outros aplicativos, as empresas que usam muitos softwarewares da Microsoft geralmente acham que também faz sentido usar o Azure.

Quando escolher o Azure

- O Azure é uma plataforma de nuvem com uma variedade de recursos, que pode ser uma plataforma preferida para clientes que já estão usando produtos da Microsoft.
- Embora o Azure ofereça suporte a vários serviços baseados em produtos de código aberto, o portfólio da Microsoft na nuvem é o que o diferencia dos clientes.
- O Azure tem mais de 151 tipos de VMs e 26 famílias que oferecem suporte a tudo, desde pequenas cargas de trabalho até as cargas de trabalho HPC, Oracle e SAP.
- O Azure utiliza Windows e vários tipos de Linux (RHEL, CentOS, SUSE, Ubuntu). O Azure tem uma família separada de instâncias para cargas de trabalho de ML/AI.
- Se você precisar executar cargas de trabalho de última geração que exijam até 128 vCPU e memória de 3,5 TB, o Azure consegue.
- Se você possui licenças existentes para Windows OS, MS-SQL e as traz para a nuvem (BYOL) por meio do Microsoft License Mobility Program, o Azure é a opção.
- O Azure também foi o primeiro player de nuvem a reconhecer a tendência da nuvem híbrida. O Azure também forneceu suporte para dispositivos de armazenamento híbridos como o StorSimple, que era único no espaço da nuvem pública.

- No caso de um data center com cargas de trabalho predominantemente da Microsoft e que precisa fazer uma migração em grande escala para a nuvem, aproveitando as ferramentas conhecidas, o Azure fornece ferramentas e serviços, como o Azure Site Recovery.
- Quando se trata de bancos de dados SQL e NoSQL, o Azure tem um conjunto de serviços bastante completo. Ele fornece o MS SQL Server e o SQL Datawarehouse Gerenciados.
- O Azure também fornece bancos de dados gerenciados para MySQL, PostgreSQL e MariaDB.
- Ele fornece uma API compatível com MongoDB, Cassandra, Gremlin (Graph) e Armazenamento de Tabela do Azure.
- Se é necessário executar vários modelos de dados gerenciados, incluindo modelos de dados de documentos, gráficos, valores-chave, tabelas e famílias de colunas em uma única nuvem, o Cosmos DB pode ser a melhor opção.
- O Microsoft Azure Cosmos DB é nomeado líder no relatório Forrester Wave™: NoSQL Big Data, relatório do 1º trimestre de 2019.
- Além do modelo de cobrança por cartão de crédito e outros modos de faturamento, os clientes com contas corporativas existentes podem comprar pré-assinaturas do Azure como parte de suas renovações anuais. Isso é útil para clientes que desejam orçar os gastos anuais da nuvem com antecedência. Evitando a incerteza e as aprovações adicionais de orçamento para o meio do ano.
- A mobilidade de licenças na nuvem para produtos da Microsoft também é relativamente fácil para clientes com vários produtos da Microsoft em execução no local.

Google Cloud Platform (GCP)


- O Google Cloud Platform (GCP) está melhorando seus recursos de borda e tomou dianteira pela terceira maior fatia do mercado.
- Com contínuo investimento para ser um provedor amplo de IaaS e PaaS e expandindo seus recursos, é um nome forte de mercado que vem se fortalecendo dentro do segmento.
- Como as outras opções, tem posições geograficamente diversificadas e os clientes são variados, desde startups até grandes negócios.
- No último balanço fornecido pelo Synergy Research Group, o GCP ocupou 10% do segmento em 2021.

Prós e contras do Google Cloud Platform (GCP)

- O Google Cloud Platform (GCP), apesar de atrasado no jogo e com a menor participação de mercado dos provedores de nuvem pública, está mostrando um crescimento nos últimos anos.
- Possui vários recursos que o colocam à frente de seus concorrentes em determinadas áreas.
- O GCP também está pegando onda, não apenas com os novos clientes que já fazem parte do ecossistema, mas também os primeiros usuários da nuvem que desejam expandir seu cenário para o Google como parte de uma estratégia para várias nuvens.
- O Google começou com serviços de PaaS, mas vem expandindo constantemente seu portfólio de produtos.

Quando escolher o GCP

- Do ponto de vista da computação, o Google tem o menor número de tamanhos de VM (28 tipos de instância em 4 categorias).
- No entanto, ele tem uma característica que torna esses números um pouco irrelevantes. O Google permite que os usuários criem seus próprios tamanhos personalizados (CPU, memória) para que os clientes possam combinar o tamanho das cargas de trabalho na nuvem com o tamanho no local.
- O faturamento também é feito com base na CPU e memória totais usadas, em vez de VMs individuais. Isso reduz o desperdício de capacidade não utilizada.
- Outro recurso exclusivo é que o GCP permite que quase todos os tipos de instância conectem GPUs. Isso pode transformar qualquer instância padrão ou personalizada em uma VM pronta para ML.
- O Google também foi líder em cobrança por segundo, o que forçou outros CSPs a seguir o exemplo. Comparado à norma usual do faturamento por hora, o faturamento por segundo reduz muito qualquer desperdício de capacidade. Isso resulta em uma economia de até 40% no geral.

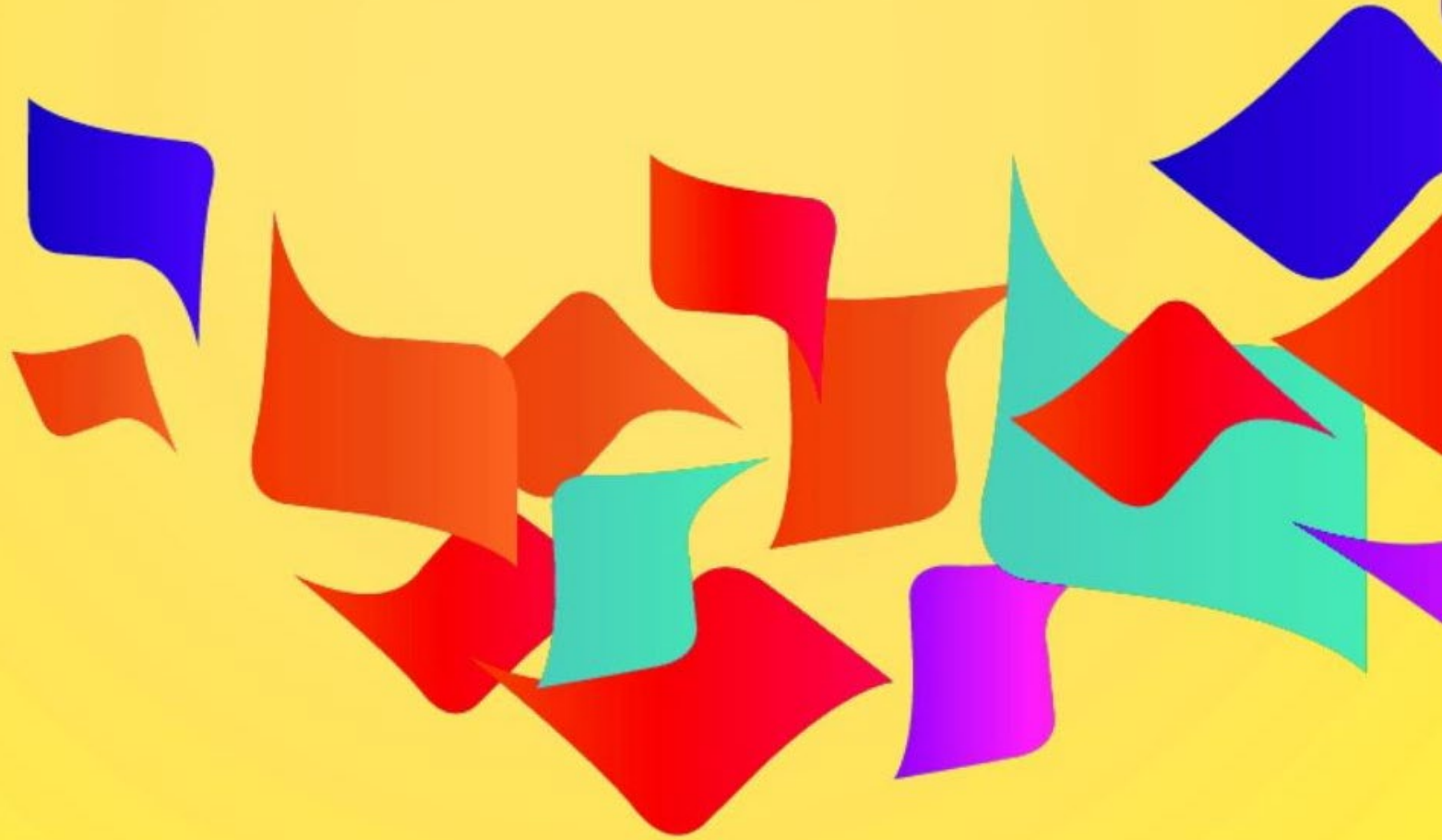
- 
- O Google também vinculou ou comprou ferramentas de migração para a nuvem de terceiros. Essas ferramentas, como Velostrata e CloudPhysics, ajudam os clientes a avaliar, planejar e migrar ao vivo suas VMs para o GCP.
 - Rede é o destaque do GCP. Eles têm uma rede global de baixa latência. Mesmo da perspectiva do cliente, uma rede VPC abrange todas as suas regiões.
 - Outros CSPs limitam as redes VPC a uma região. Isso facilita para os clientes do GCP criar aplicativos que atendem aos clientes globalmente, sem criar complexos mecanismos de design de infraestrutura entre regiões e replicação de dados.
 - Para bancos NoSQL, o GCP tem um produto chamado BigTable. O BigTable é um banco de dados NoSQL gerenciado em escala de petabytes, usado pelo Google em seus próprios produtos, como o Gmail.
 - Do ponto de vista de cobrança, o Google oferece descontos automáticos, como descontos de uso sustentado, que reduzem o preço sob demanda se uma VM executar mais de um determinado número de horas em um mês.

Desenvolvimento Full
Stack

Arquitetura de Software
em Nuvem

Considerações
Importantes !

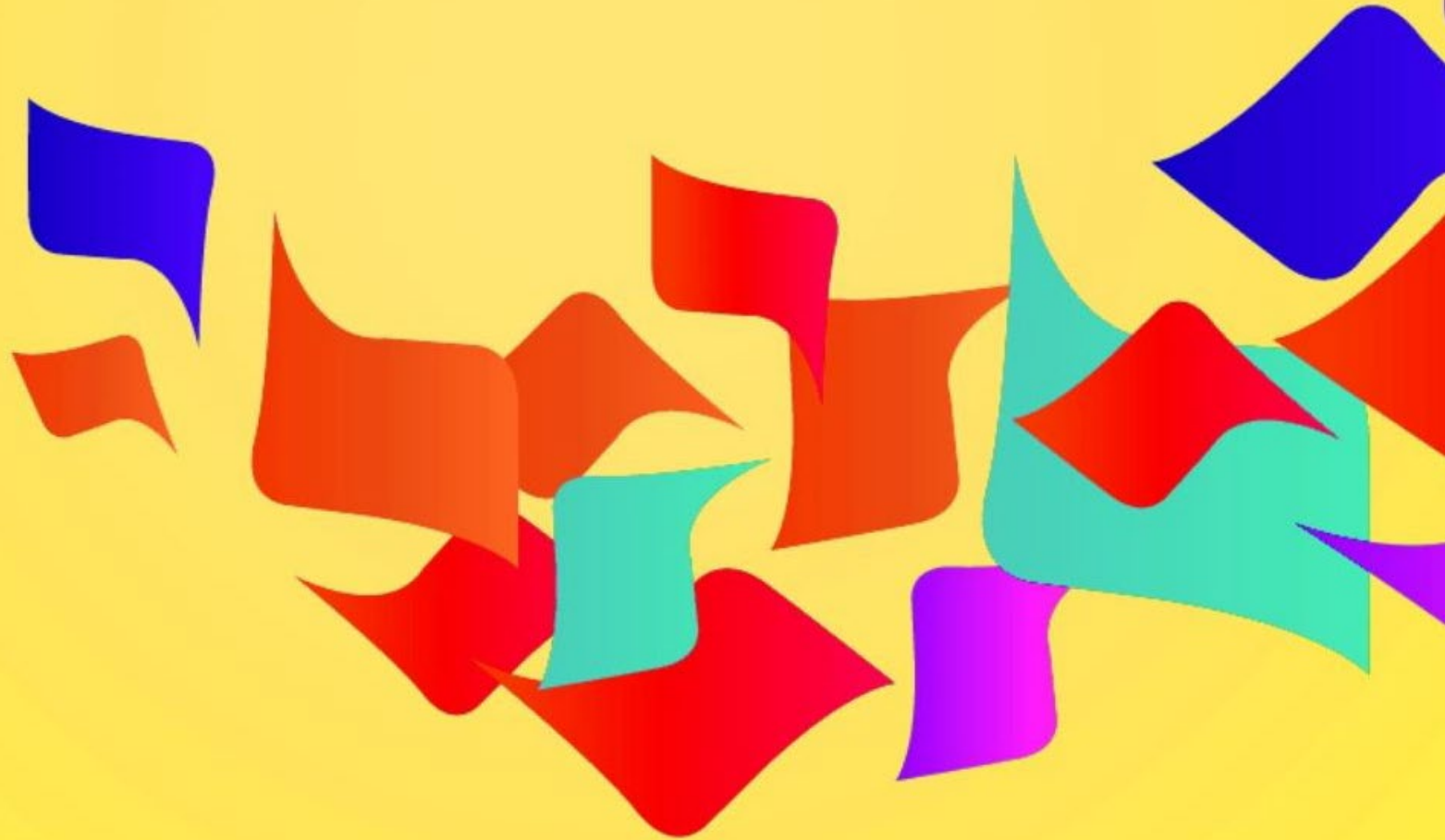
Prof. Manoel Veras



Desenvolvimento Full Stack

Arquitetura de Software em Nuvem

Prof. Manoel Veras

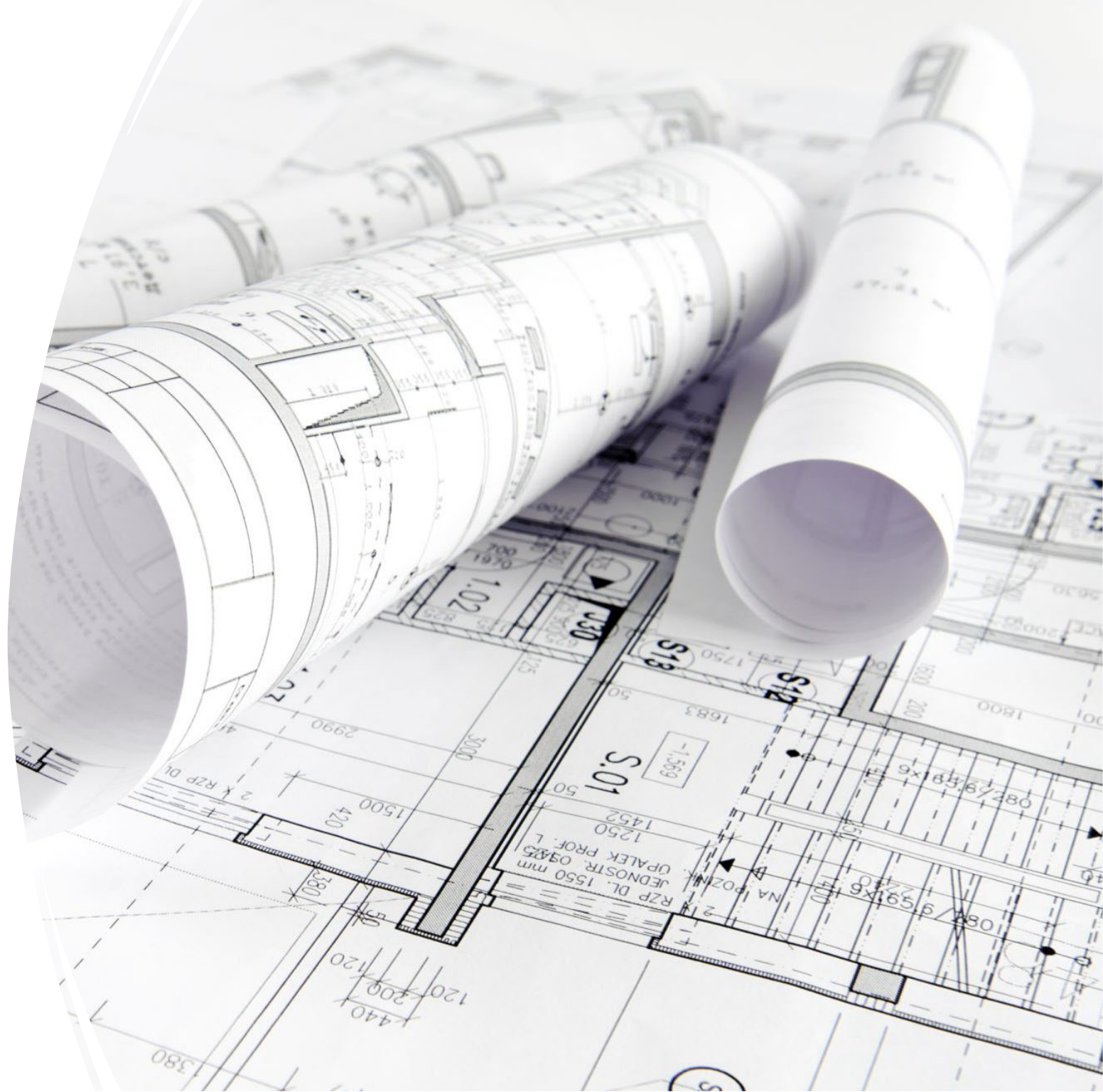


Aula 2 - Caso de Aplicação



Caso de Arquitetura AWS – Empresa de Delivery

- Problem/Statement Definition: Cliente precisava migrar de uma arquitetura monolítica para uma arquitetura escalável e resiliente
- What you proposed: Migrar de uma abordagem de datacenter para a AWS, transformando a aplicação para se beneficiar de serviços gerenciados e com capacidade de escalar horizontalmente
- How AWS services were used as part of the solution: Vários serviços da AWS foram provisionados e configurados para atingir os objetivos do cliente e estar 100% na AWS. Entre eles, EC2, ElastiCache, Aurora, ELB e S3.
- Third Party applications or solutions used: New Relic e Google Analytics foram utilizados para ajudar com métricas importantes da aplicação e de uso de recursos.
- Start and End dates of project: Nov/18 - Dez/18



Resultados

- Outcome(s)/results:

A migração para AWS eliminou as barreiras de infraestrutura que impedia o negócio crescer e gerava problemas. Por conta da arquitetura monolítica, haviam janelas de downtime, entre 15 e 30 minutos, para upgrades da infraestrutura e momentos de indisponibilidade ou instabilidade durante picos de acessos. A aplicação, agora, é capaz de auto escalar baseado no seu uso sem nenhum downtime para upgrades/downgrade de capacidade, isso possibilitou a expansão comercial da empresa.

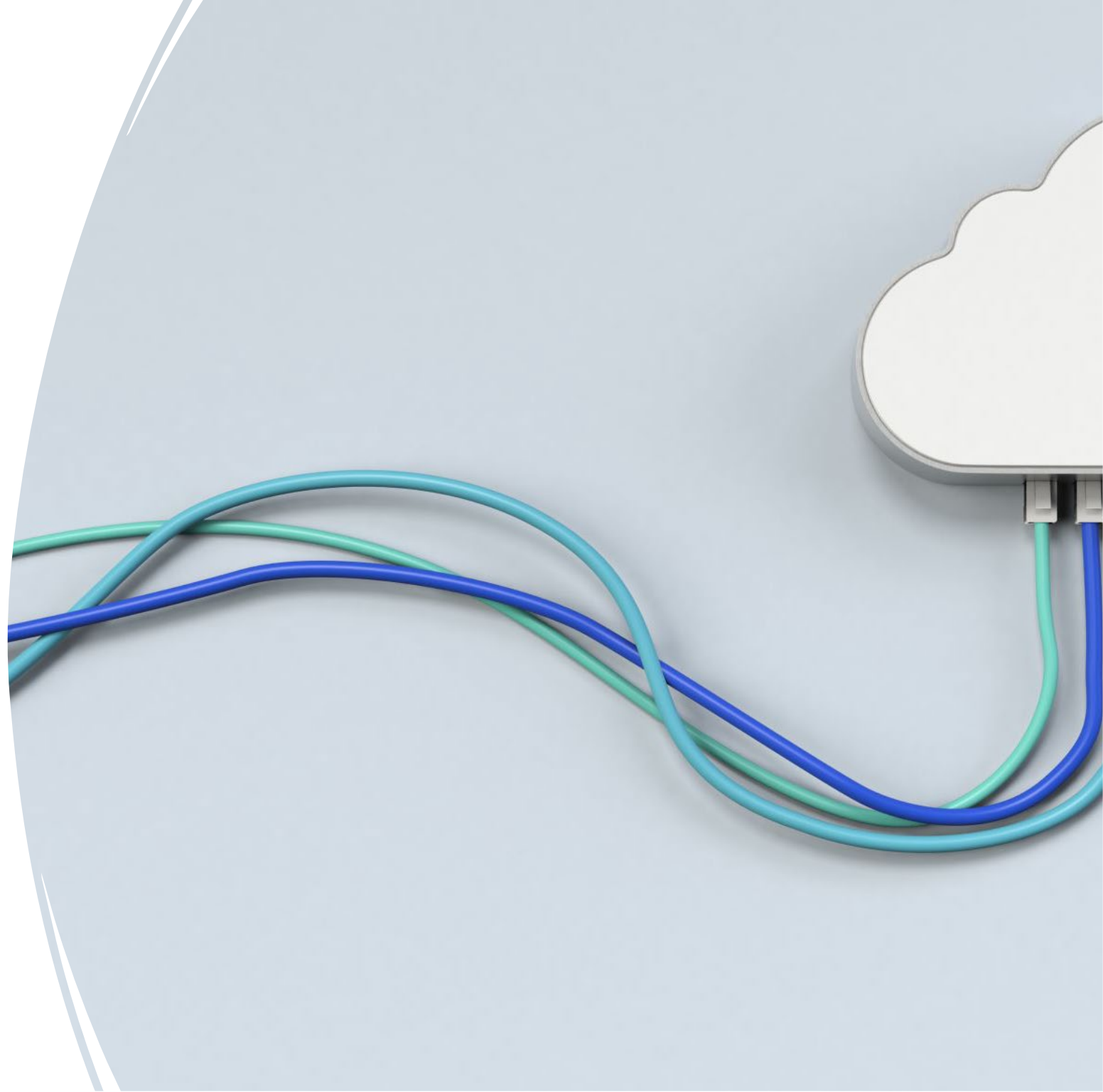


Benefícios

- Aumento da disponibilidade da aplicação em horários de pico;
- Elasticidade da infraestrutura;
- Redundância dos serviços provisionados;
- Política de continuidade (backup e restore);
- Autonomia da equipe interna para realizar mudanças e melhorias na aplicação;
- Camada de serviços integrada;
- Entrega contínua de novas versões da aplicação sem interrupções;
- Custo otimizado para um volume crescente de demanda.

Lições Aprendidas

- O time do cliente era bem jovem, reunido a pouco tempo, com pouca experiência coletiva e individual, isso demandou mentoria e assistência da Veezor (parceiro AWS) para entender os conceitos e novas possibilidades que o modelo de computação em nuvem oferecia. Aprendemos que para ter sucesso em projetos com pouca maturidade, precisávamos oferecer um suporte mais próximo e habilidades didáticas eram necessárias para ensinar os fundamentos que incluíam diversos tópicos relacionados ao código da aplicação do próprio cliente.



Arquitetura pré-existente

- Antes da migração para o Aurora, o cliente utilizava MySQL Community instalado no mesmo host que a aplicação PHP e servidor web Apache. A disponibilidade era bem limitada, baseado numa solução monolítica, sem redundância e sem plano de recuperação. O projeto final na AWS separa responsabilidades e preocupações, delegando boa parte ao serviço de banco de dados gerenciado do Aurora.



Diagrama da Arquitetura

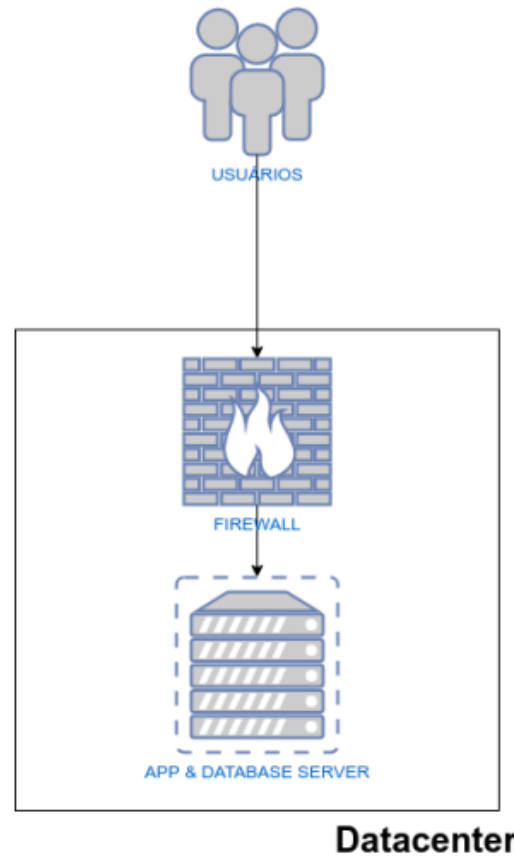


Diagrama Bee Delivery antes da Migração para AWS

Requisitos da Nova Arquitetura

- No novo cenário na AWS, o cliente exigia uma configuração que trouxesse a capacidade de redimensionar a infraestrutura do aplicativo e, portanto, a disponibilidade do banco de dados era parte da preocupação.
- Não havia requisitos de disponibilidade em múltiplas regiões, mas em momentos de pico, era importante que as réplicas de leituras escalassem automaticamente provendo capacidade suficiente para atender todas chamadas as consultas em tempo real que seriam feitas.
- A arquitetura final com um cluster Aurora utilizando uma estratégia de auto-scaling de réplicas de leitura, atendeu perfeitamente os requisitos iniciais de escalabilidade e resiliência ao mesmo tempo e supera todo o cenário anterior que o cliente tinha.

Requisitos

- Deverá ser provisionado um cluster Aurora MySQL com Auto Scaling de Read Replica, executando um procedimento de backup/restore em uma janela de manutenção do antigo datacenter para a AWS na região US-EAST-1

Requisitos

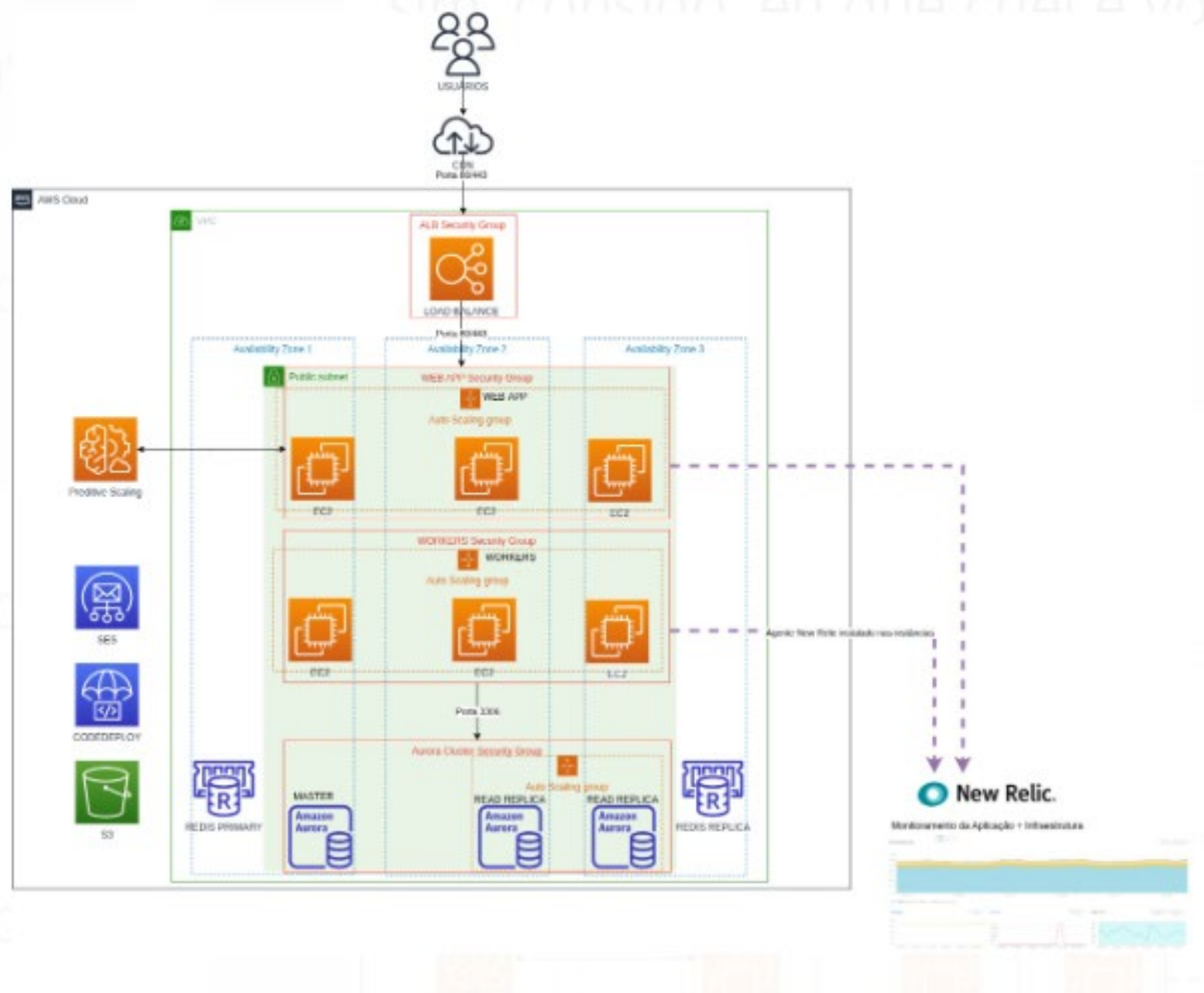
- Gerenciamento da Base de Dados pelo parceiro : O cluster Aurora da Empresa de Delivery já foi projeto para ser elástico e resiliente desde momento do projeto, adotando a estratégia não só de uso de réplica de leitura em Multi-AZ como o mecanismo de auto escalabilidade nativo que altera o cluster para até 15 réplicas de leitura, com pelo menos um primário para escrita e uma réplica para leitura. Nesse contexto, temos pelo menos uma réplica de leitura principal em AZ diferente do banco de escrita e prioridade mais alta que réplicas que são criadas sob demanda automática no auto scaling. O uso do cluster pela aplicação é feito utilizando o endpoint de leitura e o endpoint de escrita a depender da operação a ser realizada no cluster Aurora. Em caso de interrupção de uma das AZs envolvidas no banco de dados de escrita, acionamos o Failover (Action->Failover) fazendo a réplica de leitura assumir o seu papel(RPO Restore to point in time) para o último válido ou logo antes do horário da falha(RPO).

Requisitos

- Segurança: São definidas senhas de pelo menos 16 caracteres contendo minúsculo, maiúsculo e números. As rotações de senha acontecem de forma manual e sob demanda. É utilizada encriptação no armazenamento dos dados através de chave gerenciada pelo KMS.
- Aplicação : A aplicação é desenvolvida totalmente pelo cliente e todas funcionalidade de rotação automática, recuperação, armazenamento e acesso de senha dependiam de modificação da aplicação para serem utilizadas, não foram prioridades na migração para a AWS. O acesso ao banco de dados foi restrito através de security group, permitindo apenas o security group das instância EC2 que hospedam a aplicação e ao endereço IP do escritório da empresa.

sim, consigo eu criar a vol

com



Características Técnicas da Solução:

Use of AutoScaling for Read Replicas

Tamanho inicial do banco de dados: 100MB

Crescimento anual esperado: 100GB

Número de tabelas no banco de dados: 137

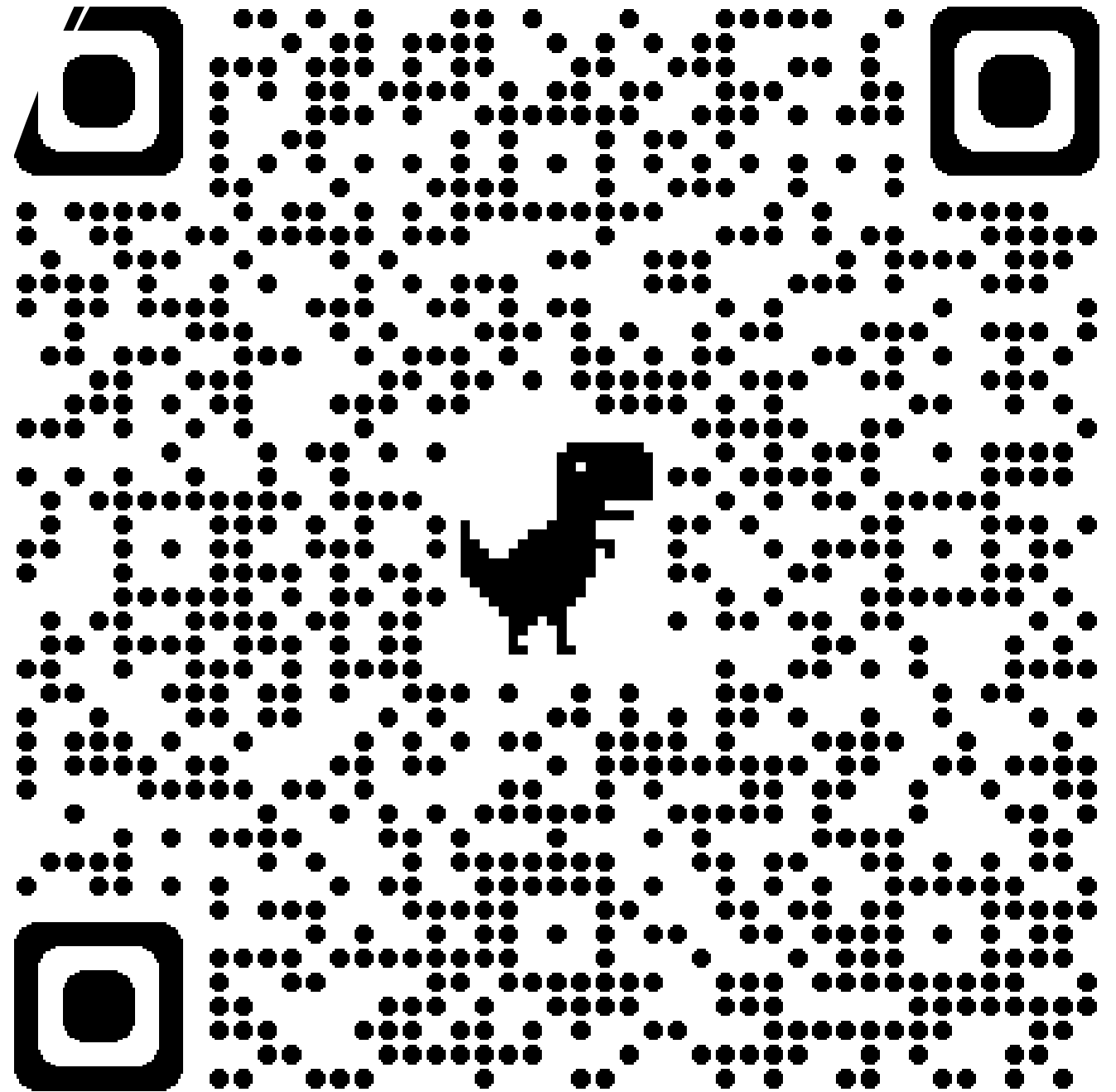
Número de requisições antecipadas durante o pico de uso do banco de dados: 150 mil IOPS/ 7 mil queries por segundo

Porcentagem antecipada de operações de leitura em relação ao banco de dados durante pico: 85%

Porcentagem antecipada de operações de escrita em relação ao banco de dados durante pico: 15%

Arquitetura de Referencia AWS

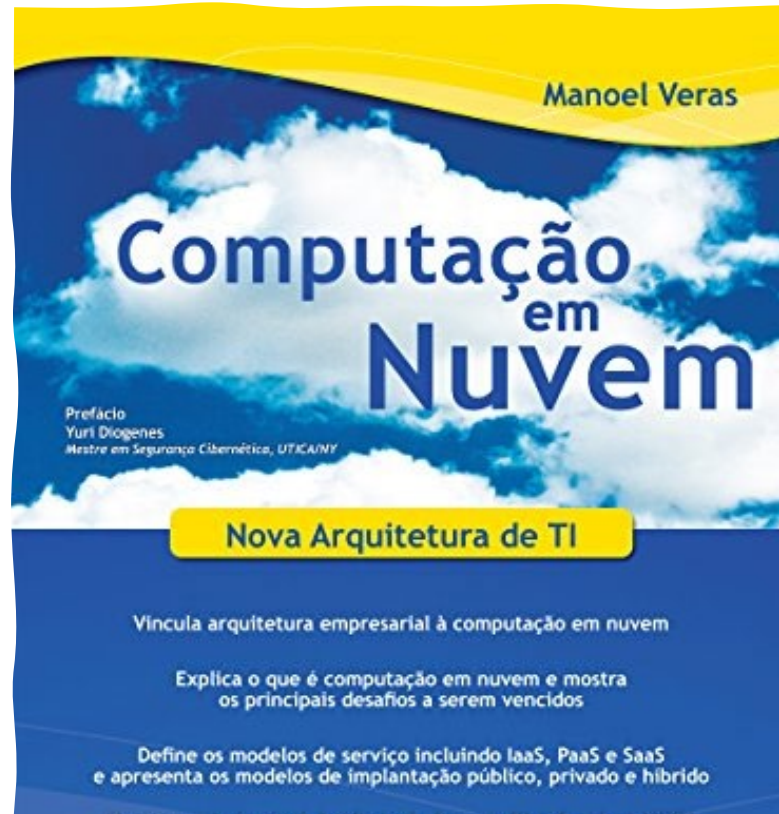
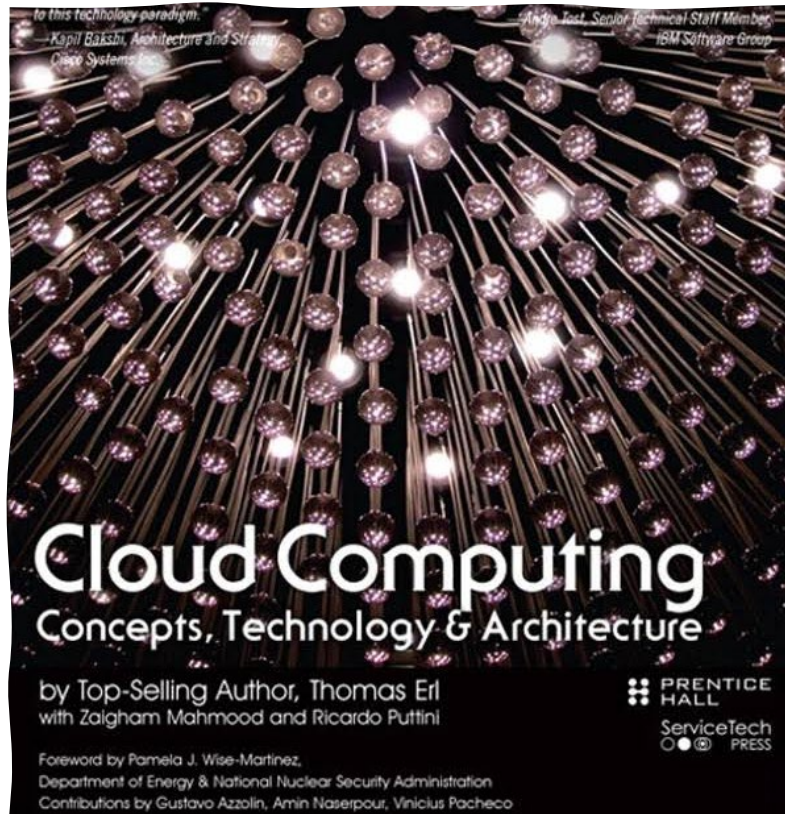
- Open Banking on AWS



AWS Well Architected Tool (AWS WA Tool)

- AWS Well-Architected Tool(AWS WA Tool) é um serviço na nuvem que fornece um processo consistente para medir a arquitetura usando AWS Práticas recomendadas. AWS WA Tool ajuda você durante todo o ciclo de vida do produto:
 - Auxiliando na documentação das decisões tomadas
 - Fornecendo recomendações para melhorar sua carga de trabalho com base nas melhores práticas
 - Orientando você para tornar suas cargas de trabalho mais confiáveis, seguras, eficientes e econômicas
 - <https://docs.aws.amazon.com/wellarchitected/index.html>

Referencias



Referencias

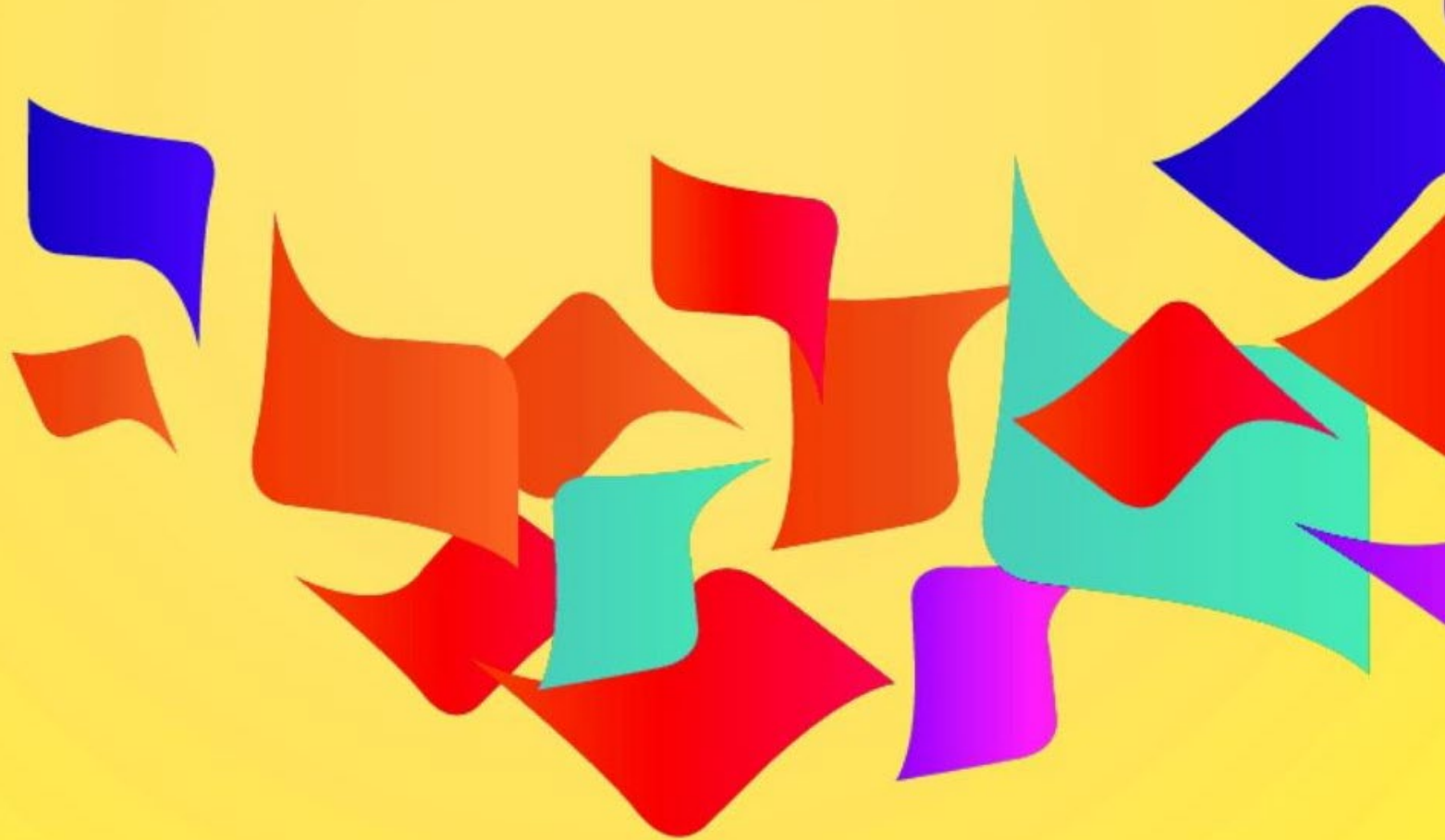
- <https://mkaschke.medium.com/virtual-machine-vm-vs-container-13ab51f4c177>
- <https://blog.saninternet.com/aws-vs-azure-vs-google>
- <https://aws.amazon.com/pt/architecture/>

Desenvolvimento Full
Stack

Arquitetura de
Software em Nuvem

Considerações
Importantes !

Prof. Manoel Veras



PUCRS online  **uol**edtech.