



ESCOLA DA NUVEM

AWS RE/START BRSAO/193

Luis Gustavo Alencar Moura

Paula Eduarda Silva

Ryan Ricardo de Souza

E-Commerce com IA

Trabalho realizado sob orientação dos Instrutores Técnicos Raphael Ohlsen, Anderson e Bruno, e Instrutora comportamental Isabella como requisito de avaliação final no AWS re/Start BRSAO/193.

Brasil

2025

SUMÁRIO

1. INTRODUÇÃO.....	2
2. DESCRIÇÃO DO PROBLEMA.....	3
2.1. STAKEHOLDERS.....	3
2.2. JUSTIFICATIVA E OBJETIVOS.....	4
3. LEVANTAMENTO DE REQUISITOS.....	4
3.1. REQUISITOS FUNCIONAIS (RF).....	4
3.2. REQUISITOS NÃO FUNCIONAIS (RNF).....	6
3.3. PRODUTO MÍNIMO VIÁVEL (MVP).....	6
4. METODOLOGIA E PLANEJAMENTO.....	8
4.1. FRAMEWORK ÁGIL.....	8
4.2. BACKLOG DO PRODUTO E ESTRUTURA DE ETAPAS.....	8
4.3. ESTRATÉGIA DE SPRINTS.....	9
4.4. QUADRO DE GESTÃO DO PROJETO (KANBAN/SCRUM).....	10
5. ARQUITETURA DA SOLUÇÃO.....	11
5.1. VISÃO GERAL DA ARQUITETURA.....	11
5.2. ANÁLISE SEGUNDO O AWS WELL-ARCHITECTED FRAMEWORK.....	12
5.2.1. Pilar de Confiabilidade.....	12
5.2.2. Pilar de Segurança.....	13
5.2.3. Pilar de Eficiência de Performance.....	13
5.2.4. Pilar de Otimização de Custos.....	14
5.2.5. Pilar de Excelência Operacional.....	14
6. MÉTRICAS DE SUCESSO.....	14
6.1. Métricas de Negócio.....	14
6.2. Métricas de Produto e Engajamento.....	15
6.3. Métricas Técnicas e Operacionais.....	15
7. CONSIDERAÇÕES FINAIS.....	15
8. REFERÊNCIAS.....	17

1. INTRODUÇÃO

Este documento apresenta o projeto de desenvolvimento de uma plataforma de e-commerce inovadora, concebida no âmbito do programa de formação AWS re/Start. A solução proposta aborda um desafio central do varejo digital contemporâneo: a fricção na descoberta de produtos e a falta de personalização, que comprometem a experiência do consumidor e as taxas de conversão.

Para resolver essa questão, o projeto detalha a aplicação de Inteligência Artificial (IA) por meio de um chatbot para buscas em linguagem natural e um sistema de recomendações personalizadas. O escopo deste trabalho abrange desde a identificação do problema de negócio e o levantamento de requisitos, passando pelo planejamento com metodologias ágeis, até o desenho de uma arquitetura de solução robusta, segura e escalável na nuvem AWS, seguindo as melhores práticas do AWS Well-Architected Framework.

Palavras-chave: E-commerce. Inteligência Artificial. AWS. Arquitetura de Nuvem. Metodologia Ágil.

2. DESCRIÇÃO DO PROBLEMA

No cenário hipercompetitivo do comércio eletrônico, a experiência do usuário tornou-se o principal diferencial. Contudo, a maioria das plataformas atuais ainda opera com mecanismos de busca genéricos e uma ausência crônica de personalização, falhando em atender às expectativas dos consumidores modernos. Os clientes enfrentam dificuldades significativas para encontrar produtos que correspondam a critérios subjetivos e multifacetados, como a busca por “roupas para o frio, em tons claros e sem estampas”.

Essa fricção no processo de descoberta não é um mero inconveniente; é uma falha estratégica que gera frustração, eleva as taxas de abandono de carrinho e impede a criação de um vínculo de fidelidade entre o cliente e a marca. Consequentemente, os varejistas enfrentam uma perda direta e mensurável de oportunidades de venda e uma incapacidade de maximizar o valor do ciclo de vida do cliente (LifeTime Value).

2.1. STAKEHOLDERS

As partes interessadas e seus respectivos interesses no projeto são:

- **Clientes/Consumidores:** Buscam uma jornada de compra fluida, intuitiva e altamente relevante, que se adapte dinamicamente às suas preferências individuais e intenções de busca.
- **Varejistas/Empresas de E-commerce:** Almejam aumentar métricas de negócio cruciais, como a taxa de conversão, o ticket médio e a retenção de clientes, transformando a experiência de compra em uma vantagem competitiva.
- **Equipe de Marketing e Vendas:** Necessita de ferramentas e insights baseados em dados para compreender o comportamento do consumidor em profundidade, permitindo a criação de campanhas e ofertas personalizadas e eficazes.

2.2. JUSTIFICATIVA E OBJETIVOS

A implementação de uma plataforma que utiliza Inteligência Artificial para potencializar a busca de produtos e orquestrar recomendações personalizadas representa uma resposta estratégica ao problema identificado. A solução proposta visa transformar a experiência de compra, convertendo a intenção do usuário em conversão de forma eficiente. Ao reduzir a frustração e aumentar o engajamento, a plataforma gera um impacto financeiro positivo e fortalece a lealdade à marca.

Objetivo Geral:

Desenvolver o projeto de uma plataforma de e-commerce de ponta, que utilize IA para entregar uma experiência de descoberta de produtos superior e personalizada.

Objetivos Específicos:

- **Do ponto de vista de negócio:** Conceber uma solução que melhore diretamente as métricas de desempenho do e-commerce, como taxa de conversão, engajamento do usuário e valor médio do pedido.
- **Do ponto de vista técnico:** Projetar uma arquitetura de solução completa na nuvem AWS que seja resiliente, escalável, segura e otimizada em custos, servindo como um blueprint para a implementação e operação do sistema em um ambiente de produção.

3. LEVANTAMENTO DE REQUISITOS

Esta seção traduz os objetivos de negócio em especificações técnicas detalhadas, que servem como alicerce para o desenvolvimento da solução. Os requisitos são categorizados em funcionais, que descrevem o comportamento do sistema, e não funcionais, que definem seus atributos de qualidade.

3.1. REQUISITOS FUNCIONAIS (RF)

Os Requisitos Funcionais (RF) descrevem as operações críticas que a plataforma deve executar para cumprir seu propósito principal.

- **RF01 - Módulo de Autenticação:** O sistema deve permitir o cadastro, login e gestão de identidade dos usuários.
 - **Serviço AWS: Amazon Cognito** será utilizado para gerenciar os pools de usuários e o fluxo de autenticação.
- **RF02 - Catálogo de Produtos:** O sistema deve ser capaz de exibir produtos com informações detalhadas.

- **Serviços AWS:** A lógica da aplicação será executada em **Amazon EC2**, com os ativos estáticos (imagens) armazenados no **Amazon S3**.
- **RF03 - Jornada de Compra:** A plataforma deve prover um carrinho de compras funcional e um processo de checkout seguro.
 - **Serviços AWS:** A lógica de negócio será processada em **Amazon EC2**, e os dados transacionais (carrinhos, pedidos) serão persistidos no **Amazon RDS**.
- **RF04 - Busca Conversacional (IA):** O sistema deve implementar uma interface de chatbot para interpretar buscas em linguagem natural.
 - **Serviços AWS:** **Amazon Lex** fornecerá a interface conversacional, **AWS Lambda** executará a lógica de busca e o **Amazon OpenSearch Service** indexará e consultará os dados.
- **RF05 - Sistema de Recomendações (IA):** A plataforma deve gerar recomendações de produtos personalizadas.
 - **Serviço AWS:** **Amazon Personalize** será utilizado para treinar os modelos e gerar as recomendações em tempo real.
- **RF06 - Interface Administrativa:** O sistema deve oferecer uma interface de back-office para a gestão de produtos, usuários e pedidos.
 - **Serviços AWS:** A interface será servida por instâncias **Amazon EC2**, com os dados gerenciados no **Amazon RDS**.

3.2. REQUISITOS NÃO FUNCIONAIS (RNF)

Os Requisitos Não Funcionais (RNF) definem os atributos de qualidade que garantem a excelência operacional e a competitividade da plataforma. São os diferenciais que asseguram uma experiência de usuário superior.

- **RNF01 - Disponibilidade (99.9%):** A plataforma deve operar com 99,9% de disponibilidade (SLA), exigindo uma arquitetura tolerante a falhas.
 - **Arquitetura AWS:** Será implementada uma arquitetura Multi-AZ, utilizando **Application Load Balancer**, **Auto Scaling Groups** para EC2, e implantações Multi-AZ para **Amazon RDS** e **Amazon ElastiCache**.
- **RNF02 - Escalabilidade:** A arquitetura deve ser elástica, escalando recursos automaticamente para suportar picos de tráfego sem degradação do serviço.
 - **Serviços AWS:** **Amazon EC2 Auto Scaling** gerenciará a camada de aplicação, enquanto o **Amazon RDS Proxy** otimizará as conexões com o banco de dados para suportar a escalabilidade.
- **RNF03 - Performance:** O tempo de carregamento das páginas (p95) não deve exceder 3 segundos, garantindo uma experiência fluida.
 - **Serviços AWS:** **Amazon CloudFront** será usado como CDN para entregar conteúdo com baixa latência, e o **Amazon ElastiCache** servirá como camada de cache em memória para acelerar as respostas.
- **RNF04 - Segurança:** O sistema deve implementar segurança em profundidade e estar em conformidade com a LGPD.
 - **Serviços AWS:** A segurança será garantida pelo **AWS WAF** na borda, **Amazon VPC** com subnets privadas para isolamento de rede, e **Security Groups** para controle de tráfego a nível de instância.
- **RNF05 - Experiência do Usuário (UX):** A plataforma deve oferecer uma interface intuitiva e com design responsivo em múltiplos dispositivos.
 - **Arquitetura AWS:** A performance garantida pelo **Amazon CloudFront** e a alta disponibilidade da arquitetura Multi-AZ são fundamentais para sustentar uma experiência de usuário positiva e consistente.

3.3. PRODUTO MÍNIMO VIÁVEL (MVP)

Para validar as hipóteses de negócio de maior risco com o menor esforço de desenvolvimento, o escopo do MVP será focado em entregar o valor central da proposta:

- **Plataforma de E-commerce Essencial:** Implementação das funcionalidades básicas de cadastro, login, navegação no catálogo, visualização de produtos, carrinho de compras e checkout.
- **Chatbot de Busca (Versão Inicial):** Capacidade de processar consultas em linguagem natural para atributos primários (ex: cor, tipo de item, categoria), validando a interface conversacional.
- **Recomendações Simplificadas:** Exibição de produtos relacionados na página de detalhes do produto (ex: "Você também pode gostar"), baseada em regras simples de associação de itens.

4. METODOLOGIA E PLANEJAMENTO

Para gerenciar a complexidade inerente ao desenvolvimento de uma solução que integra infraestrutura de nuvem robusta com componentes de Inteligência Artificial, o projeto adota uma abordagem ágil. Esta metodologia garante flexibilidade para se adaptar a novos aprendizados, permite entregas de valor incrementais e promove um ciclo de melhoria contínua.

4.1. FRAMEWORK ÁGIL

A gestão do projeto será conduzida utilizando um framework ágil, combinando elementos do Scrum para o desenvolvimento iterativo e do Kanban para a visualização do fluxo de trabalho. Serão adotados os seguintes ritos para garantir alinhamento e progresso contínuo:

- **Planejamento de Sprint (Planning):** Reuniões no início de cada ciclo para definir as metas e selecionar as tarefas do backlog.
- **Reuniões Diárias (Daily Stand-ups):** Encontros rápidos para sincronizar a equipe e remover impedimentos.
- **Revisão da Sprint (Review):** Sessões ao final de cada ciclo para demonstrar o incremento de produto desenvolvido e obter feedback dos stakeholders.
- **Retrospectiva:** Reuniões para a equipe inspecionar seu processo de trabalho e identificar oportunidades de melhoria.

O trabalho será organizado em um quadro visual (ex: Jira, Trello) com colunas como *Backlog*, *To Do*, *In Progress* e *Done*, garantindo total transparência sobre o andamento das atividades.

4.2. BACKLOG DO PRODUTO E ESTRUTURA DE ETAPAS

O backlog do produto foi estruturado em épicos, que representam grandes blocos de funcionalidade ou componentes arquitetônicos. Cada épico se desdobra em histórias de usuário, que por sua vez possuem critérios de aceitação claros.

- **E1 – Edge & Segurança:** Define a implementação da camada de borda da aplicação, focando na segurança, performance de entrega de conteúdo global (CDN) e logging de acesso. *Exemplo de História:* “Como usuário, quero que o site carregue rapidamente e de forma segura, independentemente da minha localização geográfica.”
- **E2 – Rede & Conectividade:** Contempla a construção da fundação de rede na AWS VPC, incluindo o design de subnets, roteamento e conectividade segura.
- **E3 – Compute & Escala:** Foca na implementação da camada de aplicação, garantindo que ela seja elástica e capaz de escalar automaticamente para atender à demanda.
- **E4 – Dados & Cache:** Abrange a configuração da camada de persistência de dados e cache, com foco em alta disponibilidade e performance.
- **E5 – Observabilidade & Governança:** Trata da implementação de ferramentas para monitoramento, alertas, auditoria de segurança e controle de custos.
- **E6 – IA & Autenticação:** Engloba o desenvolvimento e a integração das funcionalidades de Inteligência Artificial e do sistema de autenticação de usuários.

4.3. ESTRATÉGIA DE SPRINTS

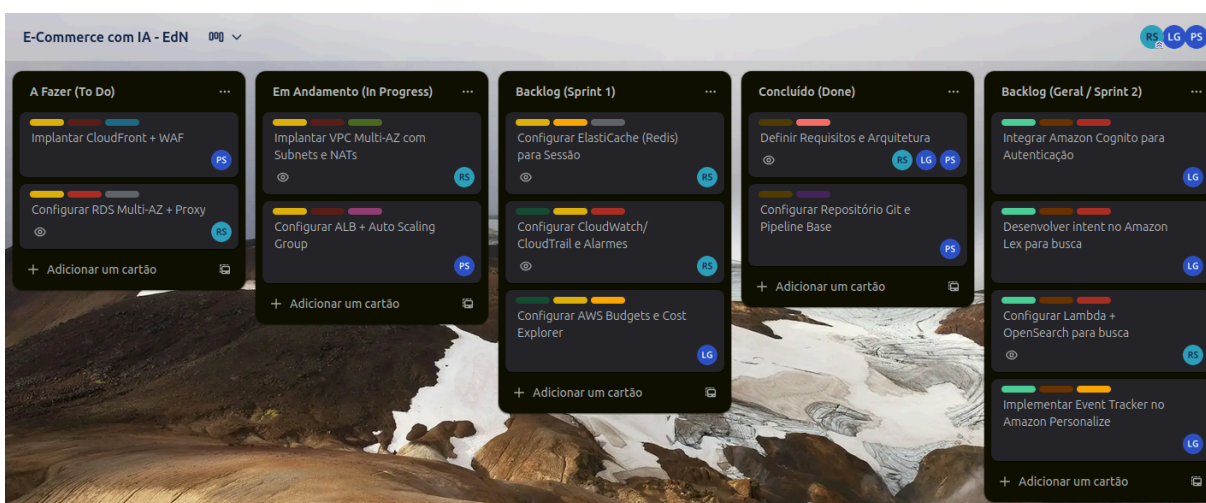
O desenvolvimento inicial será dividido em sprints de duas semanas, com uma estratégia que prioriza a construção de uma base sólida antes da implementação das funcionalidades complexas.

- **Sprint 1 – Base Confiável:** O foco desta sprint é a construção da infraestrutura fundamental na AWS, seguindo os princípios de "Infrastructure as Code" (IaC). A entrega abrange a configuração de CloudFront, WAF, VPC Multi-AZ, ALB, Auto Scaling Group, RDS Multi-AZ com Proxy, ElastiCache e as ferramentas de governança (CloudWatch, CloudTrail). Esta abordagem mitiga riscos ao garantir que a plataforma seja segura, resiliente e observável antes que a lógica de negócio seja implantada. A "Definição de Pronto" (Definition of Done) para esta sprint inclui a infraestrutura totalmente provisionada, testes de failover validados e alarmes de monitoramento ativos.

- **Sprint 2 – IA Essencial:** Construindo sobre a fundação estável da Sprint 1, esta fase foca na entrega do valor central ao usuário. A entrega inclui a integração com Amazon Cognito para autenticação, Amazon Lex para o chatbot, Amazon OpenSearch para as buscas e Amazon Personalize para o sistema de recomendações. A "Definição de Pronto" aqui estará atrelada à validação funcional, como fluxos de login bem-sucedidos e a capacidade do chatbot retornar resultados de produtos relevantes.

4.4. QUADRO DE GESTÃO DO PROJETO (KANBAN/SCRUM)

Para uma demonstração prática da metodologia ágil do projeto, utiliza-se a ferramenta de gestão visual Trello, conforme exemplificado na Figura 1. O quadro traduz o planejamento estratégico em um fluxo de trabalho tangível, organizado em colunas que representam o ciclo de vida das tarefas: *A Fazer (To Do)*, *Em Andamento (In Progress)* e *Concluído (Done)*, além de colunas para o planejamento de sprints (*Backlog (Sprint 1)* e *Backlog (Geral / Sprint 2)*). Cada tarefa (card) é um entregável técnico atribuído a um membro da equipe, Paula Silva, Ryan Ricardo de Souza e/ou Luis Gustavo, e categorizado com etiquetas que informam a etapa (*E2 – Rede & Conectividade*), a prioridade (*Crítico*) e a sprint correspondente.



(Figura 1 – Exemplo do Quadro de Acompanhamento do Projeto no Trello)

A execução da **Sprint 1 - Base Confiável** pode ser observada em tempo real: atividades críticas como a "Implantação da VPC Multi-AZ" e a "Configuração do ALB + Auto Scaling Group" já se encontram *Em Andamento*. Na fila de *A Fazer*, aguardam tarefas essenciais como a "Implantação do CloudFront + WAF" e a "Configuração do RDS Multi-AZ", demonstrando a priorização da infraestrutura fundamental. A coluna *Concluído* valida o progresso, mostrando que as etapas de "Definição de Requisitos e Arquitetura" do Sprint 0 foram finalizadas com sucesso.

Esta organização reflete a estratégia de execução do projeto: mitigar riscos ao construir uma fundação resiliente na Sprint 1 antes de avançar para as inovações de IA na Sprint 2. Essa abordagem em duas fases demonstra a aplicação prática dos princípios ágeis para o desenvolvimento de soluções complexas na nuvem.

5. ARQUITETURA DA SOLUÇÃO

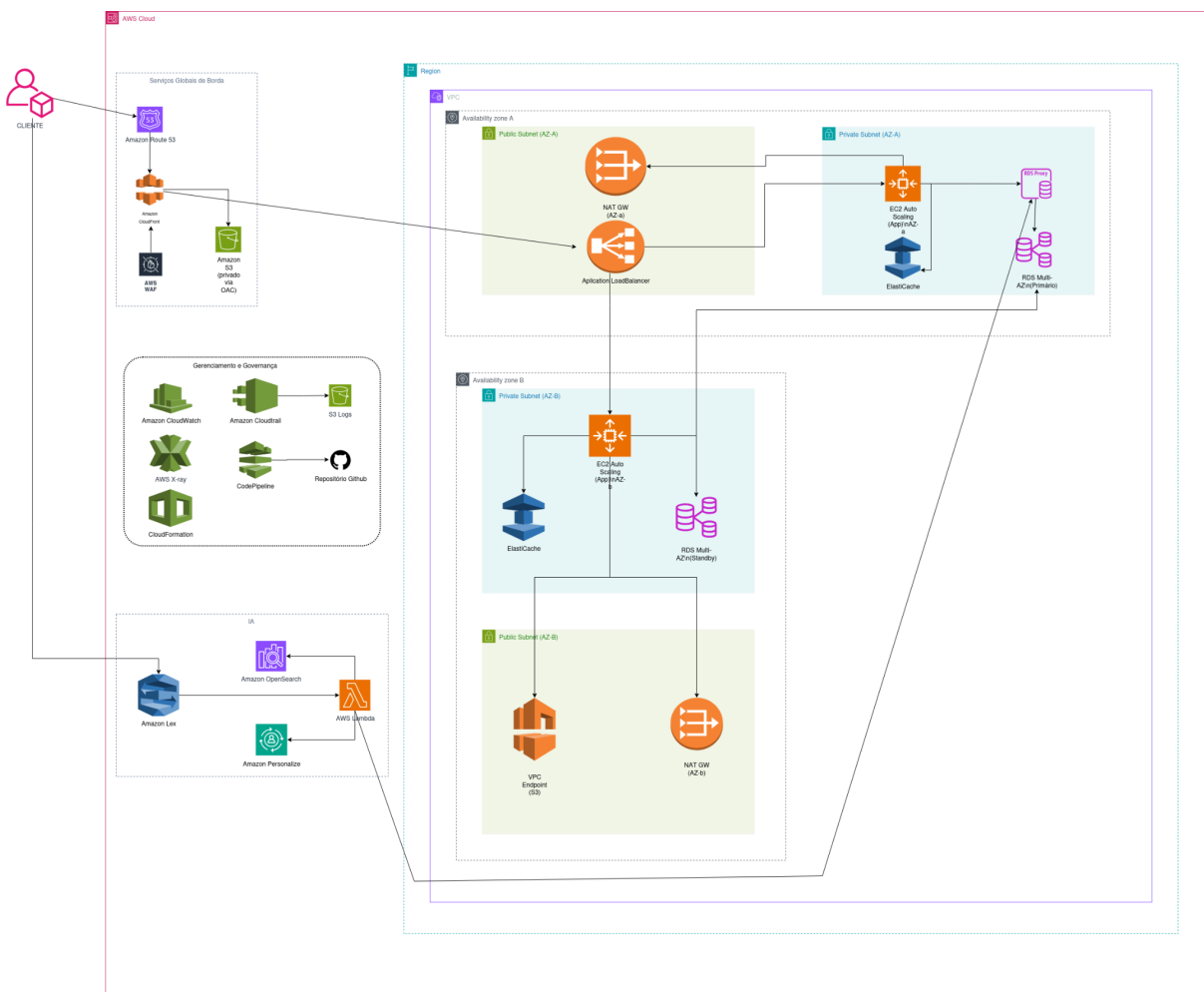
A arquitetura da solução foi meticulosamente projetada para atender aos rigorosos requisitos não funcionais de disponibilidade, escalabilidade, segurança e performance. As decisões de design foram validadas contra os cinco pilares do **AWS Well-Architected Framework** para garantir a construção de uma plataforma robusta, eficiente e resiliente.

5.1. VISÃO GERAL DA ARQUITETURA

A solução adota um padrão de alta disponibilidade distribuído em múltiplas Zonas de Disponibilidade (Multi-AZ). O fluxo de tráfego se inicia com o usuário, cujas requisições são roteadas pelo **Amazon Route 53** para o **Amazon CloudFront**. O CloudFront atua como a borda da aplicação, funcionando como Content Delivery Network (CDN) e como primeiro ponto de inspeção de segurança através do **AWS WAF**.

O CloudFront serve conteúdo estático diretamente de um bucket **Amazon S3** privado, enquanto o tráfego dinâmico é direcionado a um **Application Load Balancer (ALB)** localizado nas subnets públicas da VPC. O ALB, por sua vez, distribui a carga de forma balanceada entre instâncias **Amazon EC2** gerenciadas por um **Auto Scaling Group**, que operam de forma segura nas subnets privadas.

A camada de dados é composta por um cluster **Amazon RDS Multi-AZ**, acessado de forma otimizada e segura pelo **RDS Proxy**, e um cluster **Amazon ElastiCache** para gerenciamento de sessões e cache de dados. Os serviços de Inteligência Artificial, como **Amazon Lex** e **AWS Lambda**, operam de forma integrada, com a função Lambda executando dentro da VPC para acessar de forma segura os recursos da camada de dados.



(Figura 2 – Diagrama da Arquitetura da Solução na AWS)

5.2. ANÁLISE SEGUNDO O AWS WELL-ARCHITECTED FRAMEWORK

5.2.1. Pilar de Confiabilidade

A arquitetura é fundamentalmente resiliente, projetada para se recuperar automaticamente de falhas e garantir a continuidade do serviço.

- **Tolerância a Falhas (Multi-AZ):** A distribuição de todos os componentes críticos (EC2, RDS, ElastiCache, NAT Gateways) em múltiplas Zonas de Disponibilidade (AZs) elimina pontos únicos de falha. Se uma AZ se tornar indisponível, o tráfego é automaticamente redirecionado para os recursos na AZ saudável, garantindo a continuidade da operação.

- **Resiliência do Banco de Dados:** A inclusão do **Amazon RDS Proxy** é uma decisão estratégica que desacopla a aplicação do banco de dados. Em um evento de failover do RDS, o Proxy gerencia a transição de forma transparente para a aplicação, reduzindo o tempo de recuperação em até 66% e minimizando o impacto no usuário final.

5.2.2. Pilar de Segurança

A segurança é implementada em múltiplas camadas (defesa em profundidade) para proteger a plataforma de ameaças.

- **Segurança de Borda:** O **AWS WAF**, integrado ao **Amazon CloudFront**, atua como a primeira linha de defesa, mitigando ataques comuns da web (como SQL Injection e XSS) e ataques de negação de serviço (DDoS) antes que atinjam a infraestrutura principal. O conteúdo estático no S3 é mantido privado e acessível apenas pelo CloudFront através do Origin Access Control (OAC).
- **Isolamento de Rede:** A VPC é segmentada em subnets públicas e privadas. As subnets privadas isolam a camada de aplicação e o banco de dados, impedindo o acesso direto pela internet. O controle de tráfego é refinado com **Security Groups granulares**, que aplicam o princípio do menor privilégio, permitindo a comunicação apenas entre componentes específicos (ex: o Security Group das instâncias EC2 permite tráfego apenas do Security Group do ALB).

5.2.3. Pilar de Eficiência de Performance

A segurança é implementada em múltiplas camadas (defesa em profundidade) para proteger a plataforma de ameaças.

- **Segurança de Borda:** O **AWS WAF**, integrado ao **Amazon CloudFront**, atua como a primeira linha de defesa, mitigando ataques comuns da web (como SQL Injection e XSS) e ataques de negação de serviço (DDoS) antes que atinjam a infraestrutura principal. O conteúdo estático no S3 é mantido privado e acessível apenas pelo CloudFront através do Origin Access Control (OAC).
- **Otimização de Conexões para Serverless:** O **RDS Proxy** é crucial para a performance, especialmente na integração com AWS Lambda. Ele resolve o "impedance mismatch" entre o escalonamento massivo da Lambda e os limites de conexão de um banco de dados relacional, utilizando um pool de conexões para evitar a sobrecarga do banco de dados e garantir a escalabilidade da aplicação.

5.2.4. Pilar de Otimização de Custos

A consciência de custos é integrada ao design da arquitetura.

- **Análise de Custo-Benefício:** Decisões como o uso de NAT Gateways por AZ e do RDS Proxy, embora adicionem custos, são justificadas por prevenirem a indisponibilidade e a necessidade de super provisionamento de recursos mais caros, como instâncias de banco de dados maiores.
- **Estratégias de Otimização:** Recomenda-se a avaliação de instâncias baseadas em processadores **AWS Graviton**, que oferecem uma relação preço-performance até 40% melhor. Para cargas de trabalho previsíveis, a utilização de **Savings Plans** ou **Instâncias Reservadas** pode gerar descontos significativos sobre os preços sob demanda.

5.2.5. Pilar de Excelência Operacional

A arquitetura é projetada para ser monitorada e gerenciada de forma automatizada e proativa.

- **Observabilidade:** O **Amazon CloudWatch** é central para a estratégia de monitoramento, com alarmes críticos configurados para métricas de negócio e saúde da aplicação (ex: erros 5XX no ALB, alta utilização de CPU, latência).
- **Auditoria e Automação:** O **AWS CloudTrail** registra todas as chamadas de API, fornecendo uma trilha de auditoria completa. A integração do CloudTrail com o CloudWatch Logs permite a criação de alertas para eventos de segurança sensíveis. Toda a infraestrutura será definida como código (IaC) via

6. MÉTRICAS DE SUCESSO

O sucesso do projeto será avaliado por um conjunto de Key Performance Indicators (KPIs) que abrangem as dimensões de negócio, produto e operação técnica. Essas métricas permitirão uma análise quantitativa do impacto da solução.

6.1. Métricas de Negócio

- **Taxa de Conversão:** Aumento percentual no número de sessões que resultam em uma compra, comparando períodos antes e depois da implementação das funcionalidades de IA.

- **Aumento do Ticket Médio (AOV - Average Order Value):** Crescimento no valor médio dos pedidos, impulsionado pela eficácia das recomendações personalizadas.
- **Retenção de Clientes e Lifetime Value (LTV):** Aumento na frequência de compra e no valor total gasto por clientes ao longo do tempo, indicando maior fidelização.

6.2. Métricas de Produto e Engajamento

- **Taxa de Cliques (CTR) das Recomendações:** Percentual de usuários que clicam nos produtos recomendados pelo sistema de IA, validando a relevância das sugestões.
- **Taxa de Engajamento e Sucesso do Chatbot:** Medida pela proporção de buscas via chatbot que resultam em um clique no produto e pela redução na taxa de abandono de busca (buscas sem resultado ou sem interação posterior).
- **Adoção das Funcionalidades de IA:** Percentual de usuários ativos que interagem com o chatbot ou com os widgets de recomendação durante sua jornada de compra.

6.3. Métricas Técnicas e Operacionais

- **Disponibilidade (SLA):** Manutenção de uma disponibilidade de serviço $\geq 99,9\%$, monitorada através do Amazon CloudWatch com base nas métricas do Application Load Balancer e health checks.
- **Latência (p95):** Tempo de resposta do 95º percentil para as principais transações da aplicação (carregamento de página, busca, checkout) deve ser inferior a 3 segundos.
- **Taxa de Acerto de Cache (Cache Hit Ratio):** Eficiência do Amazon CloudFront na entrega de conteúdo estático, visando reduzir a carga sobre a origem (ALB e S3).
- **Saúde da Aplicação:** Contagem de erros do lado do servidor (HTTP 5xx) no ALB deve ser próxima de zero. A utilização de CPU do cluster RDS e das instâncias EC2 deve permanecer dentro de limites operacionais saudáveis.

7. CONSIDERAÇÕES FINAIS

O projeto "E-Commerce com IA" representa mais do que uma solução tecnológica para um desafio de mercado; ele constitui um blueprint estratégico para a construção de aplicações nativas em nuvem que são, por design, inteligentes, resilientes e escaláveis. A arquitetura proposta internaliza os princípios do AWS Well-Architected Framework, estabelecendo uma base sólida que equilibra inovação com excelência operacional, segurança e otimização de custos.

A combinação de uma metodologia de desenvolvimento ágil com uma arquitetura de nuvem robusta garante que o projeto não apenas atenda aos requisitos atuais, mas também esteja preparado para evoluir e se adaptar às futuras demandas do negócio. Ao final, esta iniciativa serve como um modelo de referência, demonstrando como a integração estratégica de serviços de IA em uma plataforma bem arquitetada pode gerar valor tangível e uma vantagem competitiva duradoura no cenário digital.

8. REFERÊNCIAS

AMAZON WEB SERVICES. **Amazon Cognito user pools — Developer Guide**.: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-user-pools.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Amazon ElastiCache — Documentation Home**.: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/elasticache/>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Amazon Lex — Sample utterances (Lex V2)**.: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/lexv2/latest/dg/sample-utterances.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Amazon OpenSearch Service — Documentation**.: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/opensearch-service/>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Amazon RDS Proxy — Overview**.: Amazon Web Services, [s. d.]. Disponível em: (<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/rds-proxy.html>). Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Analyzing your costs and usage with AWS Cost Explorer**.: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/cost-management/latest/userguide/ce-what-is.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Application Load Balancers — Cross-zone load balancing**.: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/elasticloadbalancing/latest/application/application-load-balancers.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **AWS Foundational Security Best Practices standard (Security Hub)**.: Amazon Web Services, 2025. Disponível em:

<https://docs.aws.amazon.com/securityhub/latest/userguide/fsbp-standard.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Blocking public access to your Amazon S3 storage.**: Amazon Web Services, [s. d.]. Disponível em: (<https://docs.aws.amazon.com/AmazonS3/latest/userguide/access-control-block-public-access.html>). Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **CloudWatch metrics for your Application Load Balancer.**: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/elasticloadbalancing/latest/application/load-balancer-cloudwatch-metrics.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Configuring and managing a Multi-AZ deployment for Amazon RDS.**: Amazon Web Services, [s. d.]. Disponível em: (<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.MultiAZ.html>). Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Creating a budget (AWS Budgets).**: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/cost-management/latest/userguide/budgets-create.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Encrypting CloudTrail log files, digest files, and event data stores with AWS KMS keys (SSE-KMS).**: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/awscloudtrail/latest/userguide/encrypting-cloudtrail-log-files-with-aws-kms.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Gateway endpoints for Amazon S3 (VPC).**: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/vpc/latest/privatelink/vpc-endpoints-s3.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Health checks for instances in an Auto Scaling group.**: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/autoscaling/ec2/userguide/ec2-auto-scaling-health-checks.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **NAT gateway use cases (VPC User Guide)**.: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/vpc/latest/userguide/nat-gateway-scenarios.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **PutEvents — Amazon Personalize (API Reference)**.: Amazon Web Services, 2025. Disponível em: (https://docs.aws.amazon.com/personalize/latest/dg/API_UBS_PutEvents.html). Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Query Amazon CloudFront logs — Amazon Athena**.: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/athena/latest/ug/cloudfront-logs.html>. Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Restrict access to an Amazon S3 origin (Origin Access Control – OAC)**.: Amazon Web Services, [s. d.]. Disponível em: (<https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-restricting-access-to-s3.html>). Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Standard logging (access logs) — Amazon CloudFront**.: Amazon Web Services, [s. d.]. Disponível em: (<https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/AccessLogs.html>). Acesso em: 5 ago. 2025.

AMAZON WEB SERVICES. **Using AWS WAF with Amazon CloudFront**.: Amazon Web Services, [s. d.]. Disponível em: <https://docs.aws.amazon.com/waf/latest/developerguide/cloudfront-features.html>. Acesso em: 5 ago. 2025.