

DAS (Documento de Arquitetura de Solução): Arquitetura Cloud AWS para Controle de Fluxo de Caixa

A arquitetura proposta visa gerenciar o fluxo de caixa de um comerciante, permitindo o registro de lançamentos (débito e crédito) e a geração de um relatório diário consolidado. O foco principal é a escalabilidade, alta disponibilidade, segurança e integração de sistemas usando serviços gerenciados da AWS.

Arquitetura Combinada AWS com Microsserviços e Padrões Funcionais

1. Definição dos Domínios Funcionais

- **Domínio de Lançamentos (Transações):** Gerencia todo o fluxo de caixa do comerciante, incluindo o registro de débitos e créditos.
 - **Domínio de Consolidação Diária:** Responsável por consolidar todas as transações diárias, gerando relatórios com o saldo diário.
 - **Domínio de Autenticação e Autorização:** Garante a segurança do sistema, controlando quem pode registrar transações e acessar relatórios através do **AWS Cognito**.
 - **Domínio de Monitoramento e Observabilidade:** Utiliza **CloudWatch** e **AWS X-Ray** para garantir o monitoramento contínuo e fornecer uma visão completa da saúde do sistema.
-

2. Capacidades de Negócio

- **Registro de Transações:** O sistema deve ser capaz de lidar com volumes crescentes de transações, garantindo disponibilidade e integridade de dados. As transações serão processadas via **AWS Lambda**, usando um banco de dados relacional gerenciado com **Amazon RDS**.
 - **Consolidação de Saldo:** O serviço de consolidação gerará relatórios diários com base nas transações registradas. Para isso, o sistema usará **Lambda** e armazenamento temporário em **Amazon SQS**, além de um cache em **ElastiCache (Redis)** para otimização da leitura dos dados consolidados.
 - **Autenticação Segura:** Com **OAuth 2.0** através do **AWS Cognito**, o sistema permite a autenticação e autorização robusta, protegendo o acesso às funções críticas do sistema.
-

3. Padrões Arquiteturais

- **Microserviços:** A arquitetura é baseada em microserviços, desacoplando as funcionalidades de controle de transações e consolidação. Cada microserviço é executado de forma independente, permitindo escalabilidade e resiliência.
 - **API Gateway:** O **Amazon API Gateway** gerencia as requisições e integra os microserviços de forma segura, fornecendo roteamento e escalabilidade. Ele conecta o frontend com os serviços backend.
 - **Mensageria:** **Amazon SQS** gerencia o fluxo assíncrono de mensagens entre o serviço de transações e o serviço de consolidação, garantindo resiliência e processamento desacoplado.
 - **Banco de Dados Relacional e Não Relacional:** **Amazon RDS** (PostgreSQL ou MySQL) é utilizado para a persistência dos dados de transações e consolidação. **Amazon ElastiCache** (Redis) oferece um cache para otimizar o desempenho da leitura de relatórios consolidados.
-

4. Escalabilidade e Disponibilidade

- **Escalabilidade Horizontal:** Os microserviços em **AWS Lambda** escalam automaticamente de acordo com a demanda. **Amazon RDS Multi-AZ** oferece

alta disponibilidade e failover automático para garantir que o sistema continue operacional mesmo em caso de falhas de infraestrutura.

- **Cache Distribuído: Amazon ElastiCache** garante respostas rápidas para consultas frequentes, especialmente em momentos de pico, otimizando o desempenho.
-

5. Segurança

- **Autenticação e Autorização: AWS Cognito** oferece autenticação segura, com suporte a **OAuth 2.0** e autenticação multifator (MFA), protegendo o acesso ao sistema. Cada chamada de API é autenticada e autorizada para garantir que apenas usuários e sistemas válidos possam interagir com o sistema.
 - **Criptografia:** Todos os dados são criptografados tanto em trânsito (via HTTPS/TLS) quanto em repouso (via AWS KMS para bancos de dados e armazenamento de logs).
-

6. Integração

- **APIs RESTful:** Os microsserviços comunicam-se entre si via APIs RESTful gerenciadas pelo **API Gateway**, facilitando a escalabilidade e a integração com outros sistemas.
 - **gRPC:** Para comunicação interna entre os microsserviços, **gRPC** pode ser usado para garantir maior eficiência e desempenho nas trocas de mensagens.
-

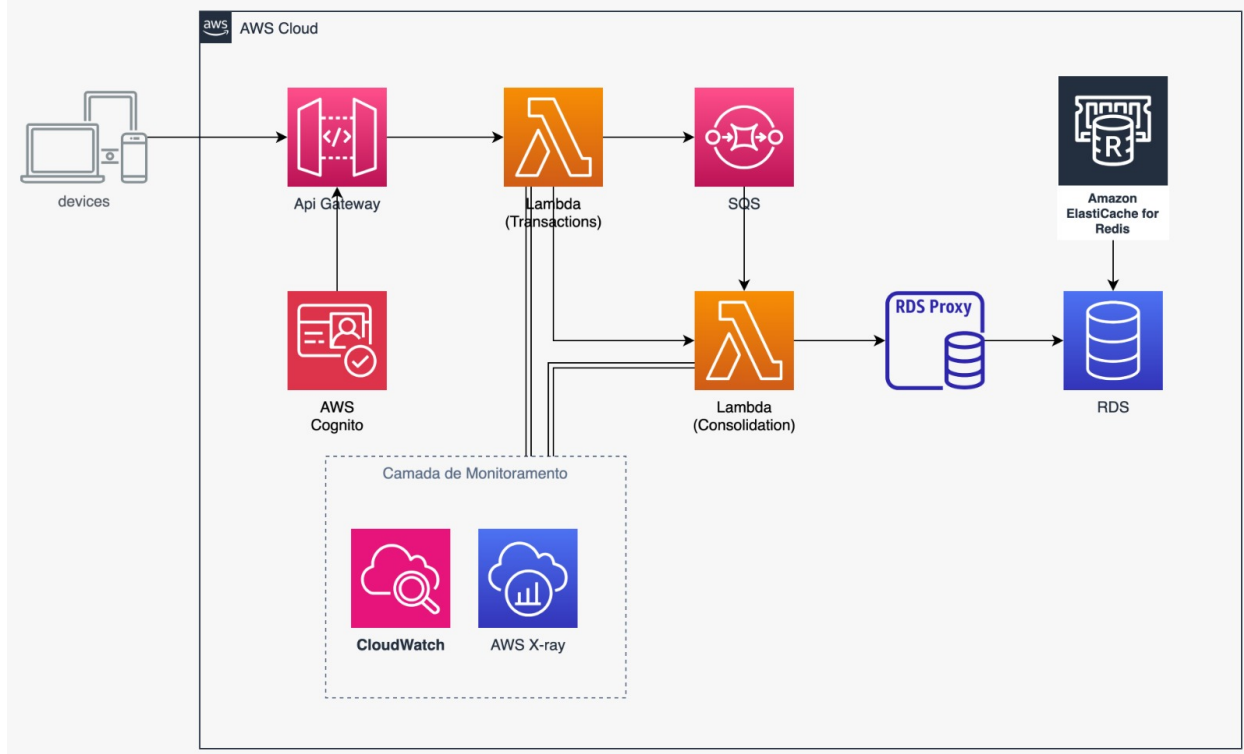
7. Monitoramento e Observabilidade

- **Logs Centralizados: AWS CloudWatch** centraliza os logs de todas as funções **Lambda** e componentes do sistema. Isso facilita o monitoramento em tempo real e a resposta a problemas.
- **Monitoramento de Saúde: AWS X-Ray** é usado para rastreamento distribuído, fornecendo uma visão abrangente da saúde de todo o sistema. **Grafana** e **Prometheus** podem ser integrados para gerar dashboards e monitorar métricas de desempenho.

- **Tracing Distribuído:** Ferramentas como **AWS X-Ray** e **CloudWatch Logs** são utilizadas para capturar e rastrear requisições entre os microsserviços, facilitando a detecção de problemas.
-

Desenho Final da Solução

1. **Frontend:** Aplicação web/mobile que se comunica via **API Gateway** com os microsserviços backend para registrar transações e acessar relatórios.
2. **API Gateway:** Gerencia as requisições, provendo segurança e escalabilidade para as chamadas aos microsserviços.
3. **Serviço de Lançamento (Transações):** Executado em **AWS Lambda**, registra transações e persiste os dados em **Amazon RDS**.
4. **Fila de Mensageria (Amazon SQS):** Garante comunicação assíncrona entre o serviço de lançamentos e o serviço de consolidação.
5. **Serviço de Consolidação:** Consome da fila de mensagens, realiza a consolidação diária e persiste o saldo em **RDS** e **ElastiCache**.
6. **Banco de Dados:** **Amazon RDS** para armazenamento de dados transacionais e **ElastiCache** para melhorar a performance de consultas de relatórios.
7. **Monitoramento:** **CloudWatch** e **AWS X-Ray** garantem a observabilidade e o rastreamento do sistema, com métricas e logs centralizados.



Conclusão

A arquitetura combinada proporciona uma solução eficiente, escalável e segura para o controle de fluxo de caixa, com uma abordagem clara de microsserviços que garante a independência dos componentes e facilita a manutenção e evolução do sistema. Além disso, o uso dos serviços gerenciados da AWS minimiza a necessidade de gerenciamento de infraestrutura, permitindo foco no desenvolvimento de valor para o negócio.