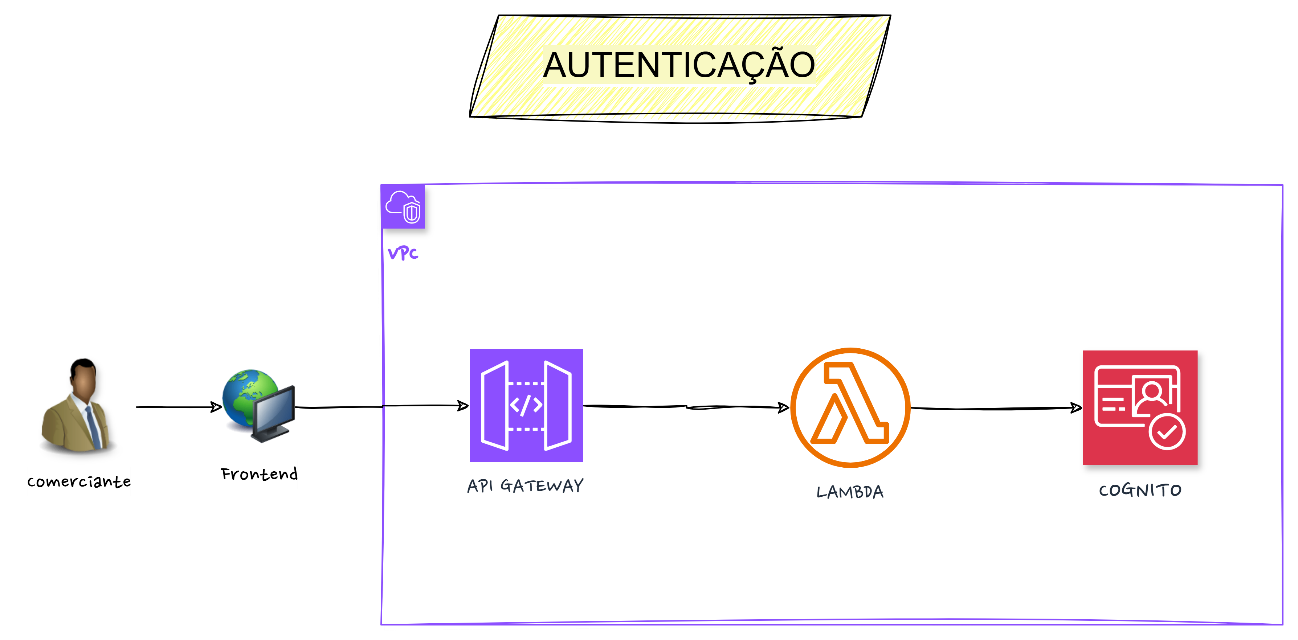
**Controle de Fluxo de Caixa**

**Proposta de Solução**

Este documento visa descrever a proposta para o sistema de fluxo de caixa, abordando dúvidas sobre a arquitetura proposta. A solução busca um crescimento rápido e sustentável do negócio, garantindo robustez, segurança e eficiência estratégica.

**Componentes Utilizados**

* **VPC (Virtual Private Cloud)**: Cria uma rede isolada na AWS para hospedar recursos com segurança, permitindo a definição de sub-redes, tabelas de rotas e controle de acesso granular.
* **Amazon Cognito**: Gerencia autenticação e autorização de usuários, com funcionalidades de registro, login, recuperação de senha e verificação de e-mail.
* **API Gateway**: Ponto de entrada para todas as chamadas da API, oferecendo autenticação, autorização, proteção contra ataques DDoS e roteamento de tráfego.
* **AWS Lambda**: Serviço de computação sem servidor que executa código em resposta a eventos, como solicitações de API ou alterações em dados. Permite executar funções sob demanda.
* **Amazon EKS e Kubernetes**: Utiliza Kubernetes para orquestrar contêineres, garantindo escalabilidade automática dos pods conforme a demanda.
* **Serviço de Balanceamento de Carga (Load Balancer)**: Distribui chamadas entre os pods da API, garantindo alta disponibilidade e desempenho.
* **Pods da API**: Processam requisições recebidas e se comunicam com outros componentes, facilitando manutenção e escalabilidade independente.
* **RDS**: Banco de dados relacional organizado e eficiente com suporte a transações ACID e consultas complexas.
* **MySQL**: Banco de dados relacional robusto, com suporte a transações e consultas complexas, ideal para administração e otimização de dados.
* **Amazon ElastiCache**: Serviço gerenciado de cache que facilita o uso de banco em memória para reduzir carga no banco de dados e melhorar a escalabilidade.
* **Redis**: Banco de dados na memória usado para cache e armazenamento chave-valor, acelerando consultas e gerenciando sessões com eficiência.
* **Amazon EventBridge:** Serviço gerenciado de eventos que conecta aplicativos e recursos da AWS, permitindo a execução de ações em resposta a eventos em tempo real ou programados.
* **Amazon CloudWatch:** Serviço de monitoramento que coleta e armazena logs e métricas de recursos da AWS, permitindo análise e criação de alarmes para otimizar e depurar aplicativos.
* **Amazon SES (Simple Email Service):** Serviço de e-mail em nuvem que permite enviar e-mails em massa de maneira econômica e escalável, utilizado para notificar usuários, enviar relatórios ou confirmações.



**1. Visão Geral**

Este serviço centraliza o controle de acesso dos usuários em toda a aplicação, utilizando o **Amazon Cognito** para fornecer uma solução segura, escalável e eficiente de **autenticação** e **autorização**.

**2. Componentes Principais**

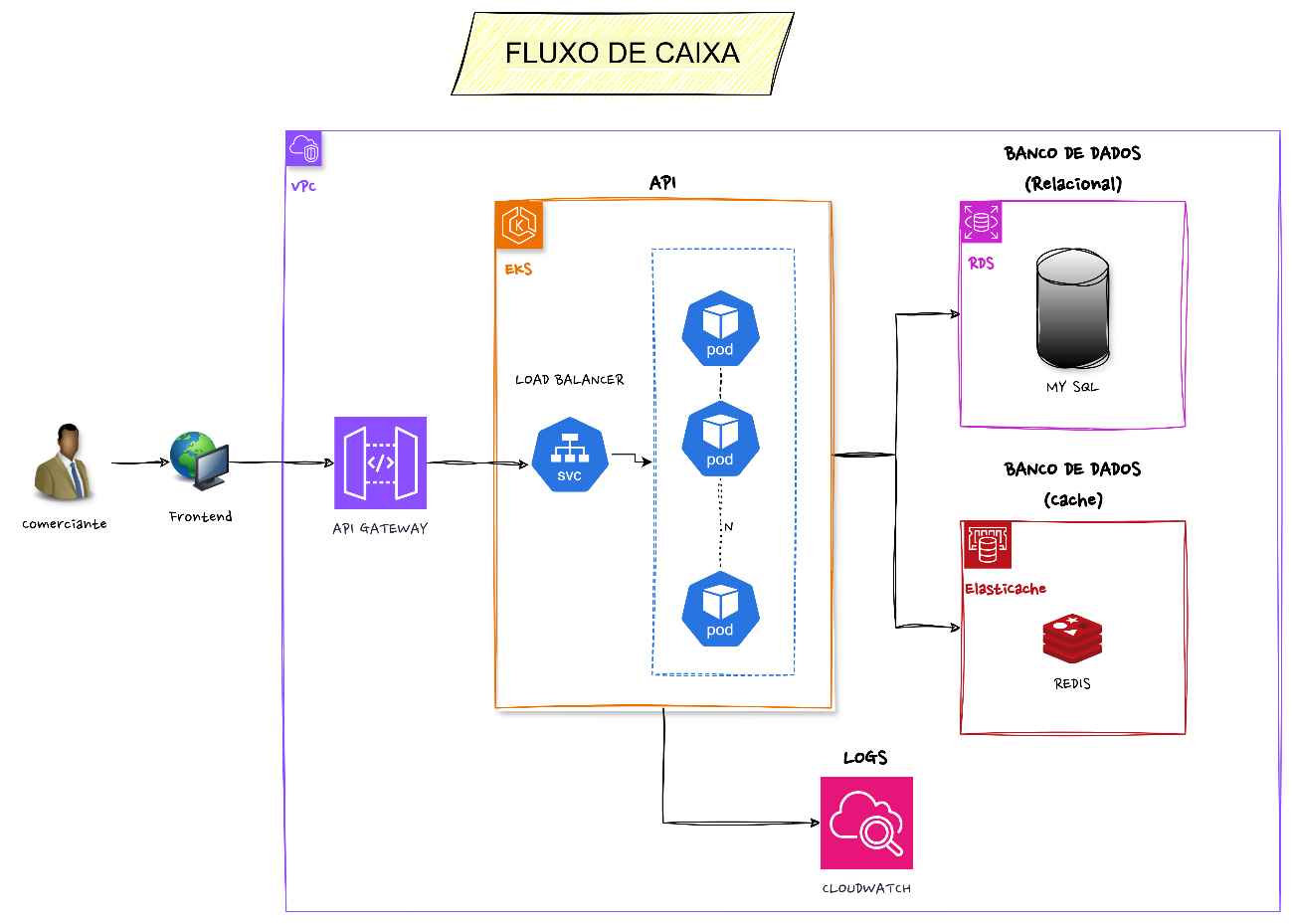
* **VPC (Virtual Private Cloud)**
* **API Gateway**
* **AWS Lambda**
* **Amazon Cognito**

**3. Explicação Detalhada**

O processo de **autenticação** começa com o **frontend**, que faz uma requisição ao **API Gateway**. O API Gateway então redireciona a requisição para uma função **AWS Lambda**, que pode realizar ações de criação, alteração e validação de usuários dentro do **Amazon Cognito**.

O **Amazon Cognito** é responsável por **autenticar** os usuários cadastrados, verificando as credenciais fornecidas e, caso válidas, emite um **token JWT** (JSON Web Token). Esse token é utilizado pelo **frontend** para realizar **chamadas autenticadas** à API de fluxo de caixa, garantindo que apenas usuários autorizados possam acessar os dados sensíveis.

Além disso, o **Cognito** pode ser configurado para incluir funcionalidades como recuperação de senha, verificação de e-mail e multifator de autenticação, proporcionando uma camada adicional de segurança ao processo.



**1. Visão Geral**

O sistema de fluxo de caixa diário permite o gerenciamento de débitos e créditos, proporcionando o controle eficiente das transações financeiras. Ele oferece uma interface robusta para registrar e consultar lançamentos, além de garantir escalabilidade e alta disponibilidade.

**2. Componentes Principais**

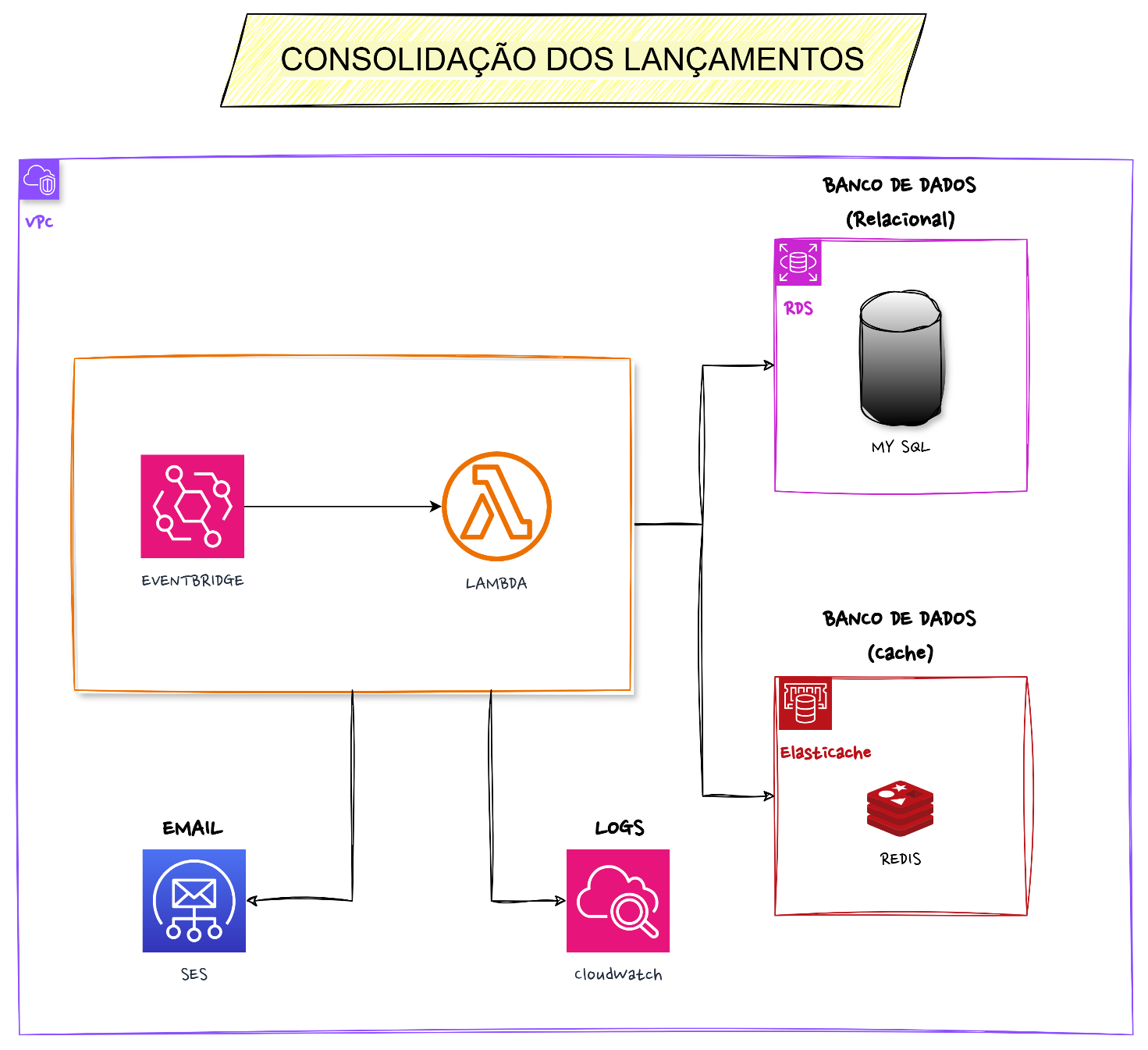
* **VPC (Virtual Private Cloud)**
* **API Gateway**
* **Amazon EKS e Kubernetes**
* **Serviço de Balanceamento de Carga (Load Balancer**
* **Réplicas do Serviço (Pods)**
* **RDS (MySQL)**
* **ElastiCache (Redis)**
* **CloudWatch**

**3. Explicação Detalhada**

A **API** é acessada pelo frontend através de um **Load Balancer** (SVC), que distribui as requisições entre os **pods** disponíveis, que são gerenciados pelo **Amazon EKS**. O EKS foi projetado para lidar com alta demanda de acessos, garantindo escalabilidade automática. A arquitetura inicia com um mínimo de 2 **pods**, podendo escalar automaticamente até N pods conforme a demanda, garantindo desempenho e disponibilidade.

A **gravação de dados** é realizada no **MySQL**, que está armazenado no **RDS**, proporcionando um banco de dados robusto para transações financeiras. As **consultas mais frequentes** são armazenadas no **Redis**, usando o **ElastiCache**, para otimizar a performance e reduzir a carga no banco de dados, melhorando a velocidade de resposta.

Todas as **execuções internas da API** (como erros e eventos) são registradas no **CloudWatch**, permitindo o monitoramento contínuo, rastreamento de erros e a análise de métricas do sistema. Isso facilita a gestão operacional, identificando rapidamente problemas e melhorando a confiabilidade da aplicação.



**1. Visão Geral**

Este processo realiza, ao final de cada dia, a consolidação dos lançamentos de débitos e créditos no fluxo de caixa, armazenando os dados consolidados no banco de dados para gerar um relatório disponível para consulta.

**2. Componentes Principais**

* **Amazon EventBridge**
* **AWS Lambda**
* **RDS (MySQL)**
* **ElastiCache (Redis)**
* **CloudWatch**
* **Amazon SES**

**3. Explicação Detalhada**

Ao final de cada dia, o **Amazon EventBridge** é configurado para disparar um evento, acionando uma função **AWS Lambda**. Essa função se conecta ao **RDS (MySQL)**, lê todos os lançamentos do dia e realiza a consolidação dos dados financeiros.

Os dados consolidados são então gravados em uma **tabela de histórico** no próprio **MySQL**, garantindo que todas as transações sejam registradas de forma segura e estruturada. Além disso, **informações resumidas** podem ser armazenadas no **ElastiCache (Redis)** para otimizar consultas futuras e reduzir a carga no banco de dados.

Após a execução bem-sucedida da consolidação, o **AWS Lambda** aciona o **Amazon SES** para enviar um **e-mail de notificação** aos usuários informando que a consolidação foi concluída com sucesso e que o **relatório consolidado** já está disponível para consulta.

O **Amazon CloudWatch** registra todos os logs e eventos, permitindo o monitoramento e auditoria do processo, além de auxiliar na resolução de problemas caso ocorram falhas durante a execução.