

Este sistema permite a los usuarios acceder al historial de sus movimientos, realizar transferencias y pagos entre cuentas propias e interbancarias. La información del cliente se extrae de dos sistemas: una plataforma Core que contiene datos básicos del cliente, movimientos y productos, y un sistema independiente que complementa la información del cliente cuando se requieren detalles adicionales.

Debido a normativas, el sistema notifica a los usuarios sobre los movimientos realizados. Para cumplir con esto, se integran al menos dos sistemas de notificación, tanto internos como externos.

Arquitectura del Sistema

1. Frontend

El sistema cuenta con dos aplicaciones frontend:

- **Single Page Application (SPA)** en React para la web, utilizando una arquitectura de micro-frontends con Single SPA para modularidad y escalabilidad.
- **Aplicación móvil** en React Native para dispositivos móviles, empleando una arquitectura de micro-frontends mediante módulos instalables, lo cual permite una alta flexibilidad y facilidad de actualización.

Justificación de la elección de tecnologías: La elección de React y React Native, junto con una arquitectura de micro-frontends, permite compartir componentes y lógica de negocio entre la aplicación web y móvil, optimizando el desarrollo y mantenimiento. Esta arquitectura modular facilita la integración y escalabilidad del sistema. No se optó por Angular en el desarrollo web ni Flutter en la aplicación móvil para mantener una coherencia en el stack tecnológico basado en React. Angular y Flutter, aunque potentes, requerirían un conocimiento adicional en diferentes frameworks y lenguajes (TypeScript para Angular y Dart para Flutter), lo cual aumentaría la complejidad y los recursos necesarios en el equipo de desarrollo.

2. Backend y Microservicios

El backend está construido sobre una arquitectura de microservicios desplegados en AWS EKS (Elastic Kubernetes Service). Estos microservicios son responsables de distintas funcionalidades específicas, como la gestión de usuarios, autenticación, y operaciones de transferencia.

Componentes Clave del Backend:

- **EKS:** Permite el despliegue y orquestación de los microservicios, proporcionando escalabilidad y eficiencia en costos.
- **Multi-región para catastro:** Se utiliza un despliegue multi-región para garantizar disponibilidad y redundancia de los datos.
- **Patrón SAGA:** Se emplea el patrón SAGA para coordinar transacciones distribuidas.
- **Patrón de Auditoría Centralizado:** Sistema de auditoría que registra todas las acciones de los clientes.

3. Infraestructura en AWS

La infraestructura del sistema se despliega en AWS, utilizando servicios como:

- **S3** para almacenamiento de documentos legales.

- **CloudFront** para distribución de contenido estático.
- **RDS y DynamoDB** para manejo de datos relacionales y no relacionales.
- **Auth0** para autenticación basada en OAuth2.0.

4. Sistema de Autenticación y Autorización

El sistema utiliza el estándar OAuth2.0 a través de Auth0 para autenticar a los usuarios. Se recomienda el flujo de autenticación de Authorization Code.

5. Onboarding con Reconocimiento Facial

Para el flujo de Onboarding en la aplicación móvil, se utiliza Auth0 con reconocimiento facial. Esto permite que el nuevo usuario, una vez registrado, pueda acceder al sistema mediante usuario y clave, huella digital u otro método. Auth0 ofrece servicios integrados de verificación biométrica que simplifican la implementación del reconocimiento facial.

6. Persistencia de Información y Auditoría

El sistema utiliza una base de datos de auditoría que registra todas las acciones del cliente.

7. Integración y Servicios de API

El sistema cuenta con una capa de integración compuesta por un API Gateway que administra las llamadas a servicios, tales como:

1. Consulta de datos básicos
2. Consulta de movimientos
3. Transferencias

Consideraciones Normativas

El sistema cumple con las siguientes regulaciones y mejores prácticas para entidades financieras:

- **Ley de Protección de Datos Personales:** Asegurar la privacidad de los datos.
- **Regulaciones de Seguridad Financiera:** Implementación de autenticación multifactorial.

Requerimientos de Arquitectura

- **Alta Disponibilidad (HA):** Despliegue multi-región y balanceadores de carga.
- **Tolerancia a Fallos y Recuperación ante Desastres (DR):** Uso de backups.
- **Excelencia Operativa y Auto-Healing:** Monitoreo continuo.

Diagrama de Arquitectura C4

Puedes visualizar el diagrama C4 completo [\[aquí\]](#).

Conclusión

Esta arquitectura modular y desacoplada permite a la institución financiera operar de manera segura, cumpliendo con las normativas y adaptándose a futuras necesidades de negocio.

Extra

Link de Github - [\[aquí\]](#).

ADR propuesto para la arquitectura - [\[aquí\]](#).