Informe de Evaluación Arquitectónica del Sistema de Banca Digital

1. Introducción

El presente documento tiene como objetivo describir y justificar la arquitectura propuesta para el Sistema de Banca Digital, evaluando los elementos técnicos, normativos y de diseño de la solución. Esta arquitectura ha sido concebida para satisfacer requisitos funcionales, no funcionales y regulatorios, garantizando seguridad, escalabilidad y mantenibilidad.

2. Justificación de la Solución

La solución propuesta está alineada con los requerimientos funcionales establecidos, incluyendo: registro de usuario, autenticación segura, acceso a historial de movimientos, ejecución de transacciones interbancarias y propias, y envío de notificaciones. Cada decisión arquitectónica se justifica en base a criterios de escalabilidad, seguridad, cumplimiento normativo y experiencia de usuario.

3. Consideraciones Normativas y Operativas

La arquitectura cumple con los siguientes elementos importantes para entidades financieras:

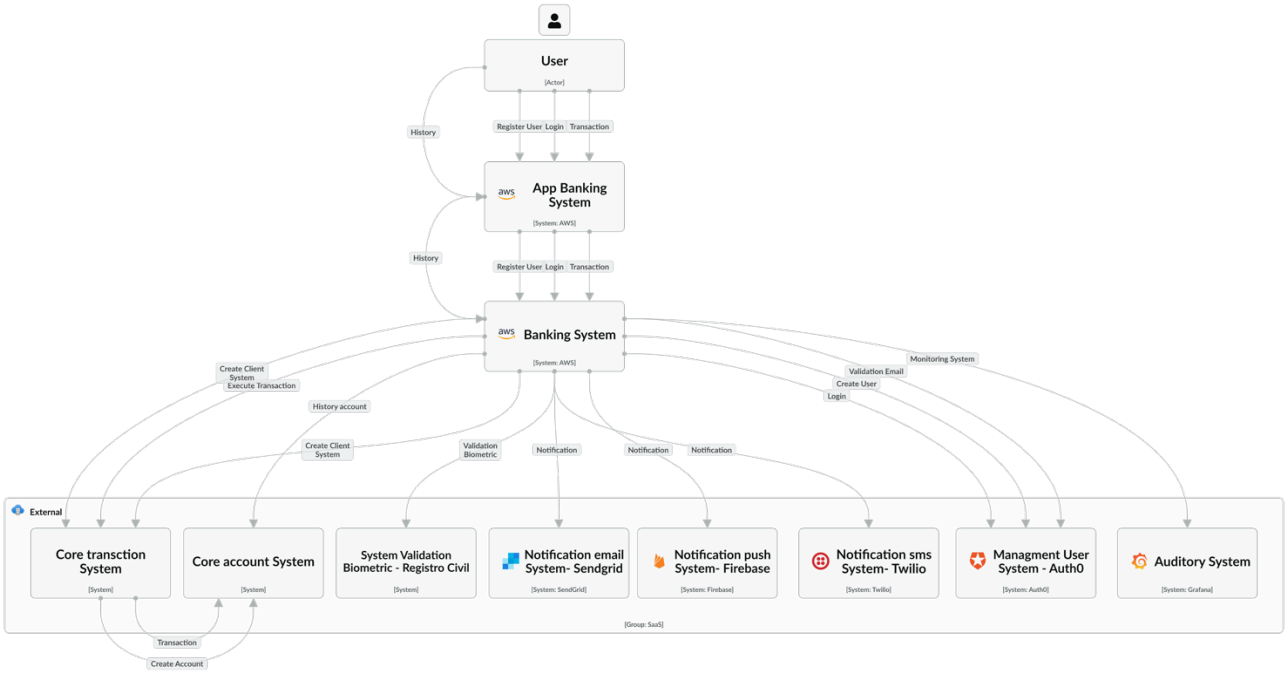
- Ley de Protección de Datos Personales.
- Seguridad Financiera (MFA, OAuth2.0, cifrado de datos sensibles).
- Alta Disponibilidad (HA) mediante despliegue multi-región.
- Recuperación ante Desastres (DR) con backups programados.
- Monitoreo, auto-healing y excelencia operativa.

Se ha optado por una arquitectura desacoplada, modular y escalable que permite fácil integración de nuevos componentes en el futuro.

4. Arquitectura y Diagramas (Modelo C4)

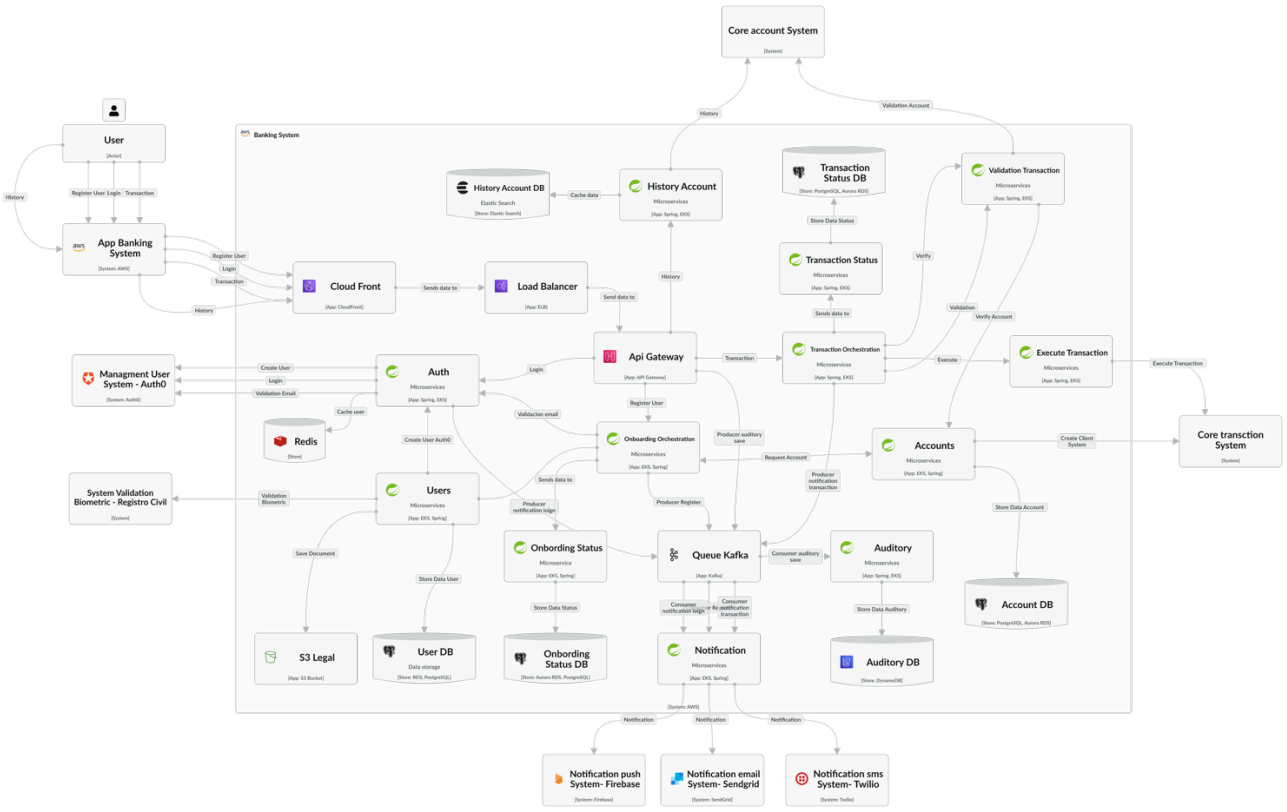
C1 - Diagrama de Contexto

Este diagrama presenta una vista general del sistema, mostrando la interacción entre los usuarios, el sistema bancario central y los servicios externos. El usuario accede a través del "App Banking System" desplegado en AWS, el cual se comunica con el "Banking System" para procesar registro, login y transacciones. Este sistema se integra con el core bancario, servicios biométricos, proveedores de notificaciones (SendGrid, Twilio, Firebase) y el sistema de auditoría (Grafana).



C2 - Diagrama de Contenedores

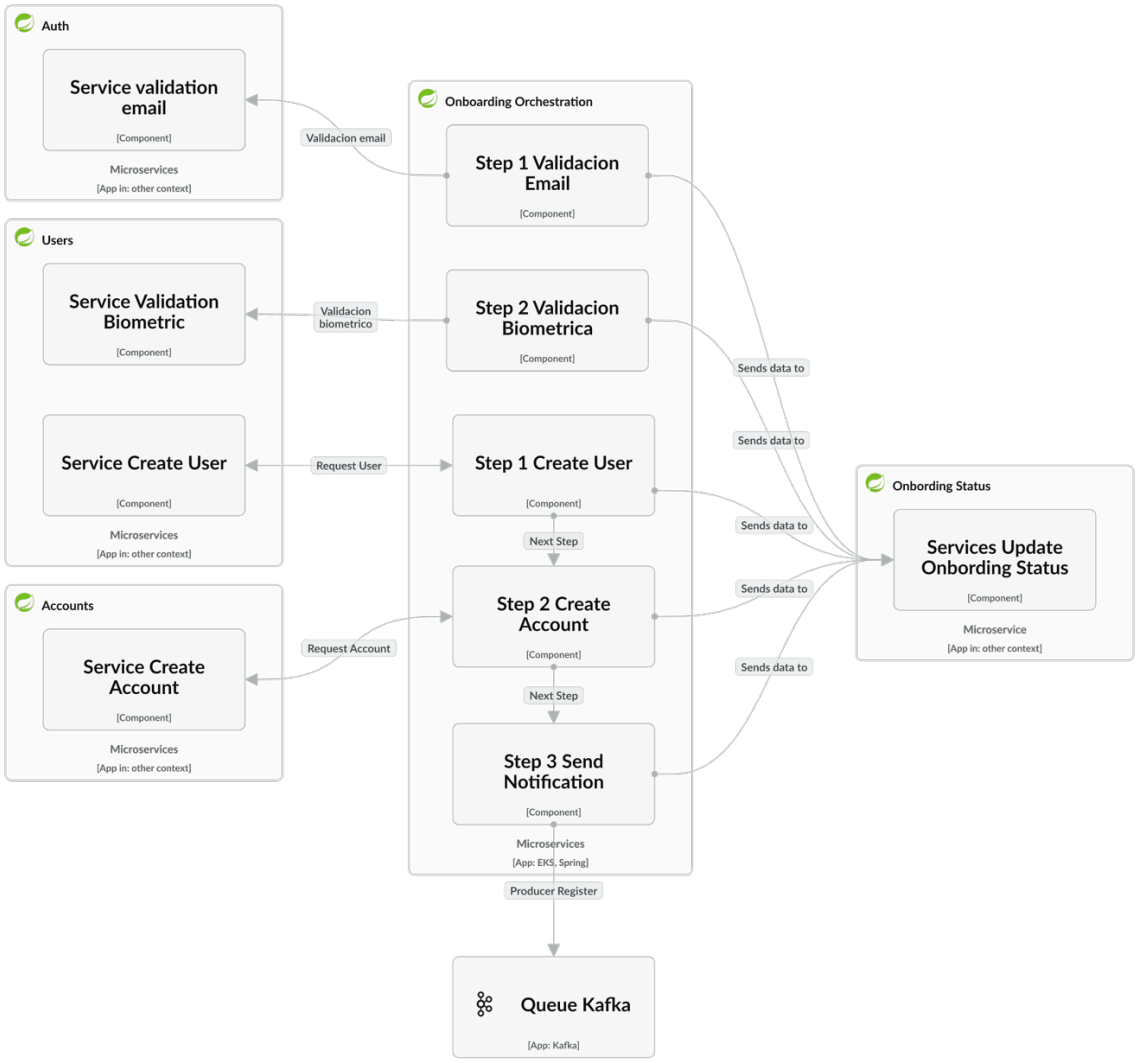
Este diagrama descompone el "Banking System" en contenedores específicos. Se evidencia el uso de microservicios desplegados en AWS EKS y servicios gestionados como Redis, S3, Auth0 y Kafka. Los microservicios cubren autenticación, usuarios, cuentas, transacciones, onboarding, auditoría y notificaciones. Un API Gateway centraliza la entrada y orquesta el flujo hacia cada contenedor según el proceso requerido. La base de datos se gestiona en PostgreSQL, DynamoDB y ElasticSearch según la necesidad del servicio. Se observa cómo se mantiene un flujo desacoplado, orquestado y resiliente mediante colas Kafka y bases de datos específicas por funcionalidad.



C3 - Diagrama de Componentes

Este nivel describe internamente componentes como onboarding, gestión de usuarios, procesamiento de transacciones, capa de auditoría, etc.

Ejemplo Onbording Orchestration



5. Segmentación de Responsabilidades y Desacoplamiento

La arquitectura de microservicios permite un desacoplamiento total entre funcionalidades como autenticación, notificaciones, transacciones y auditoría. Cada servicio tiene responsabilidades claramente delimitadas y puede escalarse de forma independiente.

6. Patrones de Arquitectura

Se implementan los siguientes patrones:

- Microservicios sobre AWS EKS
- Patrón SAGA para transacciones distribuidas
- Auditoría Centralizada
- Capa API Gateway para orquestación
- Frontend modular (micro frontends)

7. Integraciones Externas

El sistema se integra con:

- Sistema biométrico del registro civil
- Core bancario transaccional y de cuentas
- Sistemas de notificación (SendGrid, Firebase, Twilio)
- Servicio de autenticación (Auth0)

8. Arquitectura Frontend y Móvil

El frontend está desarrollado en React para la web y React Native para la móvil, utilizando micro-frontends para lograr modularidad, escalabilidad y mantenibilidad.

9. Acceso a Datos

La información se almacena y consulta desde RDS (relacional) y DynamoDB (NoSQL), dependiendo del tipo de dato. Las acciones del usuario se almacenan en una base de datos de auditoría.

10. Arquitectura Cloud (AWS)

La infraestructura está desplegada en AWS y utiliza:

- S3, CloudFront, EKS, RDS, DynamoDB
- Autenticación con Auth0 (OAuth2.0)
- Balanceadores de carga y multi-región para alta disponibilidad

11. Costos y Escalabilidad

El uso de servicios serverless, contenedores orquestados (EKS) y almacenamiento distribuido permite optimizar costos y escalar de forma automática según demanda.

12. Autenticación y Seguridad

Se emplea OAuth2.0 con Auth0, MFA, y flujos de autorización Code Flow. El onboarding incluye verificación biométrica con reconocimiento facial.

13. Integración con Onboarding

El onboarding se realiza desde la aplicación móvil e integra validaciones biométricas y creación de usuario, gestionadas a través de Auth0.

14. Auditoría

Toda acción relevante es auditada y almacenada en una base de datos. Grafana permite visualizar la actividad del sistema en tiempo real.

15. Regulaciones y Seguridad

Se cumple con:

- Ley de Protección de Datos
- Seguridad financiera (MFA, cifrado, segregación de datos)

16. Alta Disponibilidad y Tolerancia a Fallos

Se utiliza una arquitectura multi-región, backups programados, balanceo de carga, y mecanismos de auto-healing.

17. Monitoreo

Grafana + Prometheus para monitoreo de infraestructura y servicios, alertas en tiempo real, y paneles customizados para cada microservicio.

Conclusión

La arquitectura propuesta es modular, segura, escalable y cumple con todos los criterios normativos y técnicos requeridos. Se apoya en tecnologías modernas y servicios cloud para garantizar una operación eficiente y resiliente.

Links

Icepanel diagramas C4:
https://s.icepanel.io/YgnsRceG2Wj9FX/EYxq/landscape/diagrams/viewer?diagram=4FKBv3jp9s&model=MQ0LYMfVsn&overlay_tab=status&overlay_group=all&x1=-1334.6&y1=-412.4&x2=1672&y2=1307.6

Github: https://github.com/dev-ccazares/arquitectura-bp