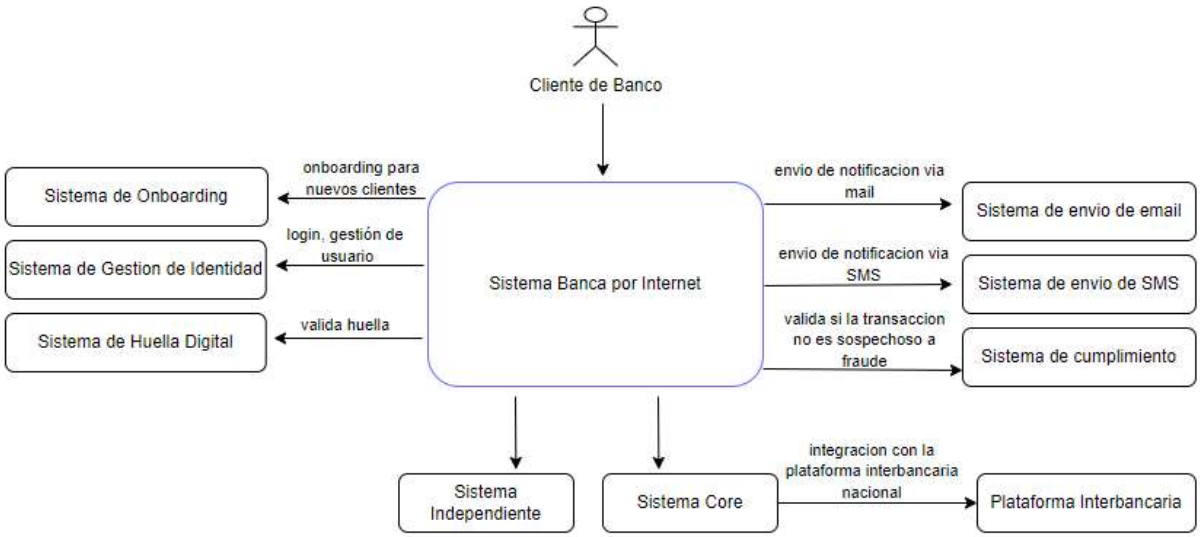
Documento de Proyecto del Sistema de Banca por Internet

1.- Diagrama de Contexto

El Sistema de Banca por Internet contará con un conjunto de sistemas complementarios para el onboarding, la gestión de identidad con el estándar Ouath 2.0, Sistema de huella digital para la validación de la huella, para las notificaciones se integrará con los sistemas de envió de email, envío SMS, para garantizar que la transacción ha sido fraudulenta se integrara al Sistema de cumplimiento. Para visualizar sus movimientos, hacer transacciones se integrará al Sistema Core bancario y para obtener mayor información del cliente a través de un Sistema independiente. Para las transacciones interbancarias el Sistema Core ya se encuentra interconectada con la Plataforma Interbancaria Nacional.



2.- Diagrama de Contenedores

En la capa frontend, la aplicación web SPA usando Angular y la aplicación mobil usando React Native como mejor opción frente a Flutter porque cuenta como lenguaje de programación TypeScript y la curva de aprendizaje y mantenibilidad sería más rápido y porque cuenta con un ecosistema de desarrollo más amplio posibilitándonos mayor número de librerías para el proyecto y soporte.

Las invocaciones a la capa backend de las APIs seria a través del API Gateway que tendría las funciones de enrutar las peticiones, invocar a la validación el token de autentificación, a la validación de la autorización e invocar para que las peticiones se guarden en una base de auditoria de forma asincrona.

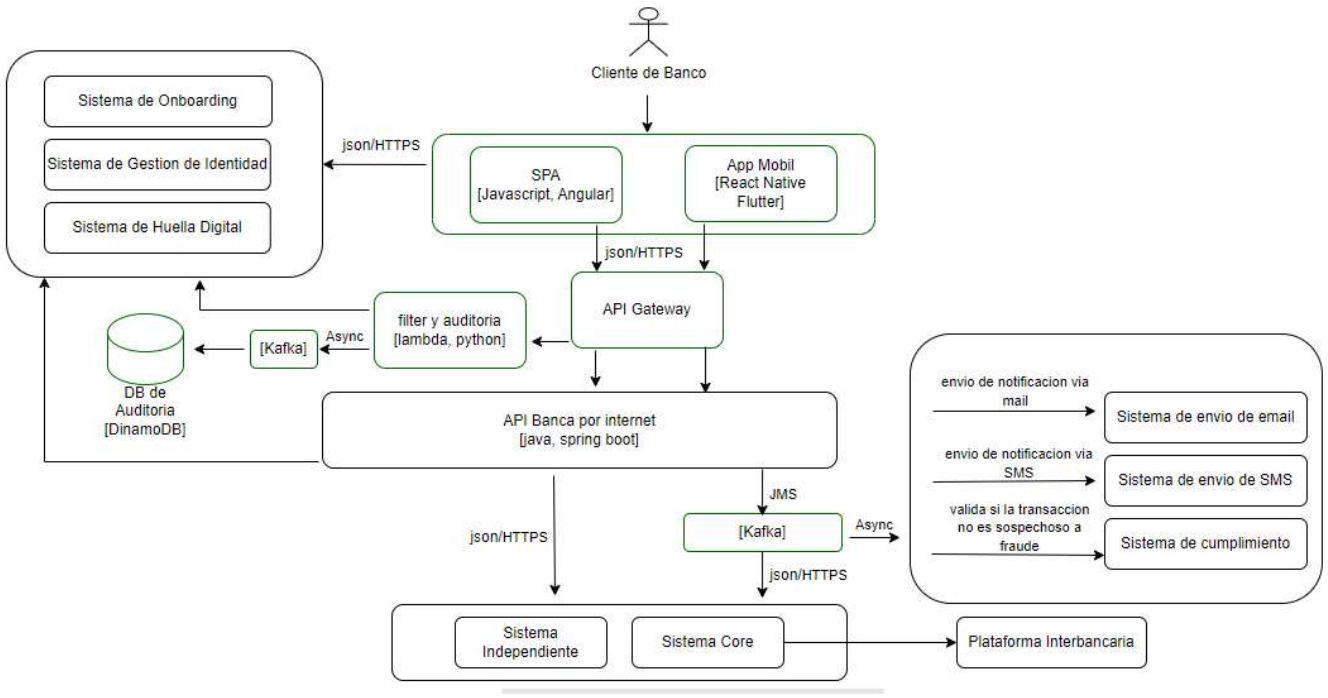
La Capa de backend es la que se comunica con el Sistema Core, con el Sistema independiente

Los sistemas onboarding, el sistema de gestión de identidad, el sistema de huella digital será necesario para todo el proceso de onboarding y autentificación.

La invocación a los servicios del sistema de notificaciones de email y SMS se realizará de forma asíncrona.

También se hará uso del sistema de cumplimiento para salvaguardar que la transacción sospechosa no sean procesadas.

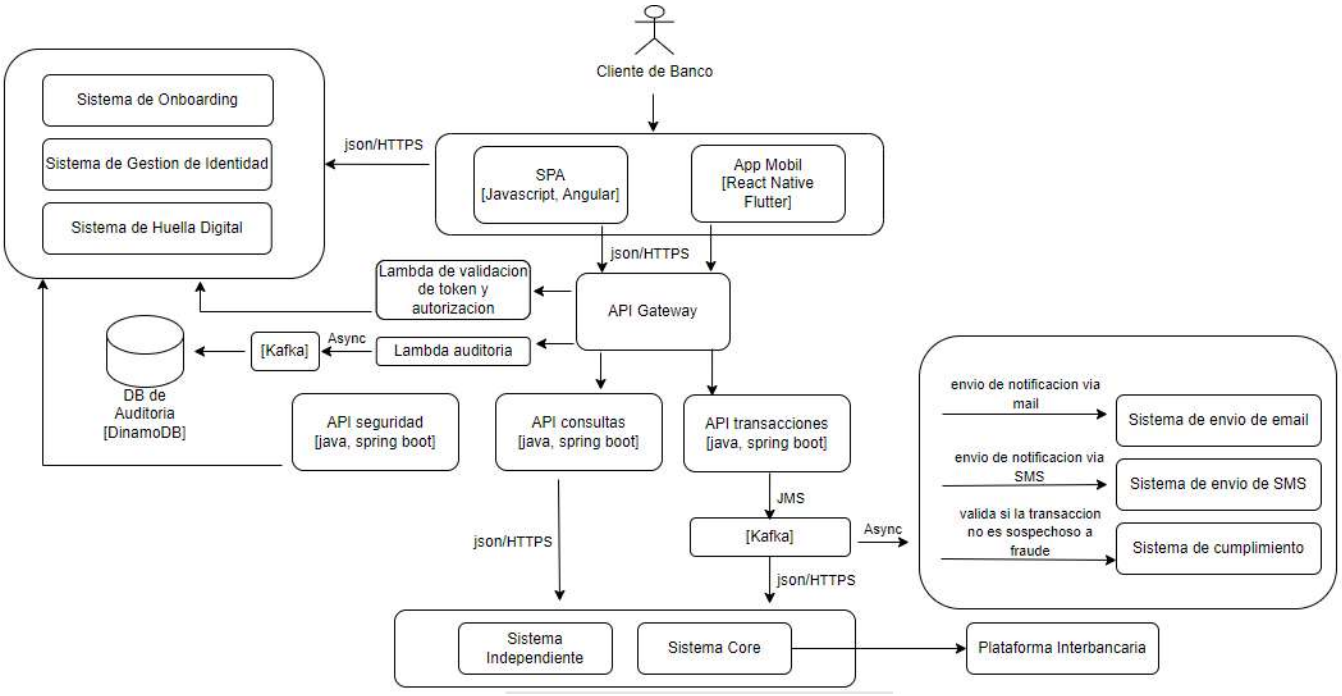
Las transacciones realizadas serán de forma asíncrona hacia el sistema core.



3.- Diagrama de Componentes

Para el filtro de validación de autentificación y autorización se usará un componente lambda e igualmente para el registro de la auditoria que será de forma asíncrona a través de una plataforma de eventos Kafka.

Se contará con 3 microservicios, una que será de dominio de seguridad, el otro microservicio es de consultas del Sistema Core y del Sistema Independiente y el otro microservicio de transacciones que realizara el cliente y se apoyara de una plataforma de eventos Kafka.



4.- Diagrama de Infraestructura

Para el despliegue la solución se ha optado por la nube AWS y la comunicación con sus sistemas internos del banco será a través de VPN.

La capa frontend SPA será desplegado en CloudFront y los dns configurado en Route53, para proteger ante ataques se ha dispuesto de AWS WAF antes de ingresar al API Gateway.

Los microservicios serán desplegados en un cluster Kubernetes auto gestionado AWS Fargate, el cual nos garantiza es escalamiento horizontal de los nodos workers los cuales estarán distribuidos en dos zonas de disponibilidad para garantizar resiliencia y alta disponibilidad.

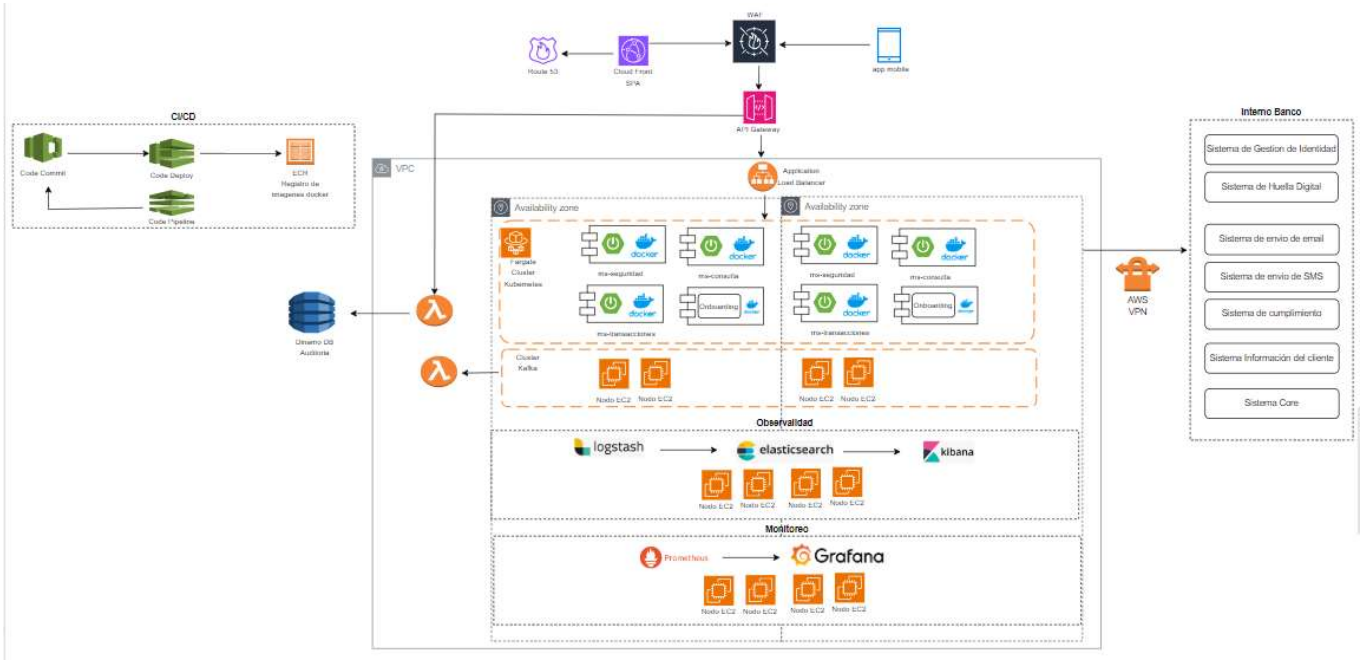
Para la base de datos que guardará datos de auditoria será una base de datos NoSQL DinamoDB que es gestionado y escalada por el propio AWS.

Los dos componentes AWS lambdas serán escritos lenguaje Python por contar una mayor cantidad de librerías y de mejor performance en serverless.

Se instalará un cluster de la plataforma Kafka con los nodos distribuidos en las dos zonas de disponibilidad.

Como herramienta de observabilidad se empleará ELK y como herramienta de monitoreo se usará Prometheus y Grafana.

Para una mejor excelencia operativa la integración continua se realizará a través de los servicios AWS: Code Commit, Code Deploy, Code Pipeline y como registro de imágenes docker AWS ECR.


WAF
Route 53
Cloud Front
SPA
app mobile
API Gateway
CI/CD
Code Commit
Code Deploy
ECR
Registro de
imagenes docker
Code Pipeline
VPC
Application
Load Balancer
Availability zone
Availability zone
Fargate
Cluster
Kubernetes
docker
ms-seguridad
ms-consulta
ms-transacciones
Onboarding
Dinamo DB
Auditoria
Cluster
Kafka
Nodo EC2
Observalidad
logstash
elasticsearch
kibana
Monitoreo
Prometheus
Grafana
AWS
VPN
Interno Banco
Sistema de Gestion de Identidad
Sistema de Huella Digital
Sistema de envio de email
Sistema de envio de SMS
Sistema de cumplimiento
Sistema Información del cliente
Sistema Core