# Prueba Técnica

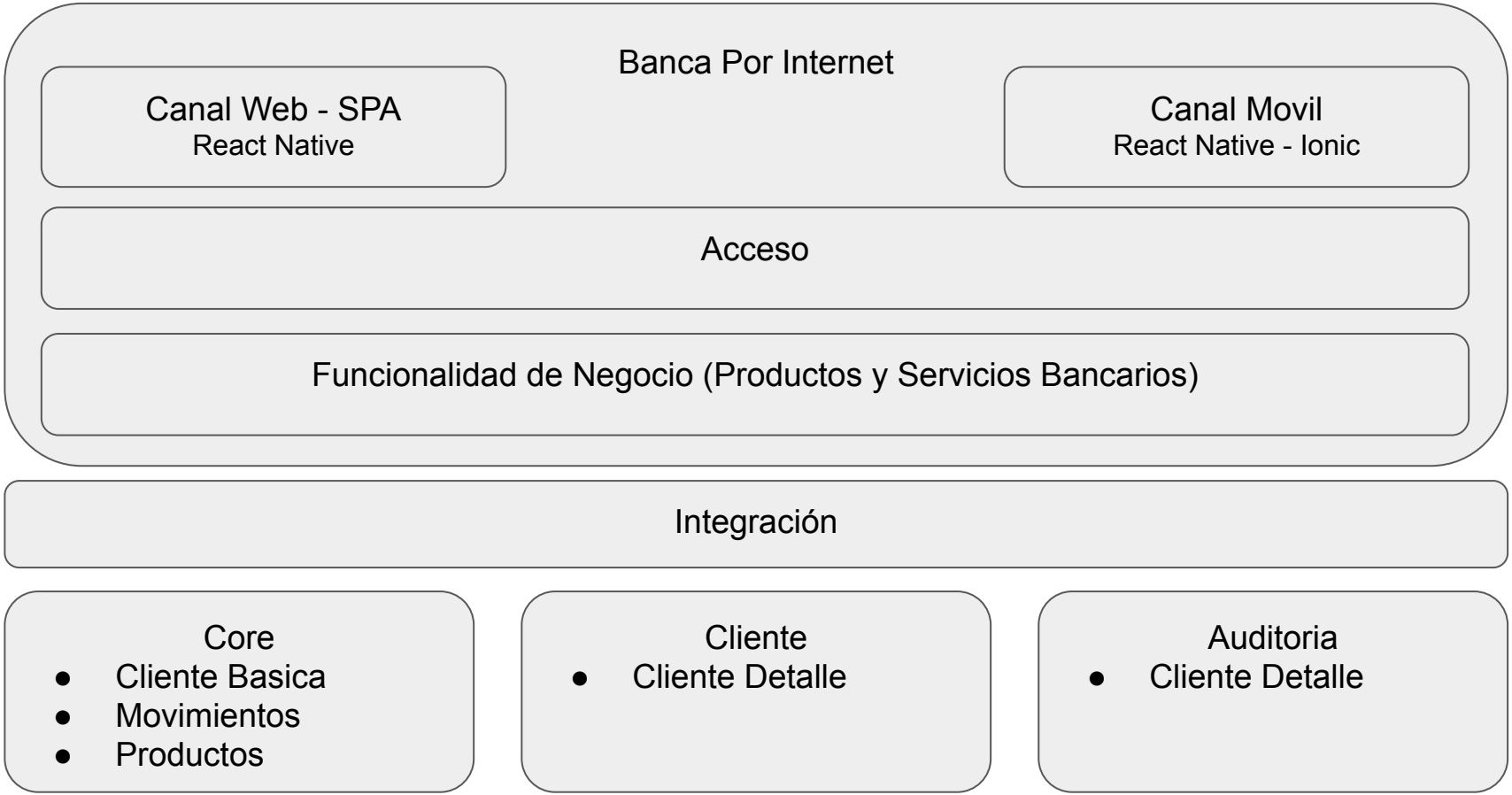
Carlos Arturo Quiroga

# Intruducción

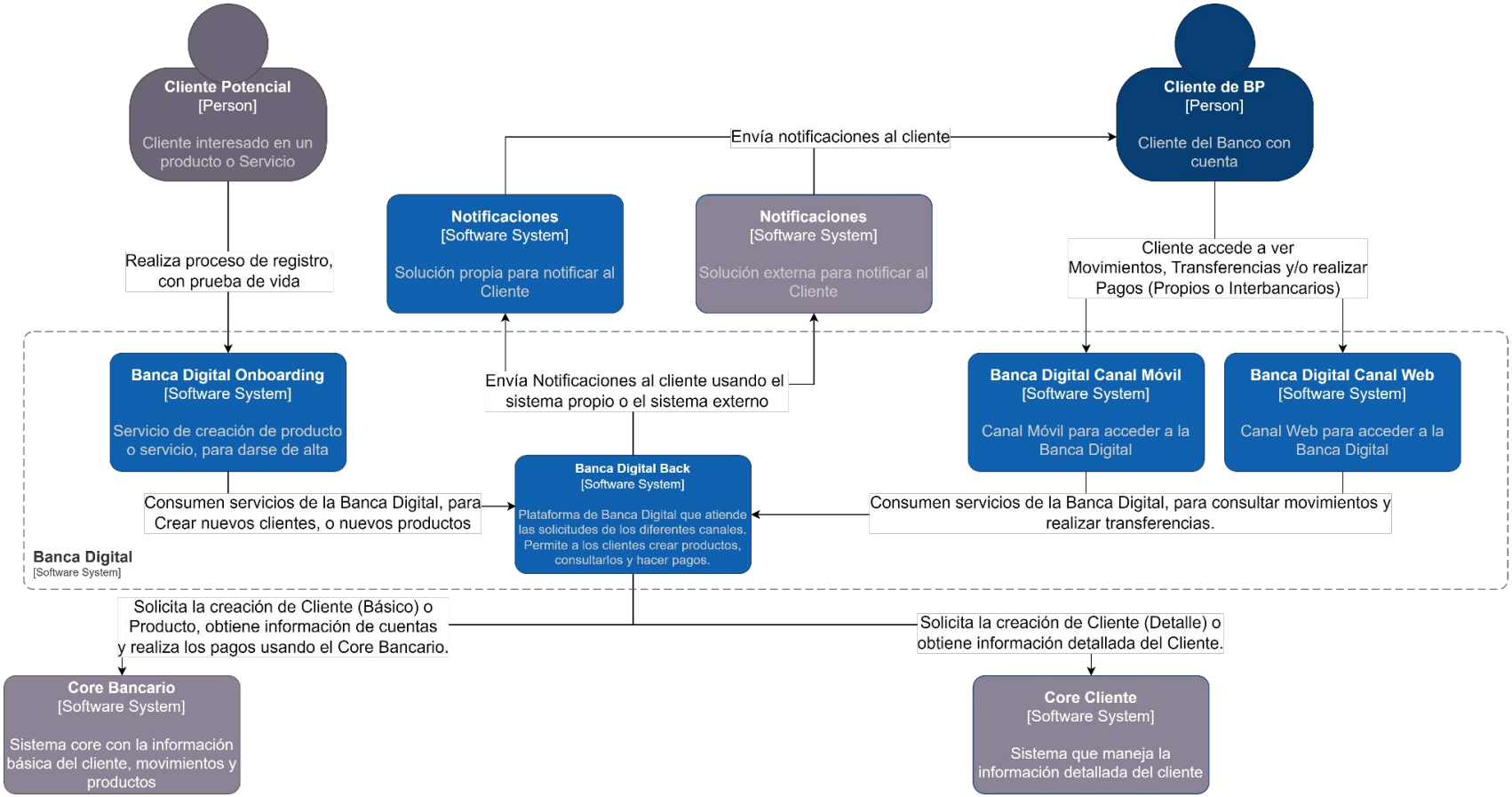
Para el análisis de la solución se crearon diagramas de entendimiento, sobre el contexto del problema, para luego poder generar los diagramas del modelo C4.

Se muestran los dos diagramas, ya que los primeros permiten tener una comunicación con las áreas de negocio y siguientes sirven para hacer la traducción al lenguaje técnico.

# Análisis de Contexto - Diagrama de Capas

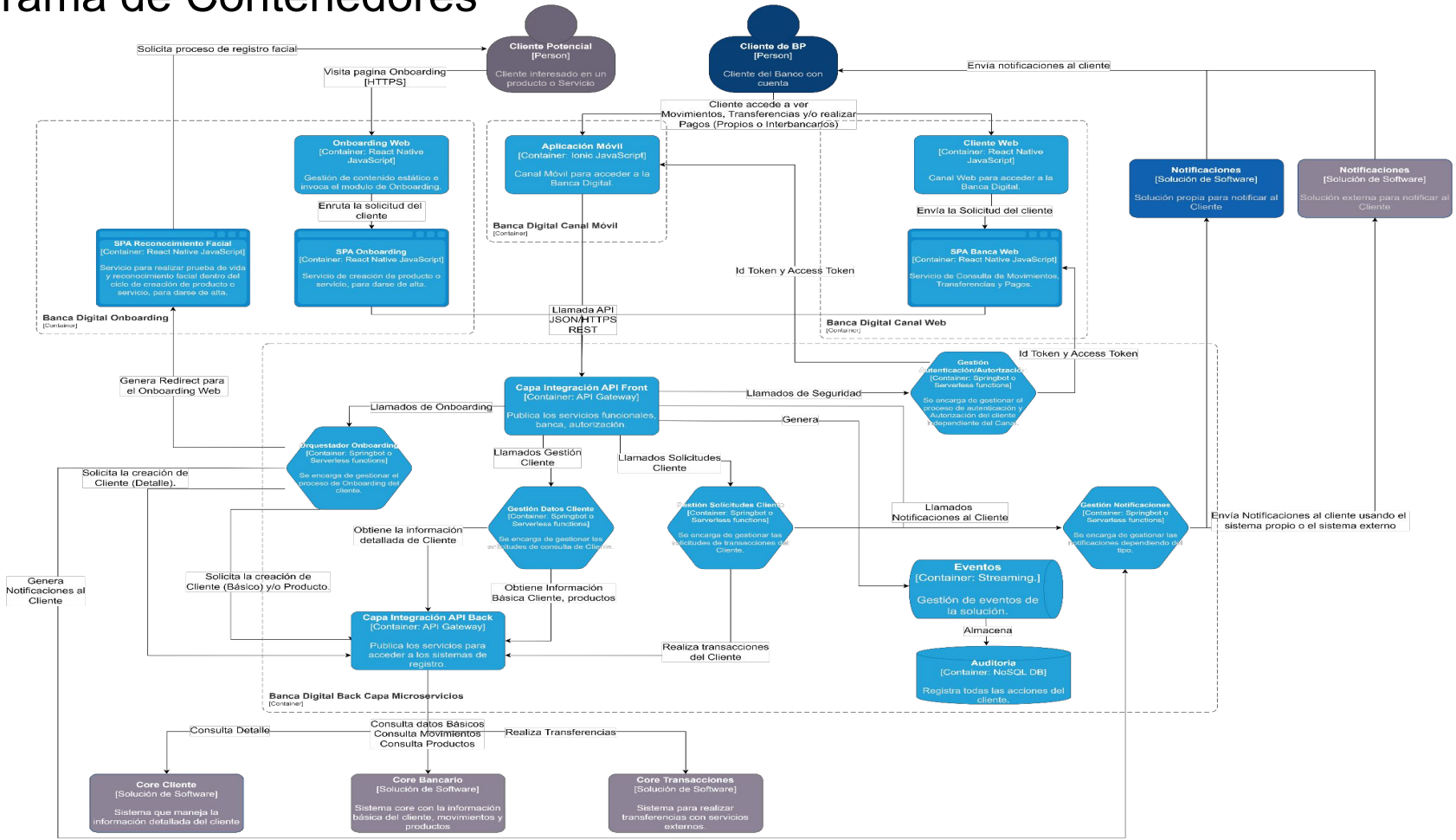


# Diagrama de Contexto

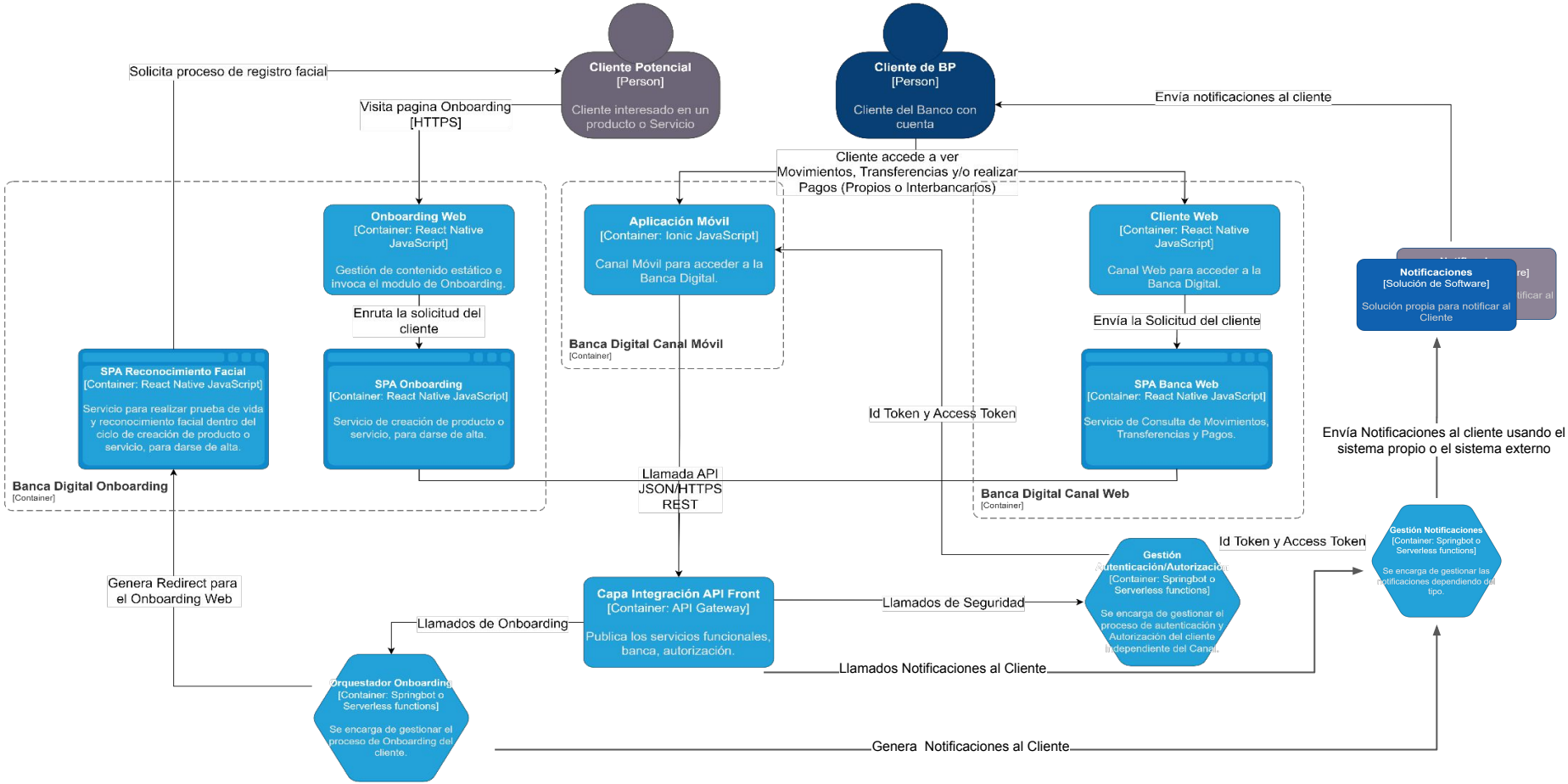


# Análisis de Contenedores - Diagrama de Capas

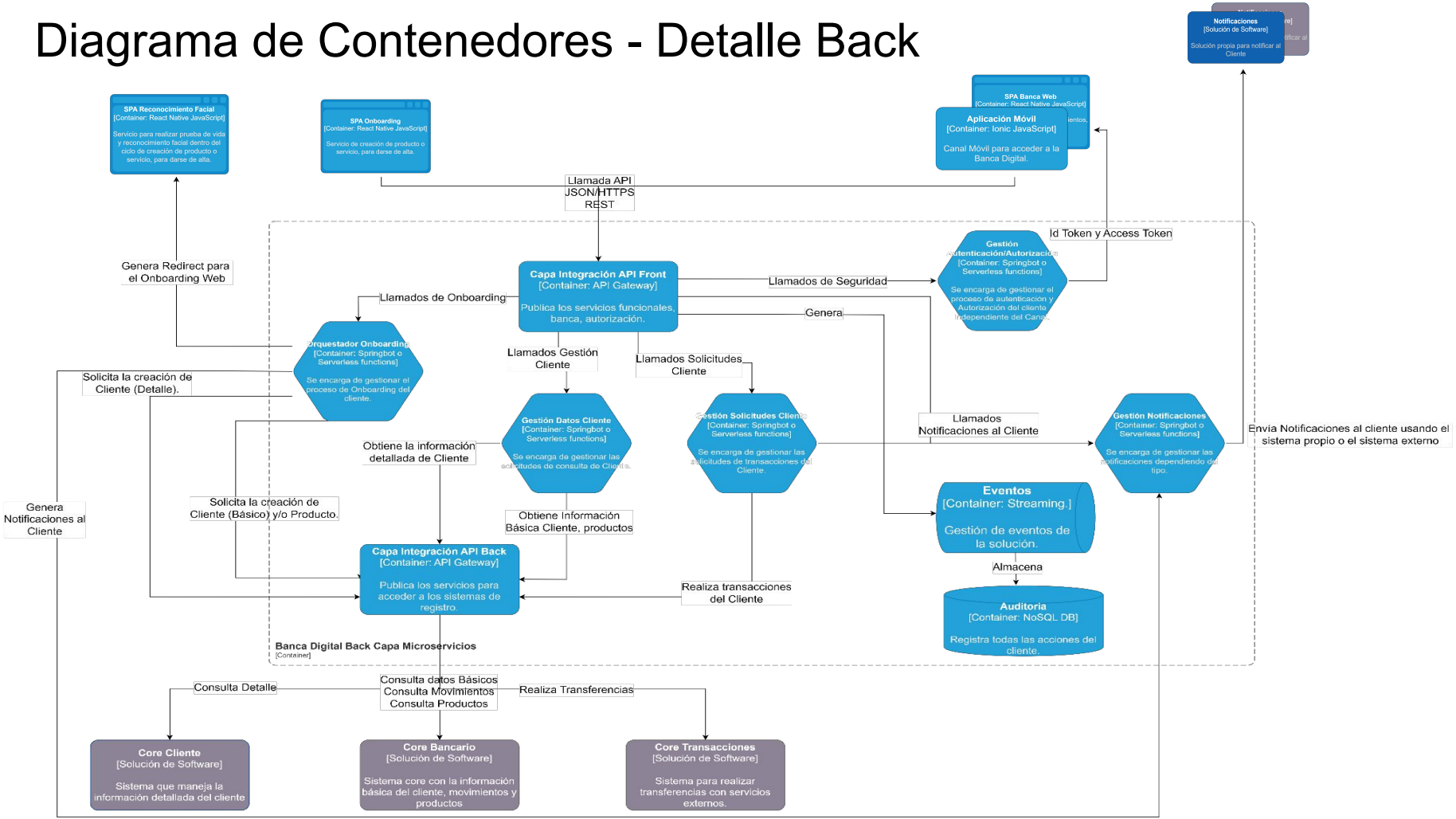
# Diagrama de Contenedores



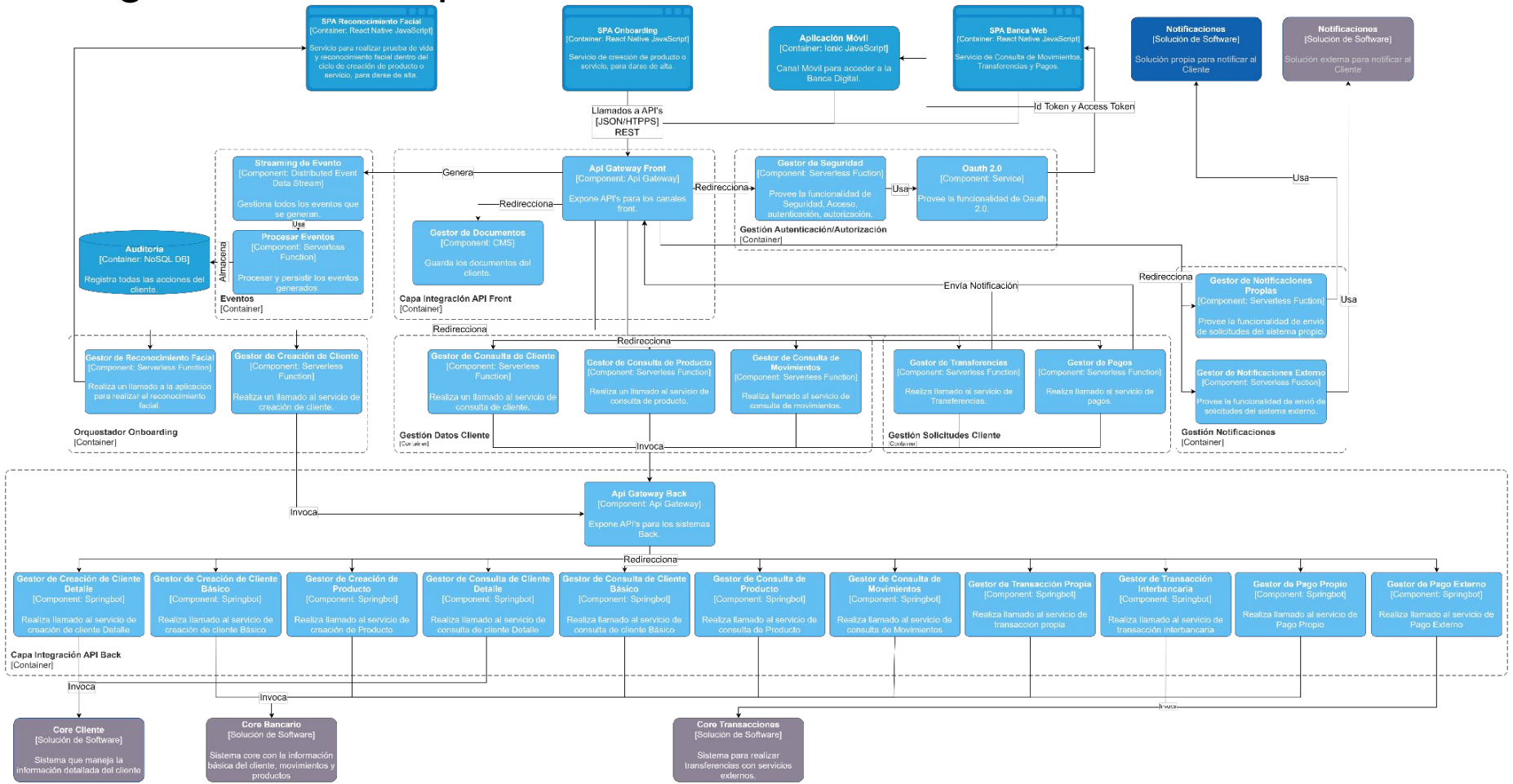
# Diagrama de Contenedores - Detalle Front



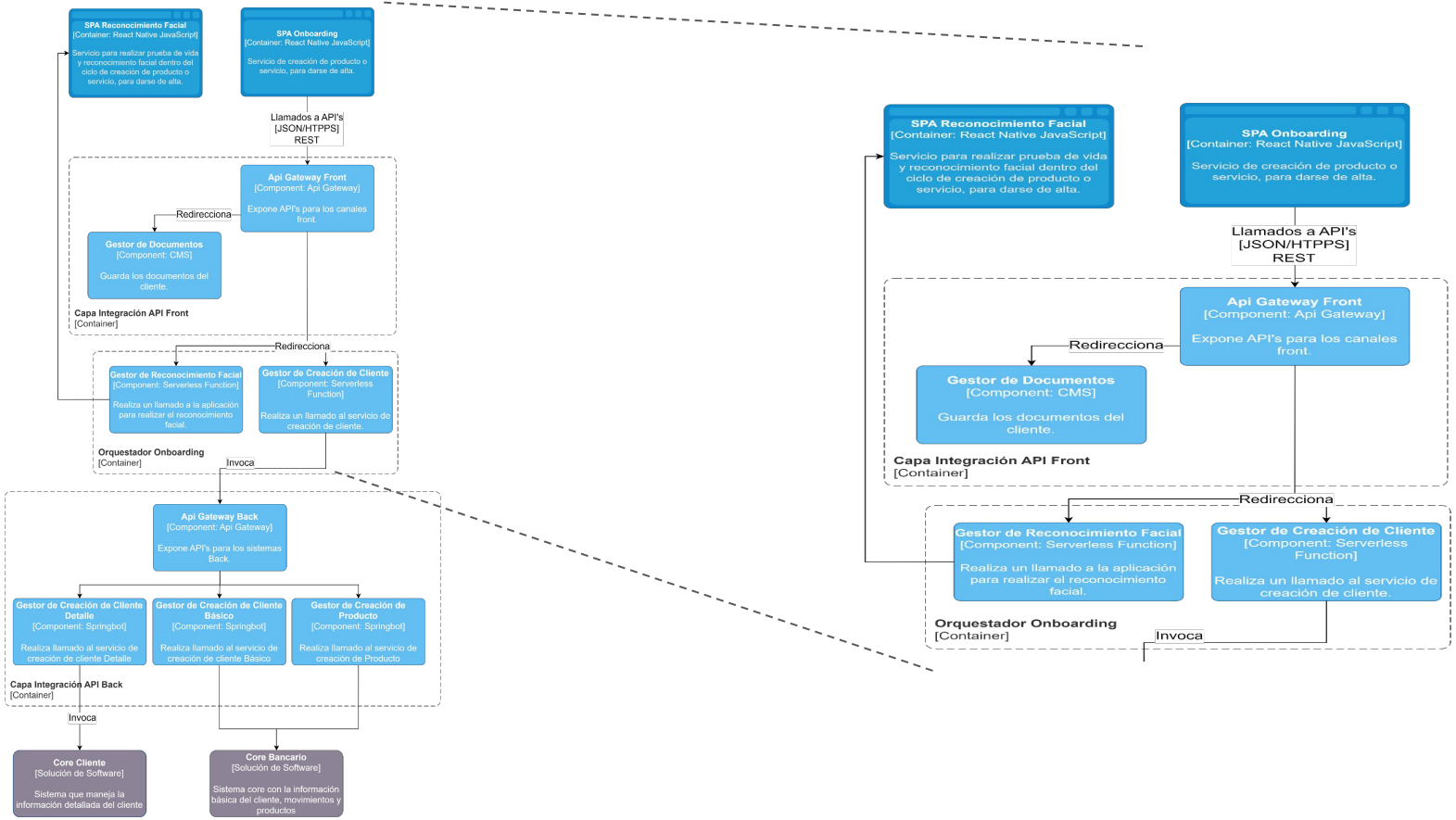
# Diagrama de Contenedores - Detalle Back



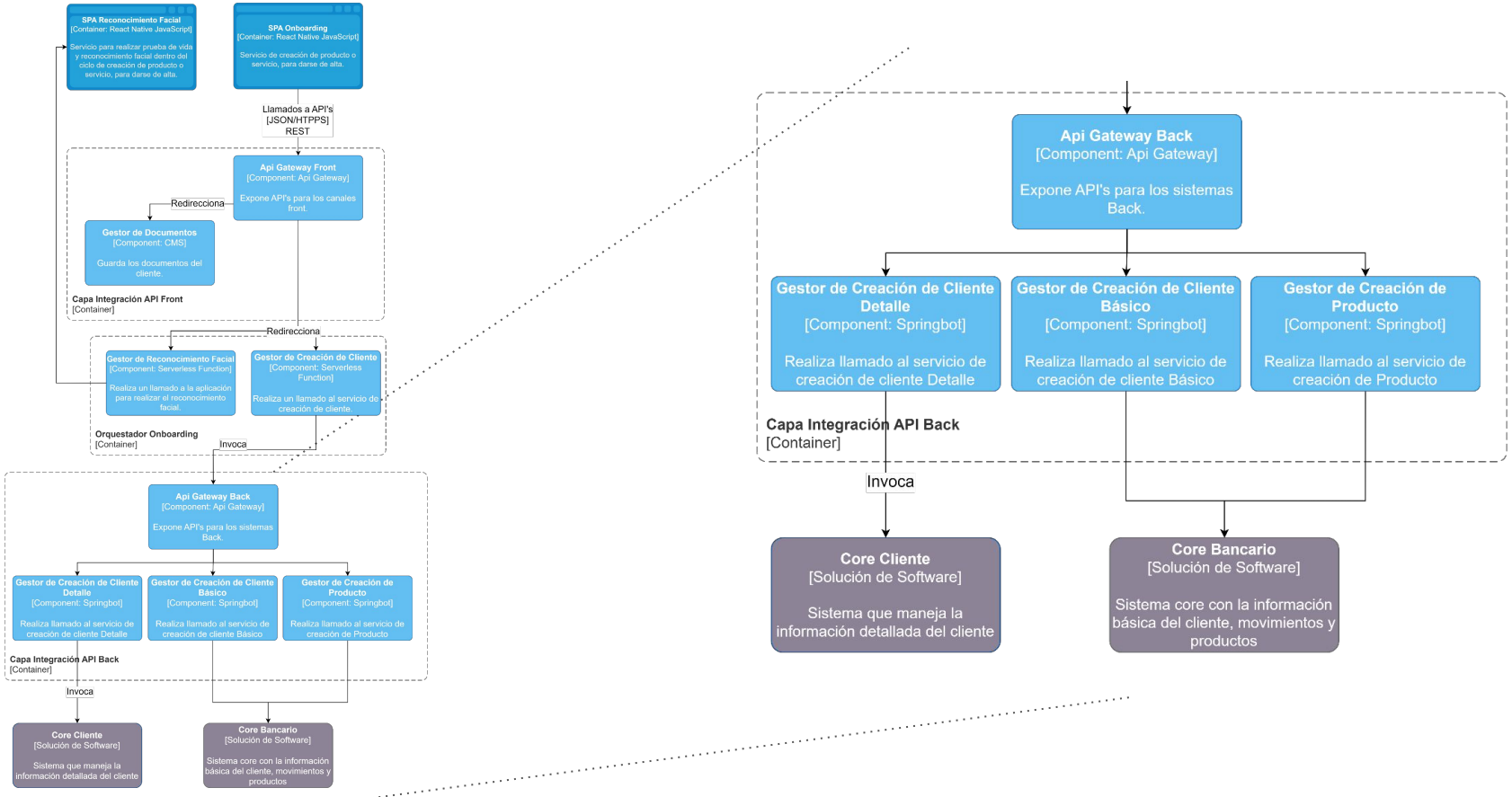
# Diagrama de Componentes



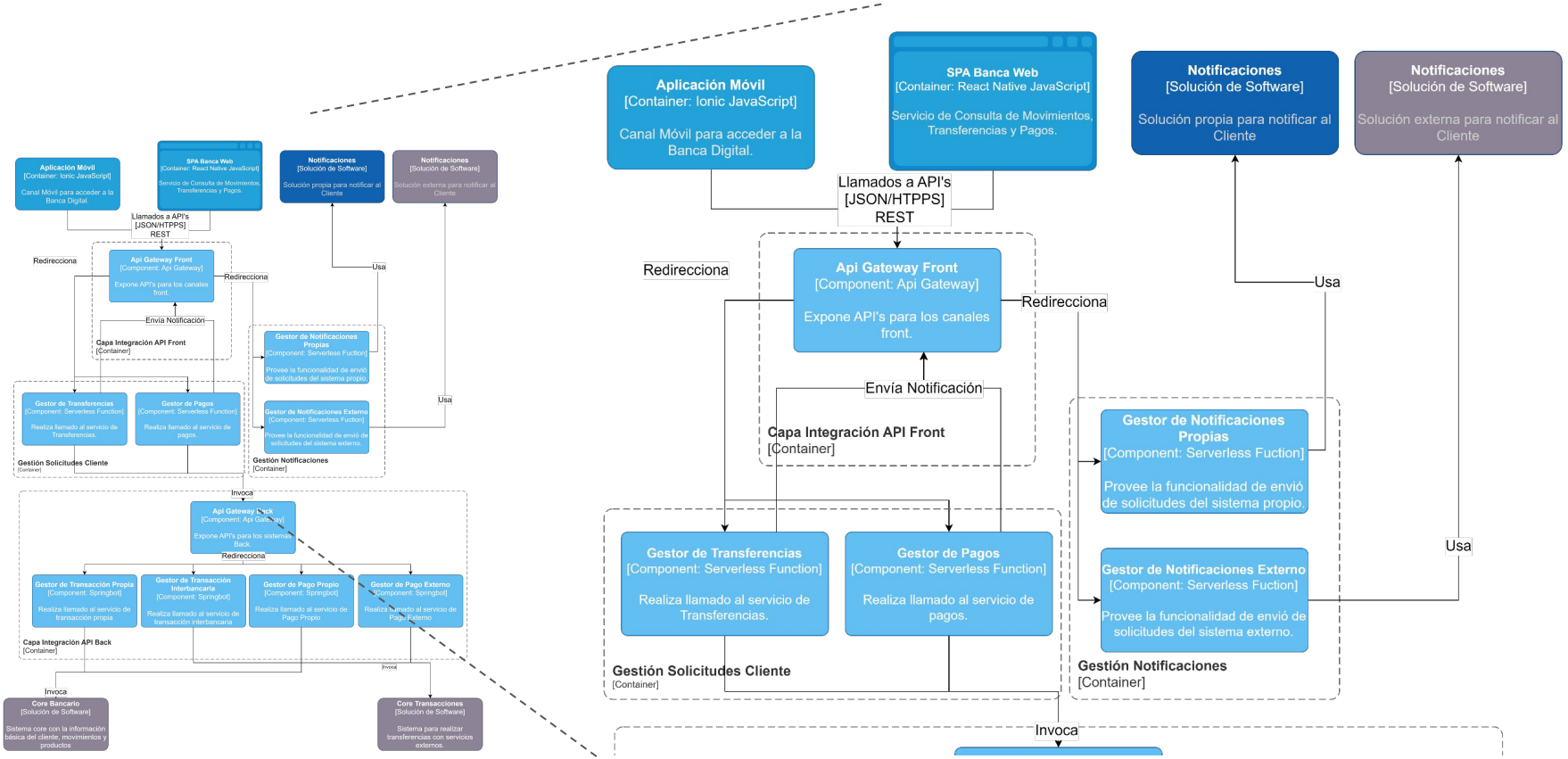
# Diagrama de Componentes - Detalle Onboarding Front



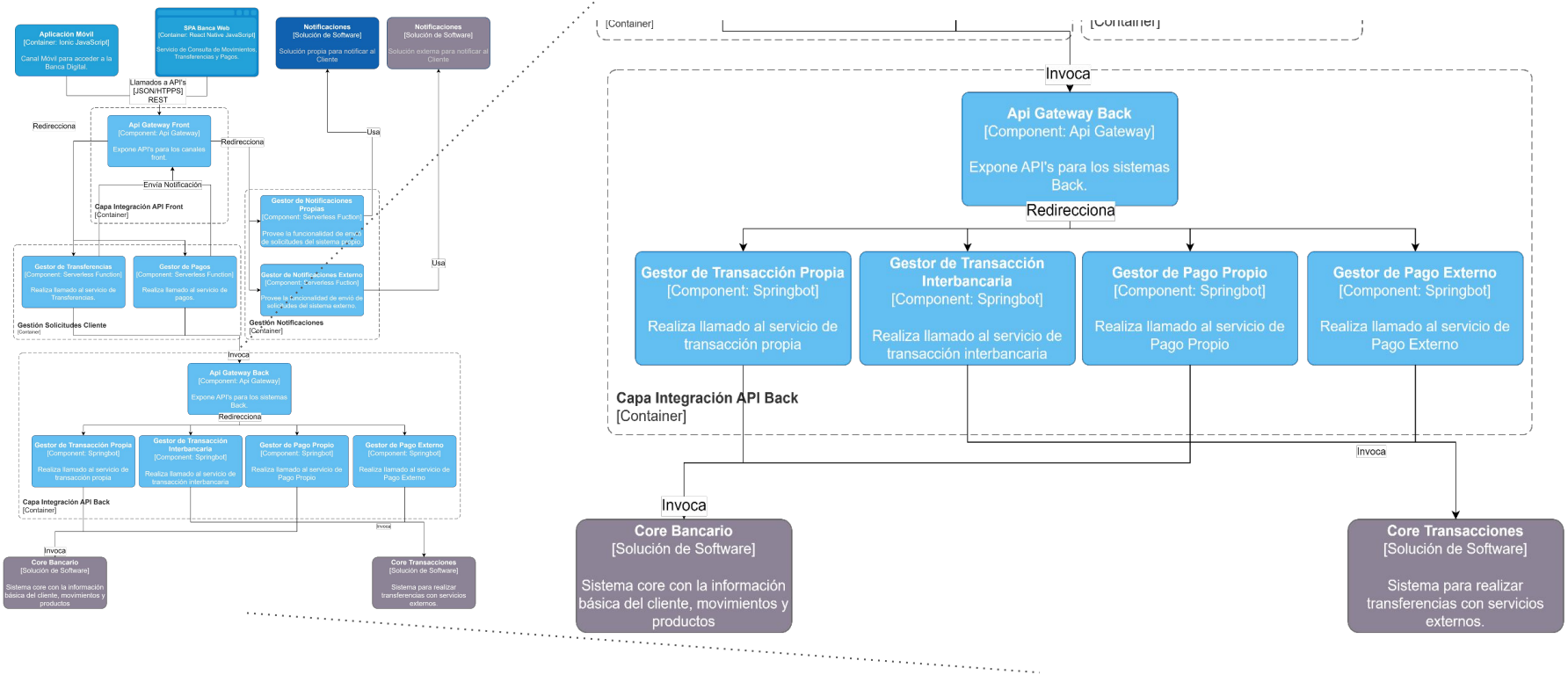
# Diagrama de Componentes - Detalle Onboarding Back



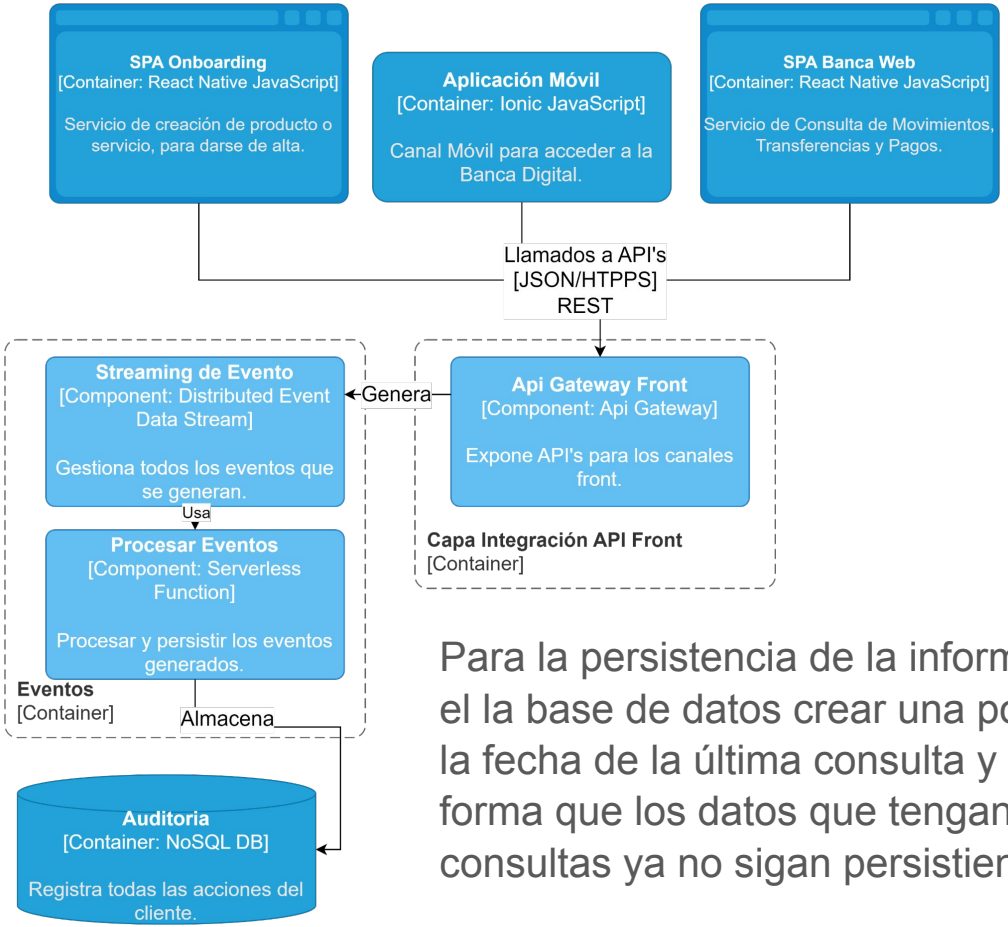
# Diagrama de Componentes - Detalle Notificaciones Front



# Diagrama de Componentes - Detalle Notificaciones Back



# Diagrama de Componentes - Detalle Auditoría




Para la solución de Auditoría se parte de la premisa que todos las solicitudes, incluso las de notificaciones pasan por el API Gateway de la capa Front, partiendo de una arquitectura de eventos, cada transacción genera un evento de streaming de datos, el cual es gestionado en una función Serverless para guardarlo en la base de datos.

Para la persistencia de la información de los clientes frecuentes se propone el la base de datos crear una política de ciclo de vida de los datos basada en la fecha de la última consulta y el número de consultas realizadas, de tal forma que los datos que tengan una fecha lejana y un número máximo de consultas ya no sigan persistiendo.

# Oauth 2.0 - Recomendaciones Flujo de Autenticación

Dado que se cuenta con un producto para configurar Oauth 2.0 y que se va a utilizar para la autenticación desde los dos canales, la aplicación SPA y la aplicación móvil, se recomienda tener un único ciclo de autorización.

En este caso partiendo de los dos Frameworks propuestos (React Native e Ionic) que utilizan JavaScript, se recomienda utilizar el flujo de autorización con Clave de Prueba para Intercambio de Código PKCE (Proof Key for Code Exchange), porque el token de acceso no está expuesto en el cliente y además este flujo permite devolver tokens de actualización para proporcionar validaciones que no generen o no necesiten interacción con el usuario.

Además se recomienda al equipo de implementación revisar el SDK para SPA de Oauth, el cual proporciona un API para implementar dicho flujo.

# Elementos Normativos

Además de la ley de protección de datos, considero que se deben tener en cuenta los siguientes puntos:

- Validar si el banco está inmerso en los procesos de Basilea 4, para análisis del riesgo crediticio, aunque en el ejercicio no se ofrecen productos o servicios de crédito, es bueno tener en cuenta esta normatividad pensando en que la arquitectura se adapte a nuevos requerimientos.
- Identificar la regulación asociada a transparencia de datos o modelos de datos abierto.
- Revisar si el banco ha implementado políticas de gestión de activos por ponderación de riesgo, teniendo en cuenta la normatividad emitida por la Superintendia de Bancos y las recomendaciones de Asobanca (https://asobanca.org.ec/wp-content/uploads/2021/09/Informe-Te%CC%81cnico-Esta%CC%81ndares-Regulatorios-Financieros-Internacionales-Oct-2019.pdf)
- Incorporar las definiciones asociadas a inclusión financiera.
- Incorporar elementos tecnológicos para evitar la filtración de datos, como modelos de redes privadas, análisis de logs de las transacciones aprovechando la solución de auditoría a implementar.

# Recomendaciones de despliegue

- Utilización de funciones serverless para asegurar pago por uso y alta disponibilidad.
- Utilización de servicios en nube, para gestión de streams para auditoría y una base de datos NoSql serverless, es decir gestionada por el proveedor de nube.
- Para la página web utilizar un servicio administrado de aplicaciones web (CloudFormation Azure Resource Manager).
- Implementar patrones de caché para dados frecuente.
- Implementar servicios onEdge que proveen los proveedores de nube.
- Implementar protocolos seguros en todas las comunicaciones TLS1.2 y TLS1.3.
- Gestionar todos los accesos a los recursos de nube con políticas, teniendo en cuenta la regla de mínimo privilegio.

# Enlace repositorio GitHub

https://github.com/carlosaq/devsupruebacaqq