

MODELO C4

El **modelo C4** es un enfoque visual para documentar y comunicar **la arquitectura de software** de forma clara y estructurada.

Un buen diagrama de arquitectura de software facilita la comunicación a los equipos de desarrollo y producto, la idea principal es que los diagramas puedan ser entendidos e interpretados por equipos técnicos y no técnicos.

Cuenta con **cuatro niveles de abstracción**, de ahí su nombre:

- Nivel 1: Diagrama de **C**ontexto.
- Nivel 2: Diagrama de **C**ontenedores
- Nivel 3: Diagrama de **C**omponentes
- Nivel 4: Diagrama de **C**ódigo (opcional)

Estos permiten mejorar la documentación de proyectos completos a fin de que la audiencia pueda comprender en que consiste, como esta conformado y como se interrelaciona con otros sistemas o servicios.

Ejercicio Práctico.

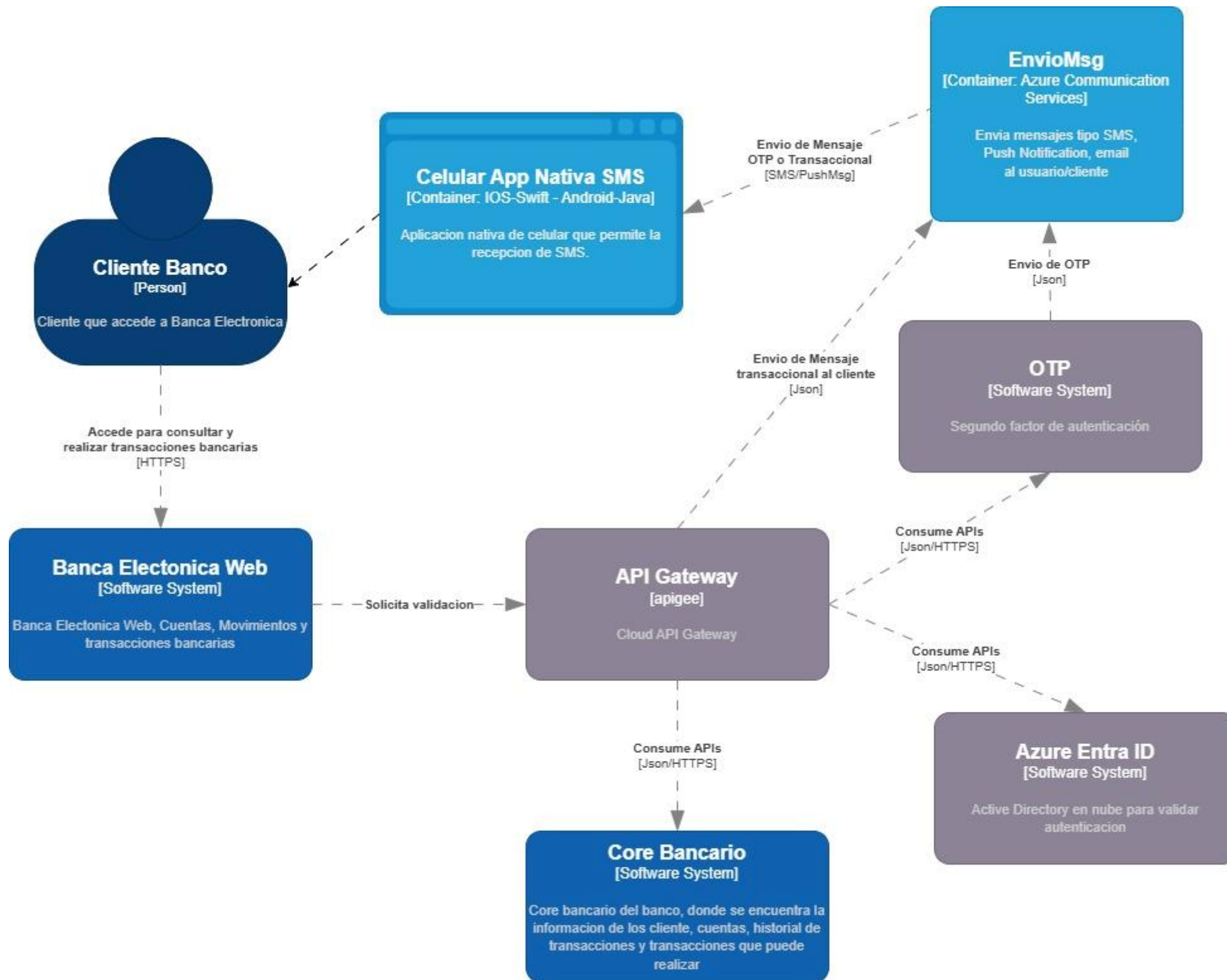
Realizaremos un Análisis y Diagramas de un Sistema de Banca Electrónica Web de un banco X, considerando los siguientes aspectos:

1. Solo puede ser accedido por clientes del banco
2. Cuenta con un esquema de autenticación y autorización (Azure Entra ID)
3. Cuenta con un modelo de factor de autenticación mediante OTP (Contraseña de un Solo Uso) que se enviará al teléfono celular del usuario como SMS o Push Notification, o email
4. La OTP tiene un tiempo de vigencia de 3 minutos, caso contrario caduca y deniega el acceso
5. Valida que el usuario se encuentre activo en el banco
6. El sistema permite consultar Posición Consolidada y Estados de Cuenta
7. Permite consultar y actualizar cupos para transacciones y pagos de las cuentas de clientes
8. Permite realizar transferencias bancarias e interbancarias
9. Permite realizar pagos de servicios con debito a cuenta
10. Envía un mensaje al usuario (cliente) como SMS o Push Notification, o email, cuando se ha realizado una transacción bancaria
11. Permite guardar los logs de sistema y transaccionales en un repositorio como Elasticsearch.

Nivel 1: Diagrama de Contexto.

Muestra el nivel más general del sistema a analizar, los usuarios que interactúan con el sistema, los sistemas externos con los que se relaciona, y las interacciones que tiene cada elemento con el sistema.

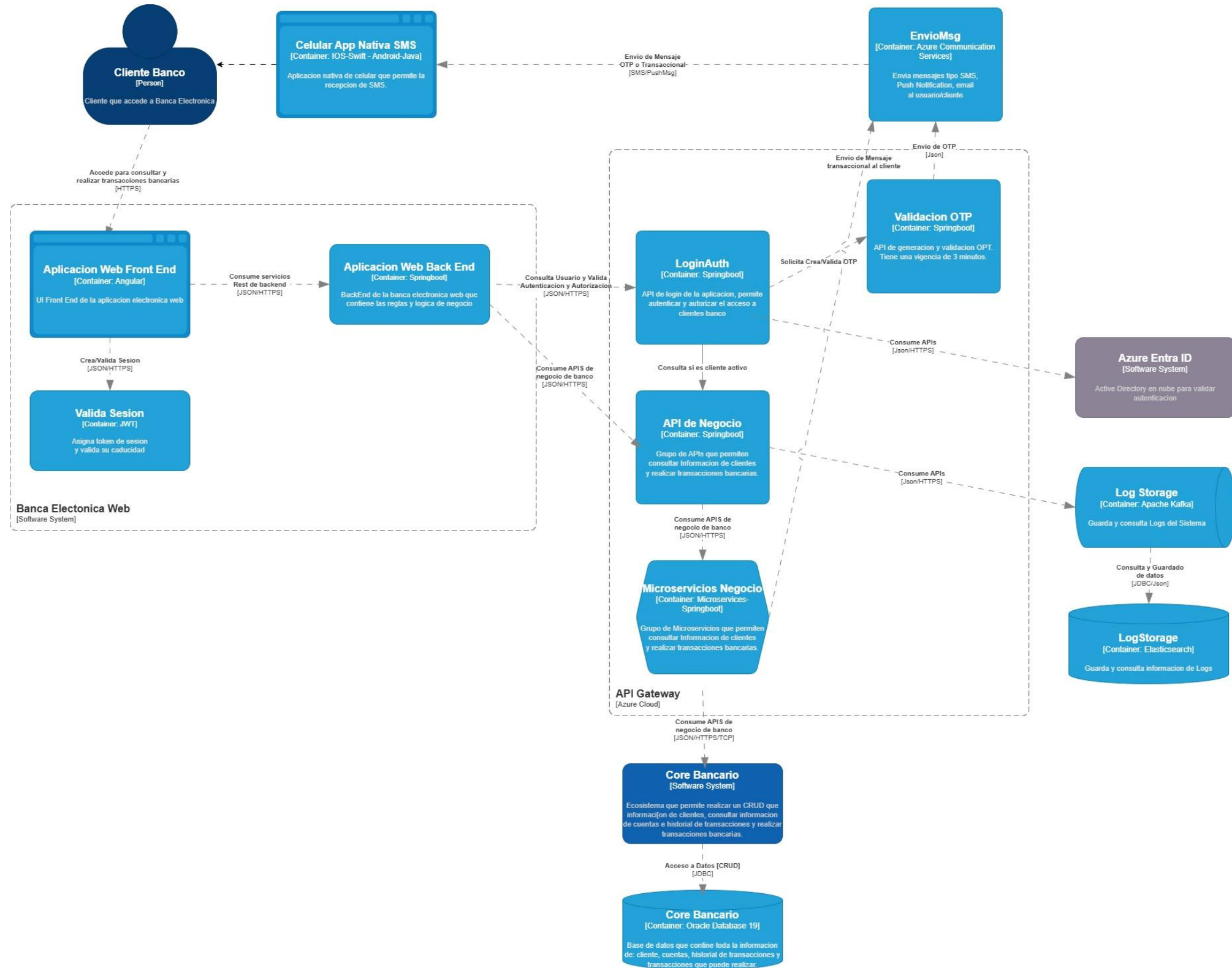
Para nuestro caso exponemos el Diagrama de Contexto de la Banca Electrónica Web, donde se indica que el sistema puede ser usado por el Cliente Banco, se integra con un API Gateway para realizar llamados a APIs:



Nivel 2: Diagrama de Contenedores.

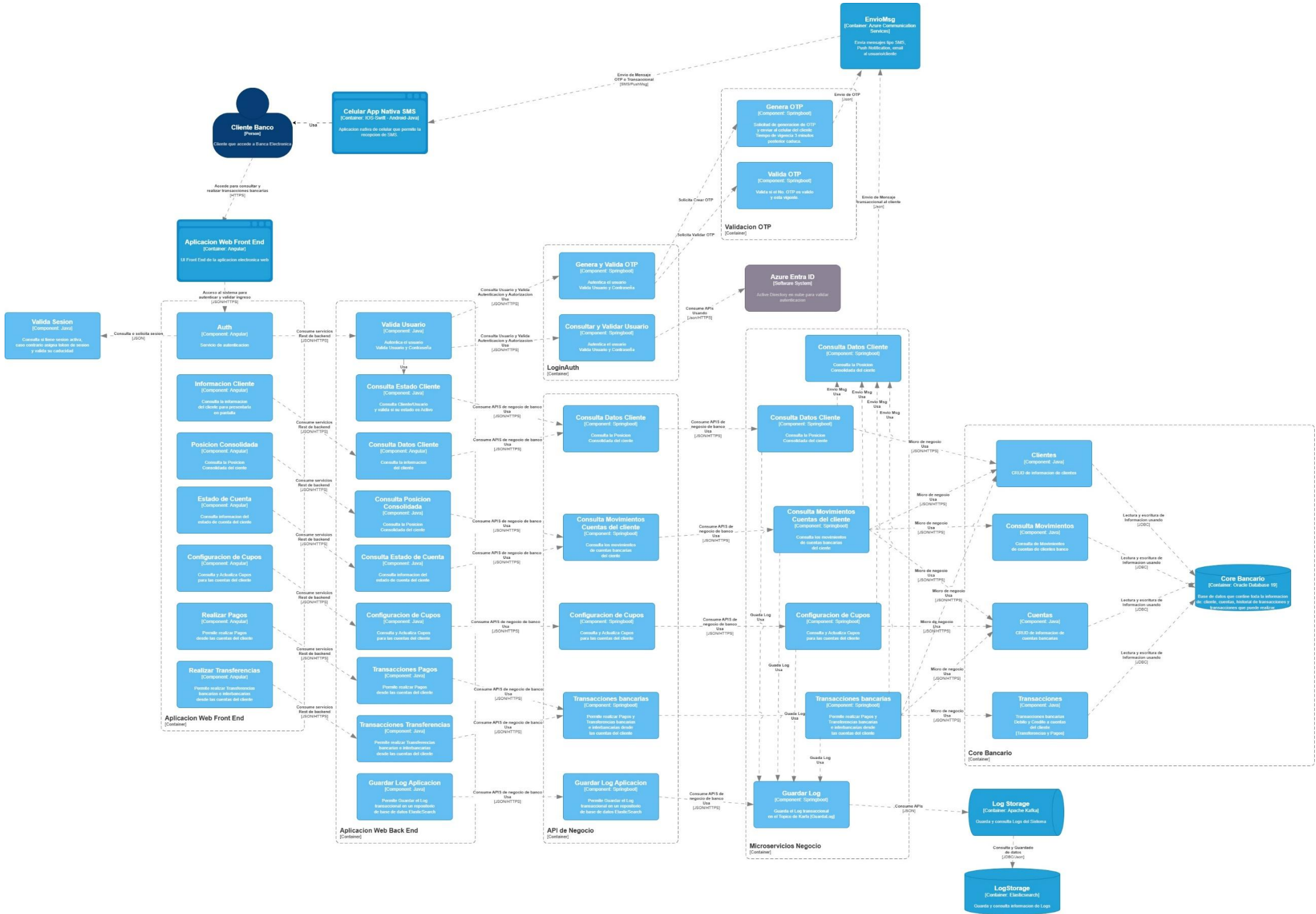
Muestra la estructura del Sistema en un alto nivel. Descompone el sistema en aplicaciones, servicios, bases de datos, etc. Ayuda a entender cómo se distribuyen las responsabilidades y las interacciones que tiene cada elemento con el sistema.

Se muestra los contenedores y servicios que utiliza el sistema y la interrelación con las otras capas.



Nivel 3: Diagrama de Componentes. Profundiza en cada contenedor o servicio, para mostrar los componentes internos de cada uno y cómo se relacionan entre sí. Visualiza la lógica de negocio y funciones específicas.

Se muestra un detalle de los componentes que utiliza el sistema, así como sus APIs y servicios, componentes en nube y el Core Bancario.



Nivel 4: Diagrama de Código (opcional). Representa detalles técnicos como clases, métodos, interfaces o módulos, que son los componentes más pequeños a nivel de programación. Útil para desarrolladores que necesitan comprender la implementación y lógica de cada componente

Se indica un diagrama muy general de clases relacionado al Sistema de Banco.

