Arquitectura de Solución: Sistema de Banca por Internet para BP

1. Introducción

- **Descripción del proyecto**: El objetivo de este documento es presentar la arquitectura de solución para el sistema de banca por internet de la entidad BP. El sistema permitirá a los usuarios acceder al histórico de sus movimientos, realizar transferencias y pagos entre cuentas propias e interbancarias.
- **Alcance**: Funcionalidades principales (histórico de movimientos, transferencias, pagos, notificaciones).

2. Requisitos Normativos y de Seguridad

- **Normativa Local (Ecuador)**:
    - **Ley Orgánica de Protección de Datos Personales (LOPDP).** Garantiza la protección y manejo adecuado de los datos personales de los usuario
    - **Normativa sobre la Seguridad de la Información**: Emitida por la Superintendencia de Bancos, esta normativa requiere que las instituciones financieras implementen un **Sistema de Gestión de Seguridad de la Información (SGSI)**
    - **Superintendencia de Bancos del Ecuador (SBE)**: La SBE establece regulaciones para la operación de instituciones financieras. Es importante asegurarse de cumplir con los requerimientos en términos de auditorías, privacidad de datos y políticas de seguridad.
- **Normativa Internacional**:
    - **ISO 27001 (Seguridad de la Información).** Norma internacional para la gestión de la seguridad de la información.
    - **PCI DSS (Seguridad para datos de tarjetas**). Estándar de seguridad de datos para la protección de información sensible relacionada con pagos.
    - **NIST Cybersecurity Framework**: Proporciona un enfoque estructurado para gestionar riesgos de ciberseguridad mediante cinco funciones clave: identificar, proteger, detectar, responder y recuperar.
    - **OWASP (Open Web Application Security Project)**: Utilizar las guías de OWASP es esencial para proteger las aplicaciones web bancarias de vulnerabilidades comunes, como inyecciones SQL
- **Recomendaciones de seguridad**:
    - Implementación de OAuth 2.0. para la autenticación y autorización de usuarios, con el flujo Authorization Code Flow con PKCE
    - Autenticación multifactor (MFA) para fortalecer la seguridad del acceso(integración nativa de Azure Active Directory (Azure AD) con OAuth 2.0).

- Onboarding con reconocimiento facial (Azure Face API).

3. Arquitectura C4

3.1. Modelo de Contexto (Nivel 1)

- **Usuarios finales**: Clientes que usan la banca en línea y la aplicación móvil.
- **Sistemas**:
  - **Plataforma Core**: Gestiona la información básica de clientes y sus productos.
  - **Sistema complementario**: Proporciona detalles adicionales sobre los clientes.
  - **Sistemas de notificación**: Envía mensajes SMS y correos electrónicos.
- **Aplicaciones**:
  - **SPA (Single Page Application)**: Para la interfaz web.(SPA Aplicación web que carga una única página HTML y actualiza dinámicamente el contenido a medida que el usuario interactúa con la página, sin recargar por completo el navegador. Esto mejora la experiencia del usuario con tiempos de respuesta más rápidos y una interacción más fluida.)
  - **Aplicación móvil**: Basada en frameworks multiplataforma, como **Flutter** o **React Native**, que son seguras y ampliamente utilizadas.

    Flutter

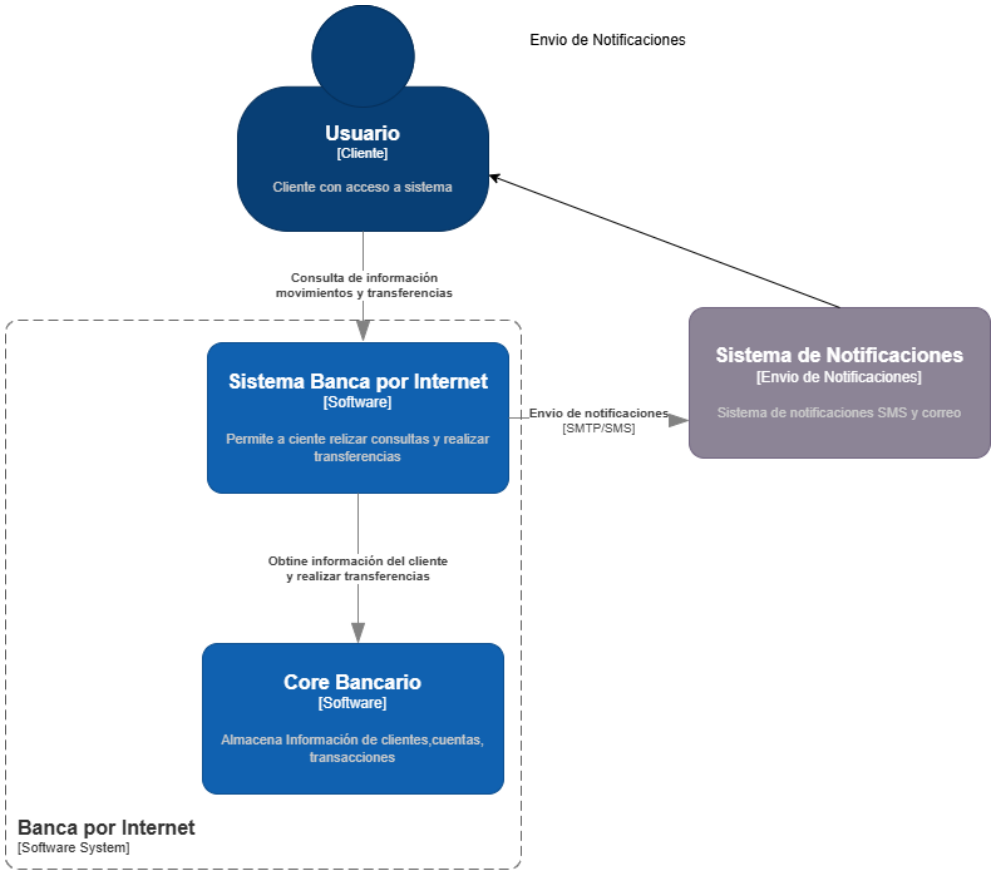
    - Multiplataforma: Un solo código base para aplicaciones en iOS, Android, web y escritorio, reduciendo tiempo y costos de desarrollo.
    - Alto rendimiento: Compila a código nativo, lo que garantiza una experiencia rápida y fluida en dispositivos móviles.
    - Interfaz personalizable: Permite crear interfaces atractivas y dinámicas con widgets flexibles y personalizables.
    - Actualizaciones instantáneas: Hot reload facilita el desarrollo ágil y permite ver los cambios de código de inmediato.
    - Gran comunidad y soporte: Respaldado por Google, cuenta con un amplio ecosistema de plugins y herramientas para desarrolladores.

    React Native

    - Multiplataforma: Permite desarrollar aplicaciones nativas para iOS y Android con una sola base de código en JavaScript.
    - Componentes nativos: Utiliza componentes nativos del sistema operativo, lo que garantiza un rendimiento y experiencia cercanos a las aplicaciones nativas.
    - Gran comunidad: Es ampliamente popular y cuenta con una gran cantidad de bibliotecas y recursos comunitarios.
    - Hot Reload: Permite a los desarrolladores ver instantáneamente los cambios en el código sin recompilar la app completa.

        - Fácil integración: Se integra bien con soluciones de terceros y sistemas preexistentes, facilitando el desarrollo de aplicaciones robustas.

- **Servicios externos**:
    - Notificaciones SMS y correo. AWS (Amazon SNS para SMS y Amazon SES para correo)
    - Servicios de autenticación OAuth 2.0. (Authorization Code Flow con PKCE)
    - Onboarding de nuevos usuarios con reconocimiento facial (Azure Face API).



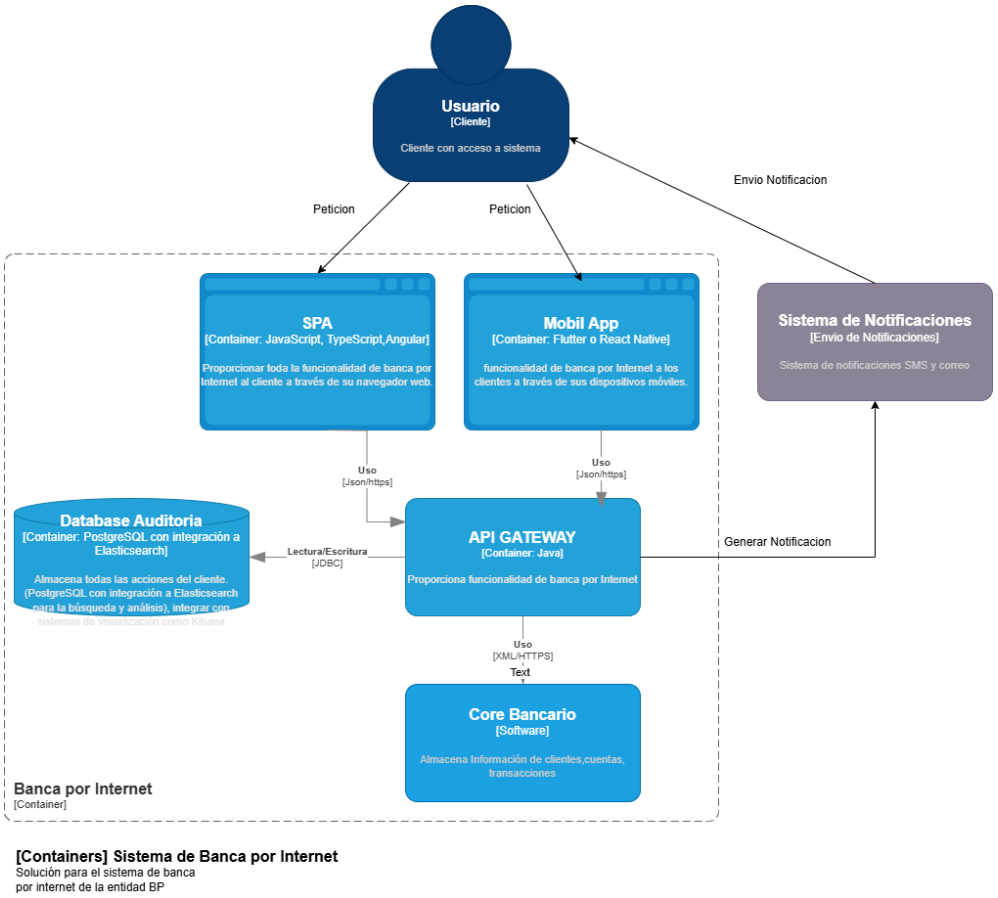
[Diagrama de Contexto] Sistema de Banca por Internet
Solución para el sistema de banca
por internet de la entidad BP

3.2. Modelo de Contenedor (Nivel 2)

- **Aplicaciones Frontend**:
    - SPA: Conectada al backend mediante una API Gateway.
    - Aplicación móvil: Realiza la autenticación y operaciones a través de la API Gateway.
- **Backend**:

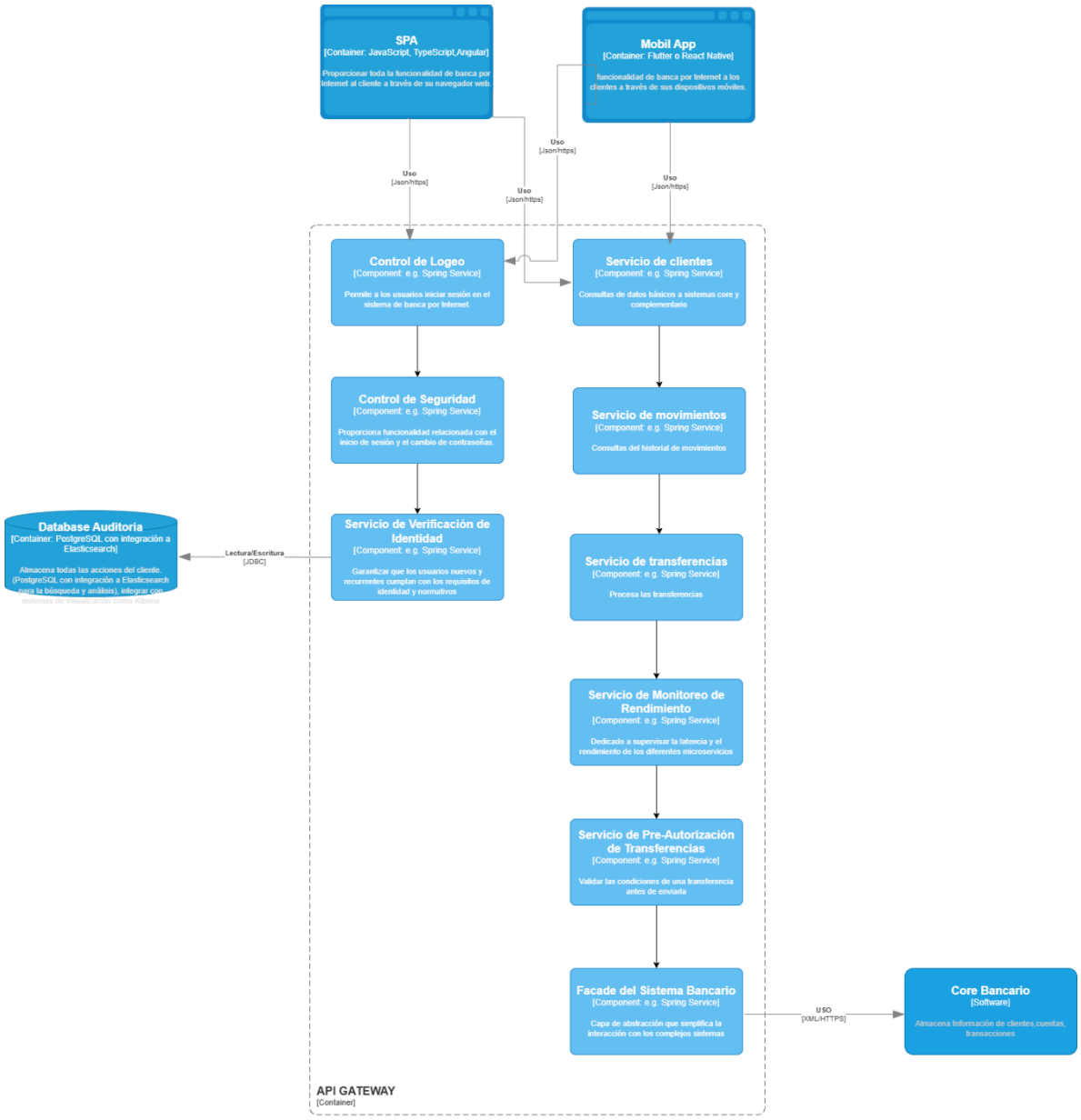
- **API Gateway**: Maneja las solicitudes del frontend y redirige a los microservicios.
- **Microservicios Principales**:
  - Servicio de clientes: Consultas de datos básicos a sistemas core y complementario.
  - Servicio de movimientos: Consultas del historial de movimientos.
  - Servicio de transferencias: Procesa las transferencias.
- **Microservicios Secundarios**:
  - Servicio de Verificación de Identidad: Garantizar que los usuarios nuevos y recurrentes cumplan con los requisitos de identidad y normativos
  - Servicio de Monitoreo de Rendimiento**:** Dedicado a supervisar la latencia y el rendimiento de los diferentes microservicios
  - Servicio de Pre-Autorización de Transferencias: Validar las condiciones de una transferencia antes de enviarla
  - Facade del Sistema: Capa de abstracción que simplifica la interacción con los complejos sistemas
- **Base de datos de auditoría**: Almacena todas las acciones del cliente.(PostgreSQL con integración a Elasticsearch para la búsqueda y análisis), integrar con sistemas de visualización como Kibana
- **Servicios de notificaciones**: Integra con sistemas de notificaciones externas AWS (Amazon SNS para SMS y Amazon SES para correo)

- **Persistencia**:
  - Cache para clientes frecuentes utilizando **Redis** para optimizar consultas.



3.3 Modelo de Componentes (Nivel 3)

- **Microservicio de Autenticación**: Utiliza OAuth 2.0 con el flujo **Authorization Code Flow** para mayor seguridad, Este flujo separa el cliente de la obtención directa de tokens de acceso, lo que lo hace ideal para aplicaciones que requieren autenticación robusta a parte de la integración con navegadores seguros y **PKCE** capa extra de seguridad al incluir una prueba de clave que es verificada al intercambiar el código de autorización por un token.
- **Onboarding con Reconocimiento Facial**: Se puede integrar con **Azure Face API** para el reconocimiento facial durante el registro de nuevos usuarios dependerá de la escabilidad
- **Autenticación y Autorización**: Luego del onboarding, los usuarios podrán acceder mediante huella digital o claves, con opciones de autenticación mediante **FIDO2**.
- **Monitoreo y Tolerancia a Fallos**:
    - Monitoreo en tiempo real con herramientas como **Prometheus** y **Grafana**.
    - Implementación de **auto-healing** con contenedores como **Kubernetes**.
    - Alta disponibilidad utilizando estrategias multi-región en la nube (Azure o AWS) y recuperación ante desastres (DR) usando backups automáticos por veeambackup para equipos virtuales Onpremise se puede manejar replicas por SRM en VMware.



[Components] Diagrama de Componentes
Solución para el sistema de banca
por internet de la entidad BP

4. Recomendaciones de Infraestructura

- Uso de infraestructura en la nube (Azure o AWS).
    - **AWS**:

      Elastic Load Balancer (ELB) para distribuir la carga.

      Amazon RDS para bases de datos relacionales.

S3 para almacenamiento y backups.

- **Azure**:

  Azure Traffic Manager para gestionar el tráfico de las aplicaciones.

  Cosmos DB como base de datos escalable.

- Alta disponibilidad, recuperación ante desastres y escalabilidad.

  Implementa balanceo de carga y replicación de bases de datos.

  Usar servicios en la nube (AWS o Azure) para asegurar alta disponibilidad y recuperación ante desastres o SRM para réplicas de equipos virtuales onpremise.

- Estrategias de caching y monitoreo.

5. Patrones de Diseño

- Propuesta de un patrón de **Event-Driven Architecture** para las notificaciones y auditoría.
- Uso de **Circuit Breaker** para asegurar resiliencia en los servicios externos.

6. Conclusión

Esta solución ofrece una arquitectura moderna, escalable y segura para BP, adaptándose a las necesidades actuales y futuras del sistema bancario. Además, su modularidad y uso de tecnologías en la nube proporcionan la flexibilidad necesaria para crecer y evolucionar según sea necesario, garantizando una excelente experiencia para los usuarios finales.