

Lcdo. Carlos Mera Arquitecto de software	DISEÑO DE ARQUITECTURA	Código	2025-0001
	Sistema de Banca por Internet BP	Versión	1
		Página	1 de 9

DISEÑO DE ARQUITECTURA

Sistema de Banca por Internet BP

Guayaquil, 20 agosto 2025

Lcdo. Carlos Mera Arquitecto de software	DISEÑO DE ARQUITECTURA	Código	2025-0001
	Sistema de Banca por Internet BP	Versión	1
		Página	2 de 9

Contenido

Sistema de Banca por Internet BP 3

1. Introducción 3

2. Alcance 3

3. Requerimientos funcionales y no funcionales 3

4. Decisiones arquitectónicas 4

5. Modelo C4 5

 5.1 Diagrama de Contexto 5

 5.2 Diagrama de Contenedores 5

 5.3 Diagrama de Componentes..... 7

6. Arquitectura de seguridad y autenticación..... 8

7. Alta disponibilidad, tolerancia a fallos y monitoreo 8

8. Auditoría y persistencia..... 8

9. Cumplimiento normativo 8

10. Cumplimiento normativo 8

11. Manejo de Costos..... 8

 11.1 Selección de Servicios Cloud 9

 11.2 Optimización de Recursos 9

 11.3 Monitoreo y Alertas de Costos..... 9

 11.4 Capa Gratuita y Servicios Open Source..... 9

Lcdo. Carlos Mera Arquitecto de software	DISEÑO DE ARQUITECTURA	Código	2025-0001
	Sistema de Banca por Internet BP	Versión	1
		Página	3 de 9

Sistema de Banca por Internet BP

1. Introducción

Este documento presenta el diseño de la arquitectura para el sistema de Banca por Internet de la entidad BP. El objetivo es ofrecer una solución segura, escalable y moderna, que permita a los clientes:

- Consultar históricos de movimientos.
- Realizar transferencias internas e interbancarias.
- Recibir notificaciones de todas las operaciones realizadas.

El diseño sigue el modelo C4 (Contexto → Contenedores → Componentes) y adopta prácticas recomendadas de seguridad, alta disponibilidad y cumplimiento normativo.

2. Alcance

La solución completa:

- Aplicación web tipo SPA.
- Aplicación móvil multiplataforma.
- Microservicios en arquitectura desacoplada.
- Integración con sistemas Core y CRM de clientes.
- Autenticación con OAuth 2.0 y flujos seguros para móviles y SPA. Persistencia, auditoría y notificaciones multi-canal.
- Implementación en nube con servicios gestionados para HA/DR.

3. Requerimientos funcionales y no funcionales

Tipo	Requerimiento
Funcional	Consultar movimientos, realizar transferencias, pagos internos y externos.
Funcional	Notificar al usuario por email/SMS/push en cada transacción.
No funcional	Alta disponibilidad, baja latencia, escalabilidad horizontal.
No funcional	Cumplimiento normativo: PCI DSS, ISO 27001, GDPR/Ley de Datos Personales.
No funcional	Autenticación multifactor y Onboarding biométrico.

Lcdo. Carlos Mera Arquitecto de software	DISEÑO DE ARQUITECTURA	Código	2025-0001
	Sistema de Banca por Internet BP	Versión	1
		Página	4 de 9

4. Decisiones arquitectónicas

Decisión	Opción elegida	Justificación
Arquitectura	Microservicios + API Gateway	Escalabilidad independiente, centralización de seguridad, integración simple.
Frontend Web	React / Angular SPA	Experiencia de usuario fluida, comunidad madura, soporte a largo plazo.
App móvil	Flutter / React Native	Multiplataforma, desarrollo ágil, reducción de costos.
Autenticación	OAuth2.0 + PKCE + MFA	Seguridad reforzada, estándar reconocido, soporte biométrico.
Base de datos transaccional	PostgreSQL (RDS/Azure DB)	Consistencia ACID, soporte HA y replicación.
Cache	Redis/ElastiCache	Baja latencia para clientes frecuentes y sesiones.
Notificaciones	SNS/SES, Twilio, Firebase	Multi-canal, alta disponibilidad, desacoplamiento por eventos.
Mensajería asíncrona	Kafka/SQS	Resiliencia, desacoplamiento, tolerancia a fallos.
Infraestructura	AWS/Azure (contenedores/Kubernetes)	Escalabilidad, HA/DR, integración con servicios nativos cloud.

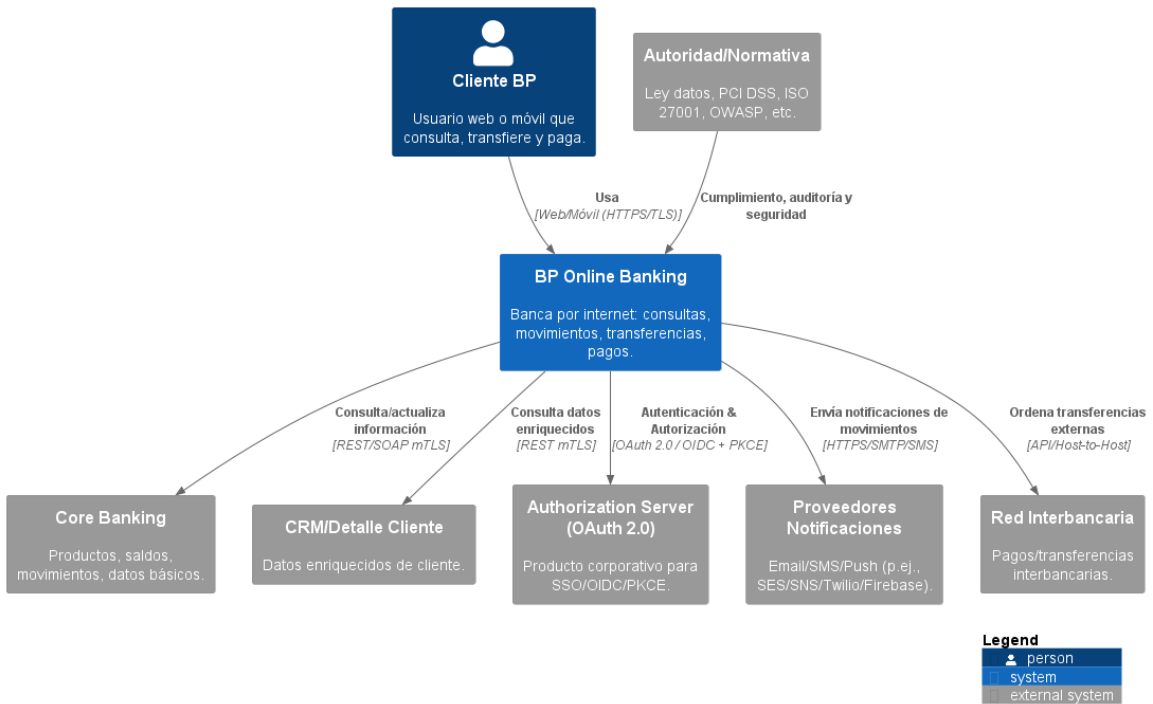
Lcdo. Carlos Mera Arquitecto de software	DISEÑO DE ARQUITECTURA	Código	2025-0001
	Sistema de Banca por Internet BP	Versión	1
		Página	5 de 9

5. Modelo C4

5.1 Diagrama de Contexto

Descripción: Vista general para usuarios de negocio.

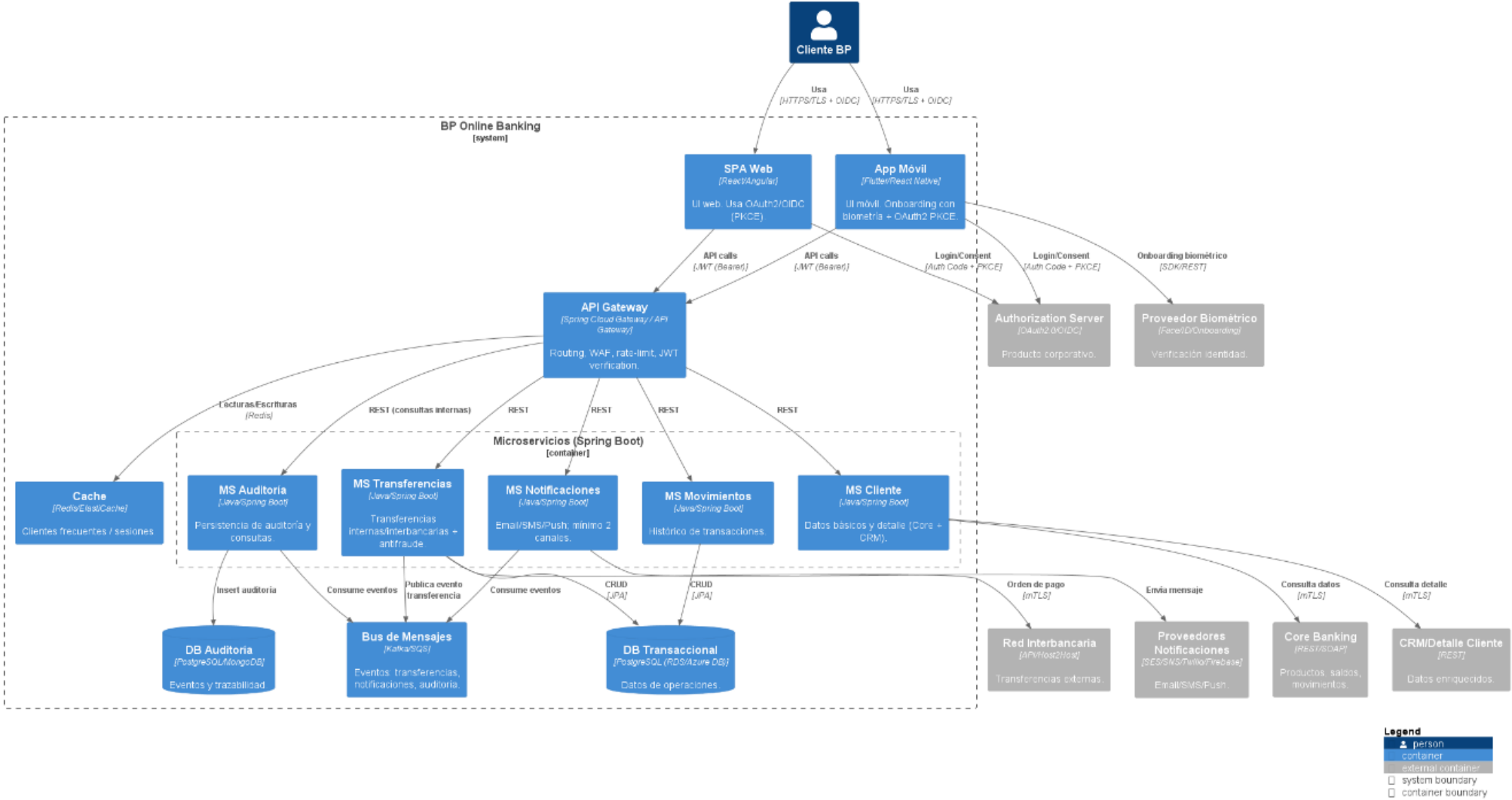
Elementos: Cliente Web/Móvil, BP Online Banking, Core Banking, CRM, Notificaciones, Autoridad Normativa.



5.2 Diagrama de Contenedores

Descripción: Vista técnica de alto nivel. Muestra frontends, microservicios, API Gateway, BBDD, mensajería y sistemas externos.

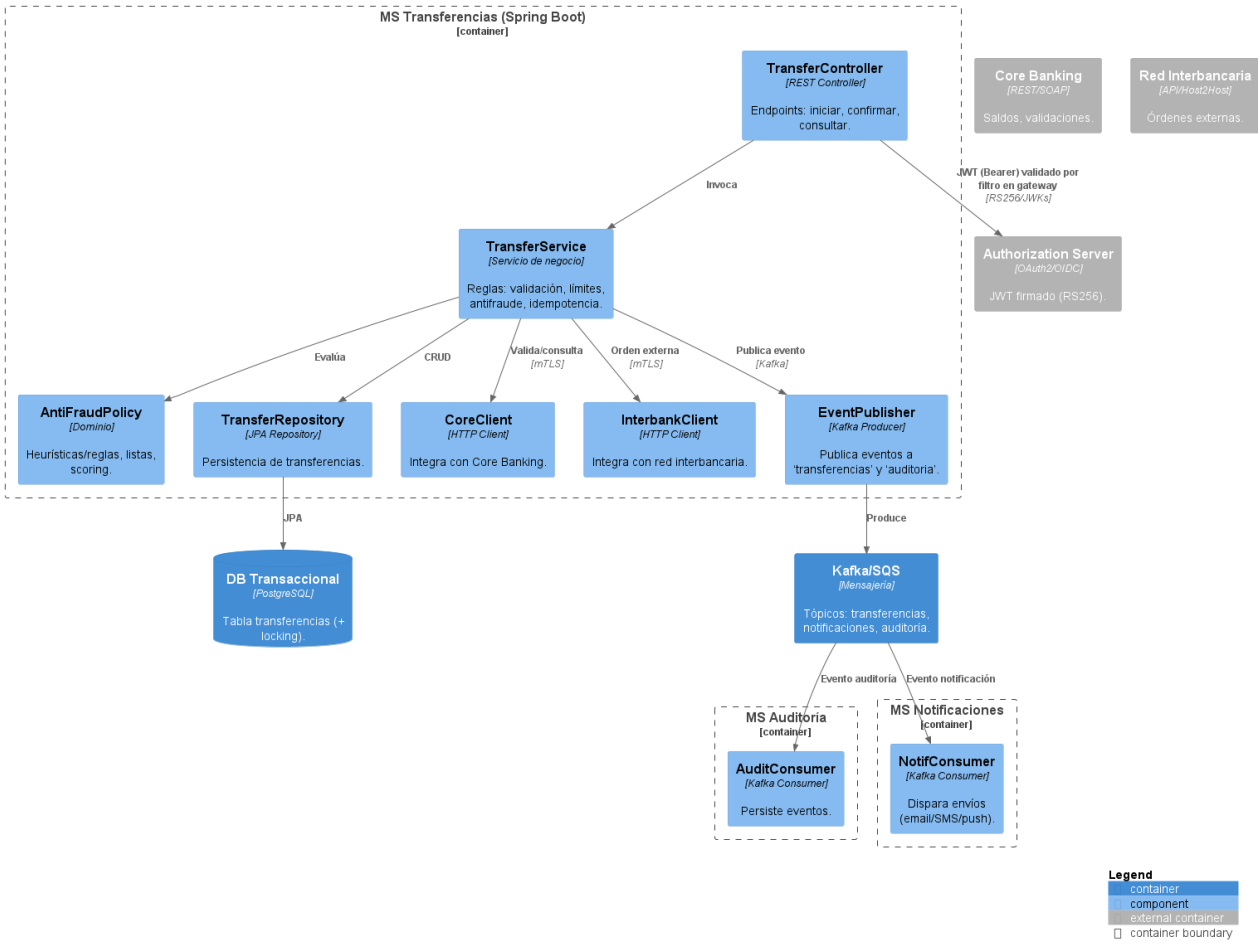
Lcdo. Carlos Mera Arquitecto de software	DISEÑO DE ARQUITECTURA	Código	2025-0001
	Sistema de Banca por Internet BP	Versión	1
		Página	6 de 9



Lcdo. Carlos Mera Arquitecto de software	DISEÑO DE ARQUITECTURA	Código	2025-0001
	Sistema de Banca por Internet BP	Versión	1
		Página	7 de 9

5.3 Diagrama de Componentes

Descripción: Vista detallada del microservicio de Transferencias.



Lcdo. Carlos Mera Arquitecto de software	DISEÑO DE ARQUITECTURA	Código	2025-0001
	Sistema de Banca por Internet BP	Versión	1
		Página	8 de 9

6. Arquitectura de seguridad y autenticación

- OAuth 2.0 con PKCE para SPA y App Móvil.
- MFA: usuario+clave + OTP/SMS o biometría (App Móvil).
- Onboarding biométrico con proveedores externos (ej. AWS Rekognition, Azure Face API).
- Cifrado extremo a extremo: TLS 1.3 en tránsito, KMS/KeyVault en reposo.
- Firewalls y WAF para proteger APIs públicas.
- Políticas de seguridad: OWASP Top 10, PCI DSS, ISO 27001.

7. Alta disponibilidad, tolerancia a fallos y monitoreo

- Multi-AZ y backups automáticos para bases de datos.
- Auto-scaling en microservicios con health checks.
- Mensajería asíncrona para resiliencia en fallos.
- Monitoreo con ELK/CloudWatch/Azure Monitor + alertas en tiempo real.
- Circuit breakers y retries en integraciones externas.

8. Auditoría y persistencia

- Base de datos de auditoría separada de la transaccional.
- Publicación de eventos a Kafka/SQS para registrar cada acción del cliente.
- Consultas de auditoría disponibles para auditores internos y externos.

9. Cumplimiento normativo

La arquitectura cumple con:

- Ley de Datos Personales/GDPR: Protección y consentimiento del cliente.
- PCI DSS: Requisitos de seguridad para datos financieros.
- ISO 27001 / NIST: Controles de seguridad y gestión de riesgos.
- Open Banking y OWASP: Estándares de seguridad web y APIs.

10. Cumplimiento normativo

La arquitectura propuesta garantiza:

- Seguridad y cumplimiento normativo.
- Escalabilidad y resiliencia mediante microservicios y nube. Experiencia de usuario óptima en web y móvil.
- Flexibilidad para futuros servicios y canales de integración.

11. Manejo de Costos

Para garantizar una arquitectura eficiente no solo en rendimiento y seguridad, sino también en costos, se proponen las siguientes estrategias:

Lcdo. Carlos Mera Arquitecto de software	DISEÑO DE ARQUITECTURA	Código	2025-0001
	Sistema de Banca por Internet BP	Versión	1
		Página	9 de 9

11.1 Selección de Servicios Cloud

- Servicios serverless o gestionados: Uso de AWS Lambda/Azure Functions para tareas event-driven (p. ej., notificaciones) reduce costos en entornos de baja carga.
- Bases de datos administradas: RDS/Azure Database para reducir costos de operación y mantenimiento.
- Almacenamiento escalable: S3/Azure Blob Storage para logs y backups con políticas de ciclo de vida.

11.2 Optimización de Recursos

- Auto-scaling: Ajuste dinámico de microservicios según la carga, evitando sobredimensionamiento.
- Instancias reservadas y savings plans: Para cargas predecibles, reducir costos hasta un 70%.
- Uso de spot instances: Para tareas batch o no críticas, aprovechando descuentos significativos.

11.3 Monitoreo y Alertas de Costos

- Integración con AWS Cost Explorer o Azure Cost Management para alertas en tiempo real.
- Paneles con métricas de costos por servicio y centro de costos para identificar rápidamente desviaciones.

11.4 Capa Gratuita y Servicios Open Source

- Uso de herramientas open source (p. ej., ELK para monitoreo, Redis para cache) en lugar de soluciones propietarias costosas cuando sea viable.