Arquitectura Soluciones en Azure - BP

# 1. Resumen Ejecutivo

BP Online Banking es una plataforma moderna en Azure que habilita consulta de movimientos, pagos y transferencias propias e interbancarias. La solución usa APIM como puerta de enlace, microservicios .NET en Azure Container Apps, lectura rápida con Cosmos DB (CQRS) y transaccional/auditoría en Azure SQL (Ledger). La autenticación se implementa con OAuth2/OIDC (Authorization Code + PKCE) sobre Entra ID/B2C, incorporando onboarding biométrico. La mensajería con Azure Service Bus desacopla procesos y alimenta notificaciones (Microsoft Graph/ACS) y auditoría. La arquitectura prioriza seguridad (WAF, APIM policies, Private Link, Key Vault), alta disponibilidad, DR multi-región, observabilidad (App Insights) y control de costos (serverless y autoscale), cumpliendo normativa de protección de datos y estándares de seguridad para el sector financiero.

# 2. Diagrama de Contexto (C1)

## Actores

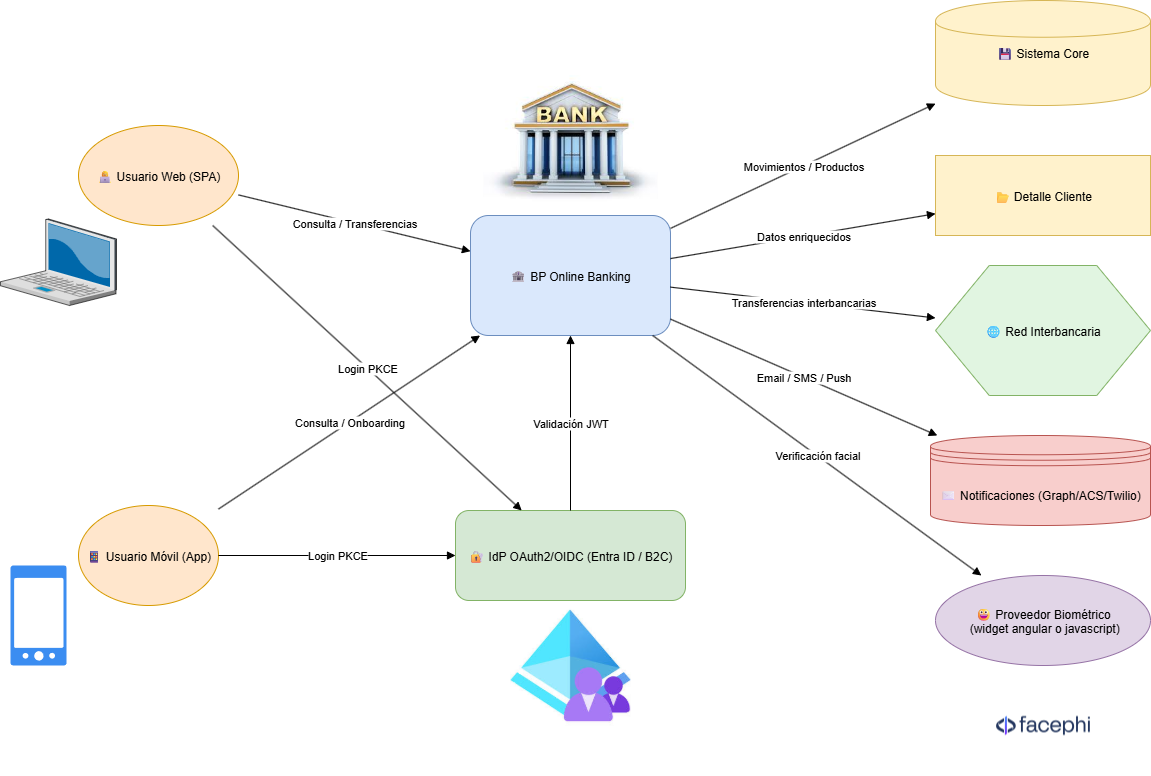
- Usuario web (SPA)  
- Usuario móvil (App)  
- Soporte / BP Ops

## Sistemas externos

- Core Bancario (productos, saldos, movimientos)  
- Sistema de Detalle de Cliente (información enriquecida)  
- Red Interbancaria / Switch (transferencias ISO 8583/REST)  
- Proveedores de notificaciones (Microsoft Graph, Azure Communication Services, Twilio/SendGrid)  
- Proveedor biométrico (Azure AI Vision, Onfido, Jumio, Facephi widget (angular o javascript))  
- Identity Provider OAuth2/OIDC (Microsoft Entra ID / Entra ID B2C)

## Sistema BP Online Banking

- SPA (Angular)  
- Mobile App (Flutter)  
- API Gateway (Azure API Management – APIM)  
- Microservicios en .NET: Customer Basic Data, Movements Query, Transfers, Notifications, Onboarding Orchestrator, Audit & Compliance, BFF Web/Mobile (opcional)  
- Bases de datos: Cosmos DB, Azure SQL, Blob Storage  
- Mensajería: Azure Service Bus



# 3. Diagrama de Contenedores (C2)

## Frontends

- SPA (Angular)  
- Mobile (Flutter)

## Edge

- Azure Front Door + WAF  
- APIM (API Gateway, policies, subscription key, OAuth2/OIDC)

## Backend – Container Apps

- BFF Web y Mobile  
- Customer Basic Data Service  
- Movements Query Service  
- Transfers Service  
- Notifications Service  
- Onboarding Orchestrator  
- Audit & Compliance  
- Identity Adapter

## Datos

- Azure SQL Database  
- Cosmos DB  
- Blob Storage  
- Redis (opcional)

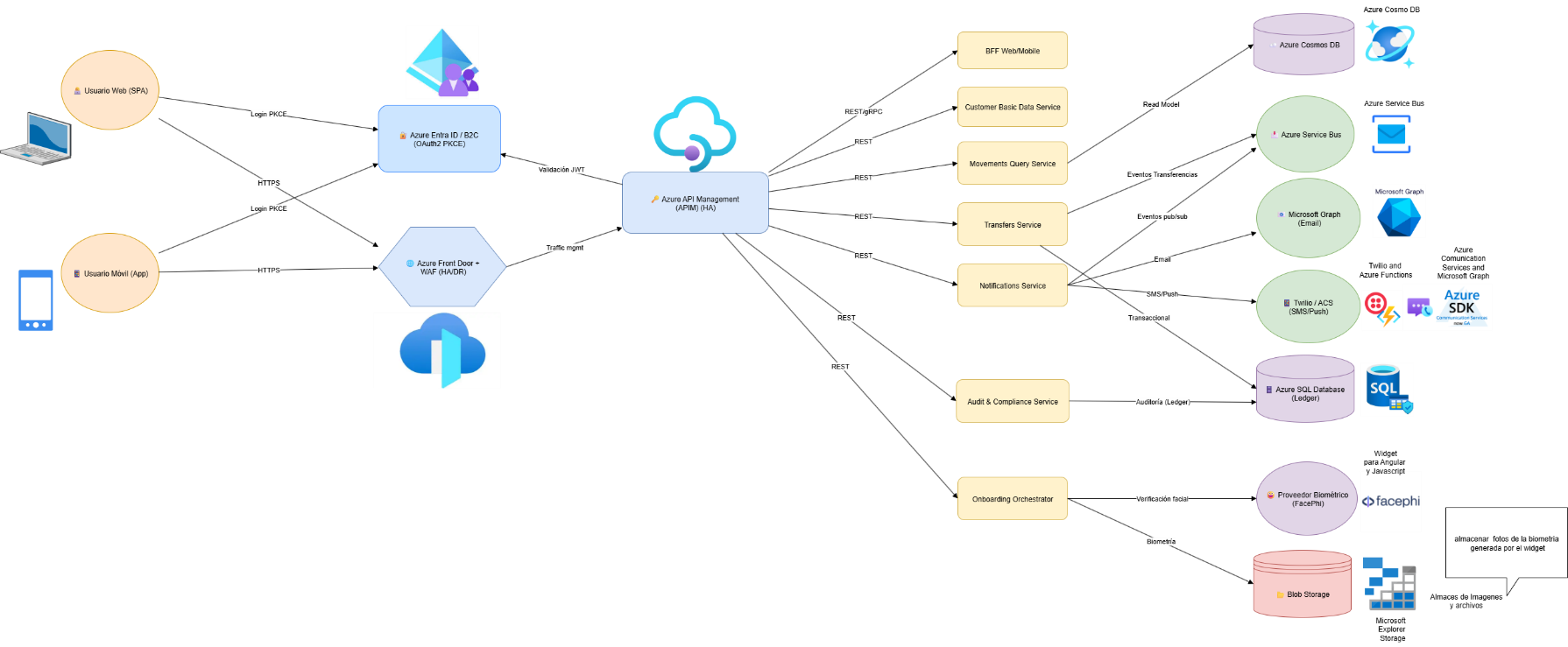
## Mensajería

- Azure Service Bus  
- Event Grid (opcional)

## Seguridad y Observabilidad

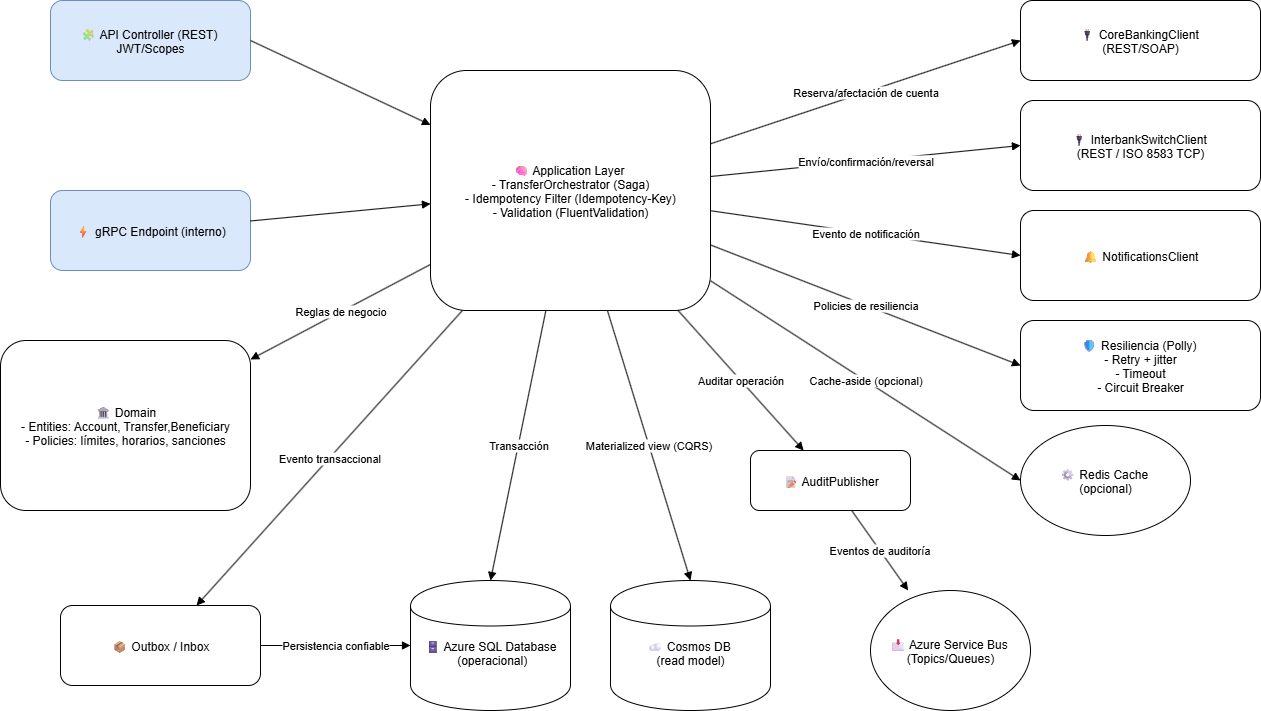
- Application Insights  
- Azure Monitor  
- Sentinel  
- Key Vault  
- Defender for Cloud  
- Private Link

1. **Front y Mobile App** → representaste **SPA** y **App Móvil** como actores ✔️
2. **Front Door + WAF** → perfecto, cumple con **seguridad y balanceo global** ✔️
3. **API Management (APIM)** → cubre **API Gateway y control de entrada** ✔️
4. **Microservicios en contenedores** → segmentaste bien: Customer, Movements, Transfers, Notifications, Audit, Onboarding, BFF ✔️
5. **Almacenamiento y BD** → incluiste SQL, Cosmos, Blob Storage, Service Bus ✔️
6. **Comunicación** → las conexiones REST/gRPC y pub/sub están claras ✔️
7. **Escalabilidad y resiliencia** → implícitamente se cumple con Azure Service Bus, Cosmos y APIM ✔️



# 4. Diagrama de Componentes (C3)

Ejemplo: Transfers Service  
  
- API Controller (REST) / gRPC Endpoint  
- Application Layer: Transfer Orchestrator (Saga), Idempotency Filter, Validations  
- Domain Layer: Entities (Account, Transfer, Beneficiary), Policies (límites, horarios)  
- Integration: CoreBankingClient, InterbankSwitchClient, NotificationsClient  
- Resiliencia: Polly  
- Outbox/Inbox, AuditPublisher  
- Storage: SQL y Cosmos  
- Seguridad: JWT, Roles/Scopes/ABAC



# 5. Flujos Clave

## Autenticación

- Authorization Code + PKCE  
- Client Credentials para servicios internos

## Onboarding con biometría

1. Captura rostro + liveness  
2. Validación con proveedor biométrico  
3. Alta en Entra ID B2C  
4. Registro en SQL Ledger y Blob

## Consulta de movimientos

BFF → Movements Query → Cosmos DB (read model)

## Transferencias

SPA/App → APIM → Transfers Service → Core → Switch → Notificaciones

## Notificaciones

- Email: Graph  
- SMS: ACS/Twilio  
- Push: Notification Hubs

## Auditoría

Eventos en Service Bus → SQL Ledger + Event Hub/Sentinel

# 6. Patrones y Justificaciones

- Microservicios + APIM: escalado independiente, gobernanza centralizada  
- CQRS + Read Models: baja latencia, separación de lectura/escritura  
- Service Bus: desacoplamiento, reintentos, DLQ  
- Outbox/Inbox + Idempotency: consistencia, trazabilidad  
- OAuth2/OIDC con PKCE: seguridad clientes públicos  
- Container Apps: serverless, autoscale, bajo costo

# 7. Normativa y Seguridad

- LOPDP, GDPR  
- MFA, OAuth2/OIDC PKCE  
- TLS 1.2+, TDE, SSE  
- APIM policies, scopes, subscription key  
- Private Link, NSG, DDoS Protection  
- SQL Ledger, retención, mascaramiento PII  
- ISO 27001, NIST, OWASP ASVS

# 8. Alta Disponibilidad, DR, Monitoreo y Costos

- HA: Front Door, Container Apps, Cosmos multi-region, SQL geo-replication  
- DR: RPO ≤ 5 min, RTO ≤ 30 min  
- Monitoreo: App Insights, Log Analytics  
- Costos: serverless per-use, autoscale, budgets Azure Advisor

# 9. Texto guía para draw.io

Contexto: Usuarios → Front Door → APIM → Microservicios → Core/Sistema Detalle Cliente/Switch/Notificaciones/Biométrico  
Contenedores: SPA, Mobile, Front Door, APIM, Services, Service Bus, SQL, Cosmos, Blob, Key Vault, App Insights  
Componentes: Transfers Service → API Controller, Orchestrator, Entities, Integration, Polly, Outbox, AuditPublisher