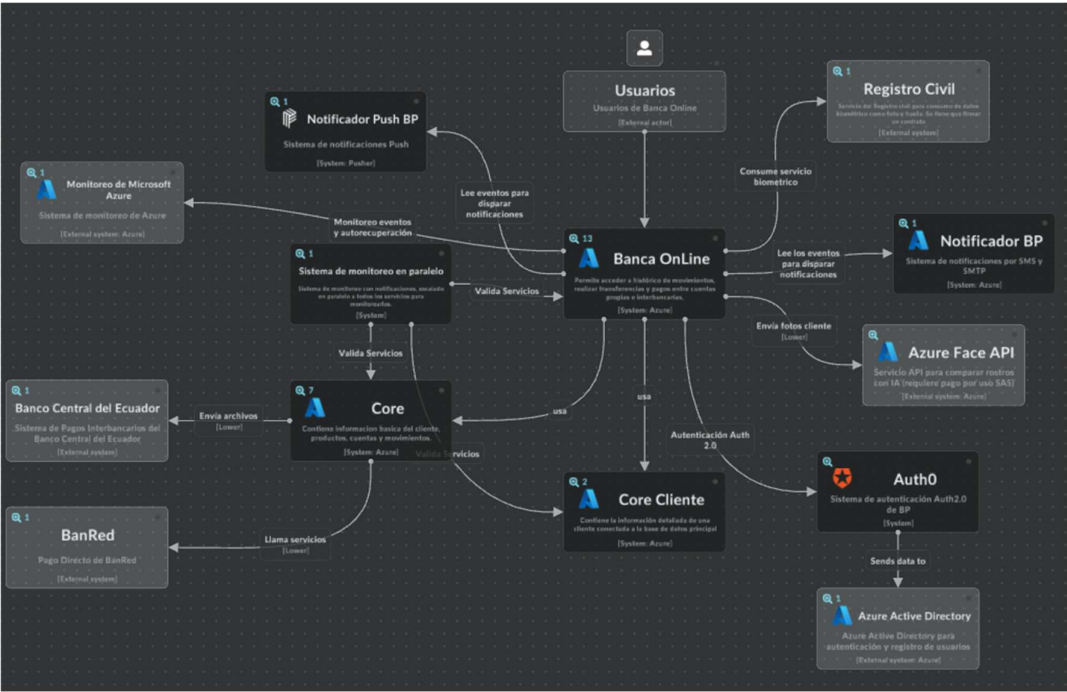


Usted ha sido contratado por una entidad llamada BP como arquitecto de soluciones para diseñar un sistema de banca por internet, en este sistema los usuarios podrán acceder al histórico de sus movimientos, realizar transferencias y pagos entre cuentas propias e interbancarias.

Toda la información referente al cliente se tomará de 2 sistemas, una plataforma Core que contiene información básica de cliente, productos, cuentas, movimientos, y un sistema independiente que complementa la información del cliente cuando los datos se requieren en detalle este sistema core está conectado a una base de datos principal.



Debido a que la norma exige que los usuarios sean notificados sobre los movimientos realizados, el sistema utilizará sistemas externos o propios de envío de notificaciones, mínimo 2.

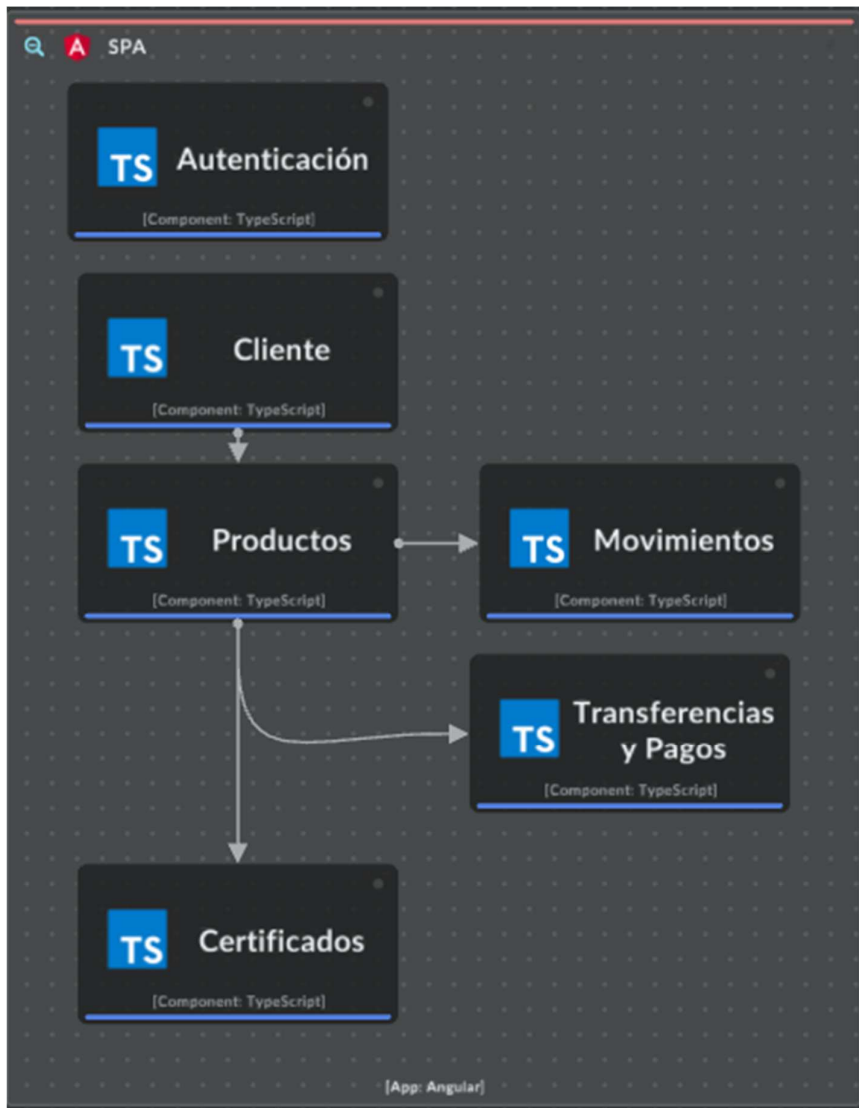
Existirá un notificador basado en Apache Kafka con subscriptores para mensajes SMS y otros SMTP, y otro adicional para notificaciones push para la app móvil.





Al realizar transferencias o pagos se publicara un mensaje para que los Sistemas de notificación Push, SMS, SMTP envíen las alertas al cliente.

Este sistema contará con 2 aplicaciones en el Front, una SPA



y una Aplicación móvil desarrollada en un Framework multiplataforma. (Mencione 2 opciones y justifique el porqué de su elección)

SPA: Angular

APP: React Native

Otras opciones pueden ser VUE para SPA y Xamarin de Microsoft.

Angular + React Native ofrece ventajas sobre **Vue + Xamarin** para una aplicación de banca móvil en términos de seguridad, escalabilidad, rendimiento y soporte.

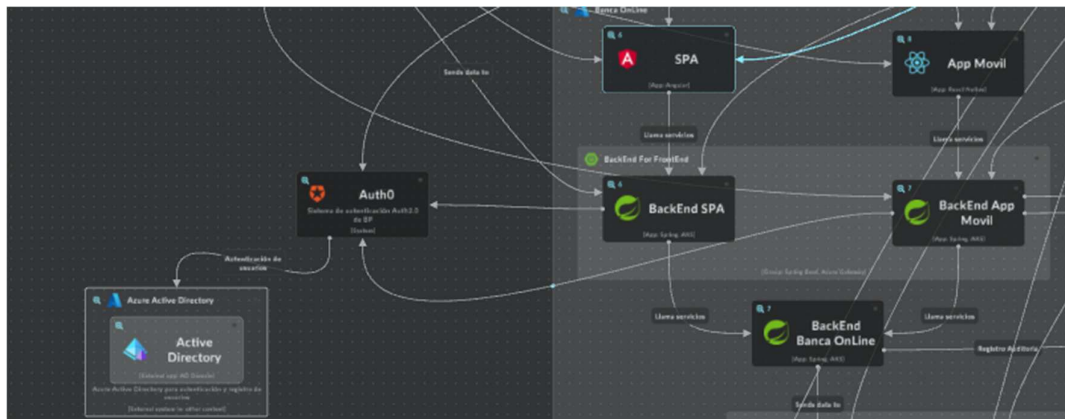
Angular es más robusto y seguro que Vue, con mejor estructura para proyectos grandes y protección contra ataques es desarrollado por Google y tiene soporte continuo y parches de seguridad.

React Native tiene mejor integración con iOS y Android que Xamarin, proporcionando una mejor experiencia de usuario y rendimiento.

Seguridad y mantenimiento: Angular maneja mejor autenticación y protección de datos, y React Native tiene un ecosistema más amplio que facilita la integración y actualizaciones.

Compatibilidad y soporte: React Native recibe mejoras constantes, mientras que Xamarin depende de Microsoft, lo que puede limitar su evolución.

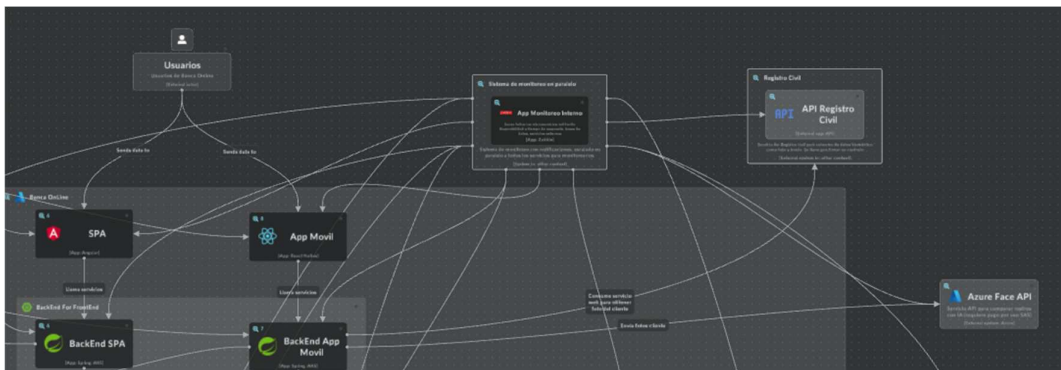
Ambas aplicaciones autenticarán a los usuarios mediante un servicio que usa el estándar OAuth2.0, para el cual no requiere implementar toda la lógica, ya que la compañía cuenta con un producto que puede ser configurado para este fin; sin embargo, debe dar recomendaciones sobre cuál es el mejor flujo de autenticación que se debería usar según el estándar.



Auth0 maneja la autenticación por MFA, roles y permisos. Adicional posee seguridad contra ataques de bots y fuerza bruta.

El flujo de código de autorización con PKCE (Authorization Code Flow with PKCE) se usa para evitar el ataque de interceptación de código de autorización podemos

Tenga en cuenta que el sistema de Onboarding para nuevos clientes en la aplicación móvil usa reconocimiento facial, por tanto, su arquitectura deberá considerarlo como parte del flujo de autorización y autenticación, a partir del Onboarding el nuevo usuario podrá ingresar al sistema mediante usuario y clave, huella o algún otro método especifique alguno de los anteriores dentro de su arquitectura, también puede recomendar herramientas de industria que realicen estas tareas y robustezca su aplicación.



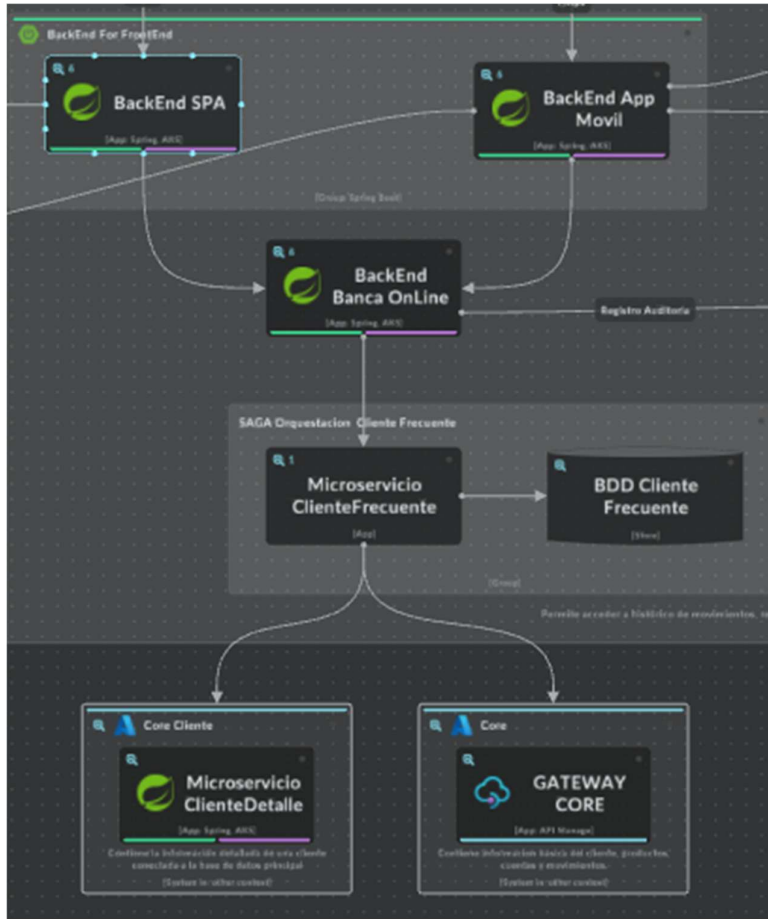
El sistema utiliza una base de datos de auditoría que registra todas las acciones del cliente



Para este caso vamos a implementar el patrón de publicador/suscriptor mediante Apache Kafka toda acción en la app publicara tópicos para que un suscriptor de

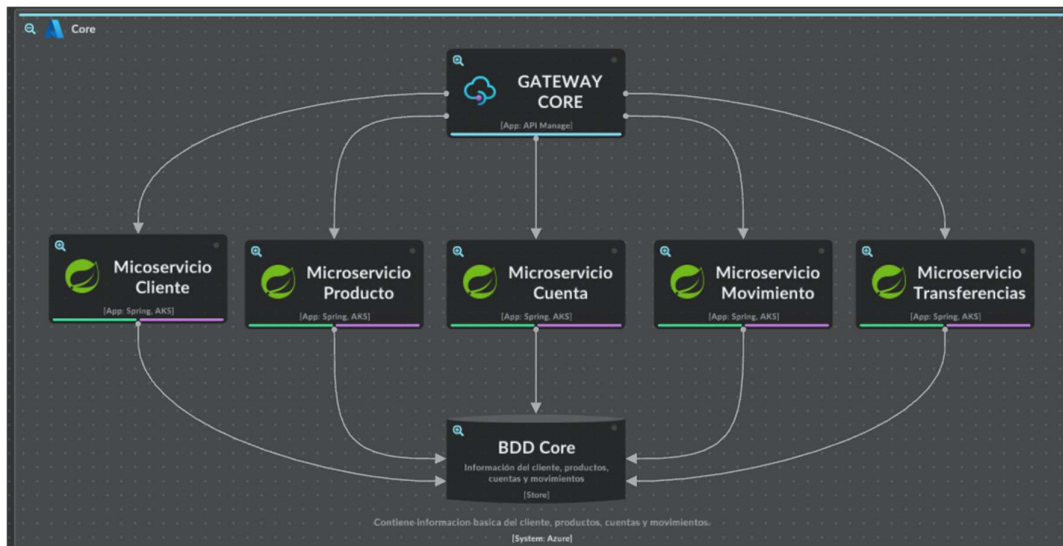
auditoria los esté escuchando y los vaya procesando sin afectar al rendimiento de la aplicación.

y cuenta con un mecanismo de persistencia de información para clientes frecuentes, para este caso proponga una alternativa basada en patrones de diseño que relacione los componentes que deberían interactuar para conseguir el objetivo.



Existirá un servicio que identificara si el cliente es frecuente y persistirá la información del mismo en una BDD usando el patrón SAGA Orquestación

Para obtener los datos del cliente el sistema pasa por una capa de integración compuesta por un api Gateway y consume los servicios necesarios de acuerdo con el tipo de transacción, inicialmente usted cuenta con 3 servicios principales, consulta de datos básicos, consulta de movimientos y transferencias, que realiza llamados a servicios externos dependiendo del tipo, si considera que debería agregar más servicios para mejorar la repuesta de información a sus clientes, es libre de hacerlo.



Para este reto, mencione aquellos elementos normativos que podrían ser importantes a la hora de crear aplicaciones para entidades financieras, Ejemplo ley de datos personales, seguridad etc.

Datos personales.

Intereses, tarifas y comisiones.

ISO 27000

Código Orgánico Monetario y Financiero (COMF):

Define el marco legal para la banca, incluyendo la regulación de la banca electrónica y los sistemas de pago.

Ley Orgánica para Defender los Derechos de los Clientes del Sistema Financiero Nacional:

Protege a los usuarios de la banca, estableciendo normas para la transparencia, la protección de datos y la prevención de abusos.

Resoluciones de la Superintendencia de Bancos:

La Superintendencia de Bancos y Seguros (SB) emite resoluciones que complementan el marco legal y regulan aspectos específicos de la banca electrónica, como la seguridad de los canales electrónicos y la gestión de riesgos.

Resoluciones de la Junta de Política y Regulación Monetaria (JPRM):

La JPRM dicta normativas sobre los medios de pago electrónicos y la actividad de las Fintech.

Aspectos clave de la regulación de la banca electrónica:

Seguridad:

Las instituciones financieras deben implementar medidas de seguridad tecnológica y de control para garantizar la integridad, disponibilidad y confidencialidad de los datos de los usuarios en los canales electrónicos.

Transparencia:

Las entidades financieras deben informar de manera clara y concisa a los clientes sobre los costos, tarifas, riesgos y beneficios de los productos y servicios financieros ofrecidos a través de canales electrónicos.

Protección de datos:

Se debe garantizar la confidencialidad y la protección de los datos personales de los usuarios en los canales electrónicos, cumpliendo con la Ley Orgánica de Protección de Datos Personales.

Derechos de los usuarios:

Los usuarios tienen derecho a reclamar, presentar quejas y denunciar irregularidades ante la Superintendencia de Bancos y Seguros o las instancias competentes.

Garantice en su arquitectura, alta disponibilidad (**HA**), tolerancia a fallos, recuperación ante desastres (**DR**), Seguridad y Monitoreo, Excelencia operativa y auto-healing.

OnPremise

HA

Para servidores físicos o virtuales, es fundamental desplegar un clúster con balanceo de carga para optimizar la distribución del tráfico y garantizar la disponibilidad.

Si se requieren contenedores OnPremise, BP dispone de OpenShift, que ofrece alta disponibilidad mediante la escalabilidad horizontal de PODs. Esta funcionalidad debe configurarse adecuadamente en los archivos de configuración vinculados a los pipelines de implementación.

DR

Banco Pichincha ya cuenta con su DCA implementado, por lo que es necesario desplegar la infraestructura adecuada para la réplica de información. Por ejemplo, los datos y aplicaciones pueden replicarse mediante copias de almacenamiento, mientras que las bases de datos pueden sincronizarse utilizando tecnologías como AlwaysOn o el modelo de Publicador/Distribuidor/Suscriptor.

Seguridad y Monitoreo, Excelencia operativa y auto-healing

Las aplicaciones OnPremise se conectan a una plataforma de monitoreo a través de agentes encargados de detectar eventos, analizar puertos y realizar búsquedas en logs, presentando los datos en dashboards de supervisión y notificaciones.

El estado de los componentes se evalúa mediante reglas predefinidas, mientras que las acciones correctivas pueden ser ejecutadas de forma automática según reglas establecidas o de manera manual a través de tareas específicas.

Cloud

HA

Al implementar esta solución en Azure sobre ASK podemos incluir más seguridad como Microsoft Defender para Kubernetes, incluimos el tema de monitoreo y control y también la excelencia operativa ya que AKS escalara automáticamente de manera horizontal nuestros Microservicios y esto brindara calidad de servicio a nuestros clientes y reducirá el mantenimiento de infraestructura local.

DR

Dentro de Azure se debe configurar planes con redundancia y replica a otras zonas.

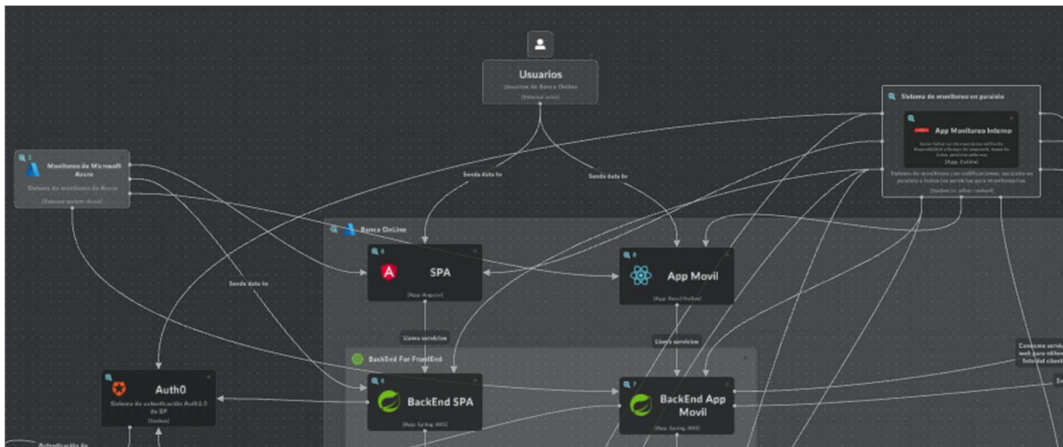
Tolerancia a fallos:

Al implementar nuestra arquitectura desacoplamos y generamos Microservicios con una única responsabilidad, además implementaremos en los componentes críticos como consulta de clientes y productos un patron de reintentos adicional al cliente frecuente para traer la información, y si servicios como el de movimientos no responde para una cuenta, no parara la funcionalidad de todo el sistema por ello.

Seguridad y Monitoreo, Excelencia operativa y auto-healing

Al implementar esta solución en Azure sobre ASK podemos incluir más seguridad como Microsoft Defender para Kubernetes, incluimos el tema de monitoreo y control y también la excelencia operativa ya que AKS escalara automáticamente de manera horizontal nuestros Microservicios y esto brindara calidad de servicio a nuestros clientes y reducirá el mantenimiento de infraestructura local.

Para el Auto-healing vamos a aprovechar la capacidad de Azure dentro del Azure App Service donde configuraremos las reglas y las acciones que Azure auto healing services.



Sí lo considera necesario, su arquitectura puede contener elementos de infraestructura en nube como Azure o AWS, garantice baja latencia en sus servicios, cuenta con presupuesto para esto.

Se implemento microservicios sobre Azure que se despliegan sobre AKS lo que brinda resiliencia, escalabilidad, supervisión y monitoreo con Azure Monitor y adicional Zabbix para tener un doble monitoreo ya que puede ser que existan problemas de comunicación hacia el exterior de Azure.

En lo posible plantee una arquitectura desacoplada con elementos y reusables y cohesionados para otros componentes que puedan adicionarse en el futuro.

Toda la arquitectura se implemento con Microservicios respetando los principios SOLID, para que cumplan una sola función y se encuentren desacoplados siendo reutilizables.



El modelo debe ser desarrollado bajo **c4**(Modelo de Contexto, Modelo de aplicación o contenedor y Componentes), describa hasta el modelo de componentes, la infraestructura la puede modelar como usted lo considere usando la herramienta de su preferencia.

Link del modelo C4 en IcePanel: <https://s.icepanel.io/MJn9sBcJNj3Wp6/AZTo>

Link Github: <https://github.com/majurado/TestArquitectura>

¡Éxitos!

Alexander Jurado's landscape

Generated by [IcePanel](#) at 2025-05-30 04:27:03

Table of contents

- [1. Context diagrams](#)
 - [1.1. Banca OnLine - Contexto](#)
- [2. App diagrams](#)
 - [2.1. Azure Active Directory App Diagram](#)
 - [2.2. Azure Face API App Diagram](#)
 - [2.3. Banca OnLine App Diagram](#)
 - [2.4. Banco Central del Ecuador App Diagram](#)
 - [2.5. BanRed App Diagram](#)
 - [2.6. Core App Diagram](#)
 - [2.7. Core Cliente App Diagram](#)
 - [2.8. Monitoreo de Microsoft Azure App Diagram](#)
 - [2.9. Notificador BP App Diagram](#)
 - [2.10. Notificador Push BP App Diagram](#)
 - [2.11. Registro Civil App Diagram](#)
 - [2.12. Sistema de monitoreo en paralelo App Diagram](#)
- [3. App diagram flows](#)
 - [3.1. New customer flow](#)
- [4. Component diagrams](#)
 - [4.1. App Movil Component Diagram](#)
 - [4.2. BackEnd App Movil Component Diagram](#)
 - [4.3. BackEnd Banca OnLine Component Diagram](#)
 - [4.4. BackEnd SPA Component Diagram](#)
 - [4.5. Micoservicio Cliente Component Diagram](#)
 - [4.6. Microservicio ClienteDetalle Component Diagram](#)
 - [4.7. Microservicio ClienteFrecuente Component Diagram](#)

- [4.8. Microservicio Cuenta Component Diagram](#)
- [4.9. Microservicio Movimiento Component Diagram](#)
- [4.10. Microservicio Producto Component Diagram](#)
- [4.11. Microservicio Transferencias Component Diagram](#)
- [4.12. MicroservicioNotificaciones Component Diagram](#)
- [4.13. SPA Component Diagram](#)
- [4.14. Subscriptor Auditoria Component Diagram](#)
- [4.15. Subscriptor Notificacion Component Diagram](#)
- [5. Actors](#)
 - [5.1. Usuarios](#)
- [6. Groups](#)
 - [6.1. BackEnd For FrontEnd](#)
 - [6.2. Eventos Auditoria](#)
 - [6.3. SAGA Orquestacion Cliente Frecuente](#)
- [7. Systems](#)
 - [7.1. Auth0](#)
 - [7.2. Azure Active Directory](#)
 - [7.3. Azure Face API](#)
 - [7.4. Banca OnLine](#)
 - [7.5. Banco Central del Ecuador](#)
 - [7.6. BanRed](#)
 - [7.7. Core](#)
 - [7.8. Core Cliente](#)
 - [7.9. FACE API](#)
 - [7.10. Monitoreo de Microsoft Azure](#)
 - [7.11. Notificador BP](#)
 - [7.12. Notificador Push BP](#)
 - [7.13. Registro Civil](#)
 - [7.14. Sistema de monitoreo en paralelo](#)
- [8. Apps](#)
 - [8.1. Active Directory](#)

- [8.2. API Registro Civil](#)
- [8.3. App Monitoreo Interno](#)
- [8.4. App Movil](#)
- [8.5. Azure Monitor](#)
- [8.6. BackEnd App Movil](#)
- [8.7. BackEnd Banca OnLine](#)
- [8.8. BackEnd SPA](#)
- [8.9. core.bp](#)
- [8.10. Face API](#)
- [8.11. GATEWAY CORE](#)
- [8.12. Micoservicio Cliente](#)
- [8.13. Micoservicio ClienteDetalle](#)
- [8.14. Micoservicio ClienteFrecuente](#)
- [8.15. Micoservicio Cuenta](#)
- [8.16. Micoservicio Movimiento](#)
- [8.17. Micoservicio Notificaciones Push](#)
- [8.18. Micoservicio Producto](#)
- [8.19. Micoservicio Transferencias](#)
- [8.20. MicoservicioNotificaciones](#)
- [8.21. Pago Directo](#)
- [8.22. SPA](#)
- [8.23. SPI](#)
- [8.24. Subscriptor Auditoria](#)
- [8.25. Subscriptor Notificacion](#)
- [9. Stores](#)
 - [9.1. Store](#)
 - [9.2. BDD Auditoria](#)
 - [9.3. Bdd Cliente](#)
 - [9.4. BDD Cliente Frecuente](#)
 - [9.5. BDD Core](#)
- [10. Components](#)

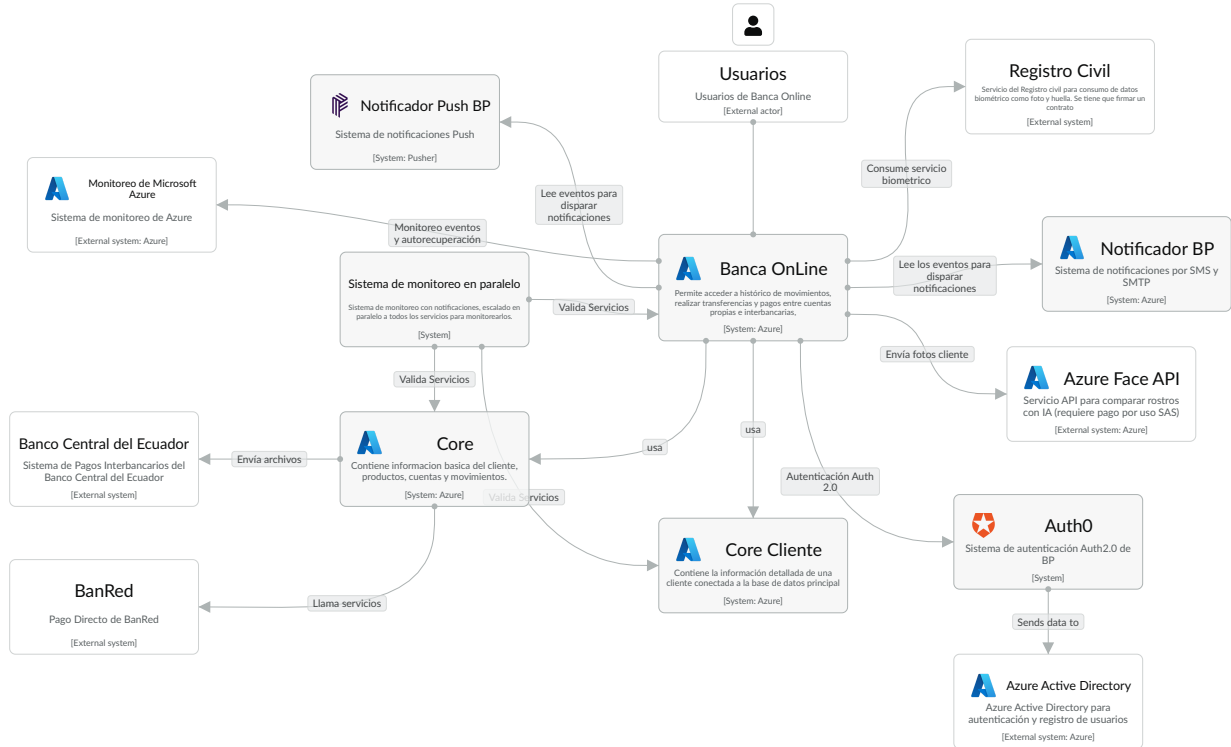
- [10.1. Auditoria Controller](#)
- [10.2. Autenticación](#)
- [10.3. Autenticacion BP Controller](#)
- [10.4. Autenticacion Controller](#)
- [10.5. Autenticacion Movil](#)
- [10.6. Autenticacion Movil Controller](#)
- [10.7. Autenticación Movil Face ID](#)
- [10.8. Autenticacion Movil Huella](#)
- [10.9. Certificados](#)
- [10.10. Certificados BP Controller](#)
- [10.11. Certificados Controller](#)
- [10.12. Certificados Movil](#)
- [10.13. Certificados Movil Controller](#)
- [10.14. Cliente](#)
- [10.15. Cliente](#)
- [10.16. Cliente Controller](#)
- [10.17. Cliente Detalle](#)
- [10.18. Cliente Frecuente Controller](#)
- [10.19. Cliente Movil](#)
- [10.20. Cliente Movil Controller](#)
- [10.21. Clientes BP Controller](#)
- [10.22. Cuentas](#)
- [10.23. Movimiento Controller](#)
- [10.24. Movimiento Movil Controller](#)
- [10.25. Movimientos](#)
- [10.26. Movimientos](#)
- [10.27. Movimientos BP Controller](#)
- [10.28. Movimientos Movil](#)
- [10.29. Notificacion Controller](#)
- [10.30. Onboarding BP Controller](#)
- [10.31. Onboarding Movil Controller](#)
- [10.32. Producto](#)

- [10.33. Producto BP Controller](#)
- [10.34. Producto Controller](#)
- [10.35. Producto Movil Controller](#)
- [10.36. Productos](#)
- [10.37. Productos Movil](#)
- [10.38. SMS Controller](#)
- [10.39. SMTP Controller](#)
- [10.40. Transferencias](#)
- [10.41. Transferencias y Pagos](#)
- [10.42. Transferencias y Pagos BP Controller](#)
- [10.43. Transferencias y Pagos Controller](#)
- [10.44. Transferencias y Pagos Movil](#)
- [10.45. Transferencias y Pagos Movil Controller](#)

1. Context diagrams

1.1. Banca OnLine - Contexto

Context diagram in the Default domain.



2. App diagrams

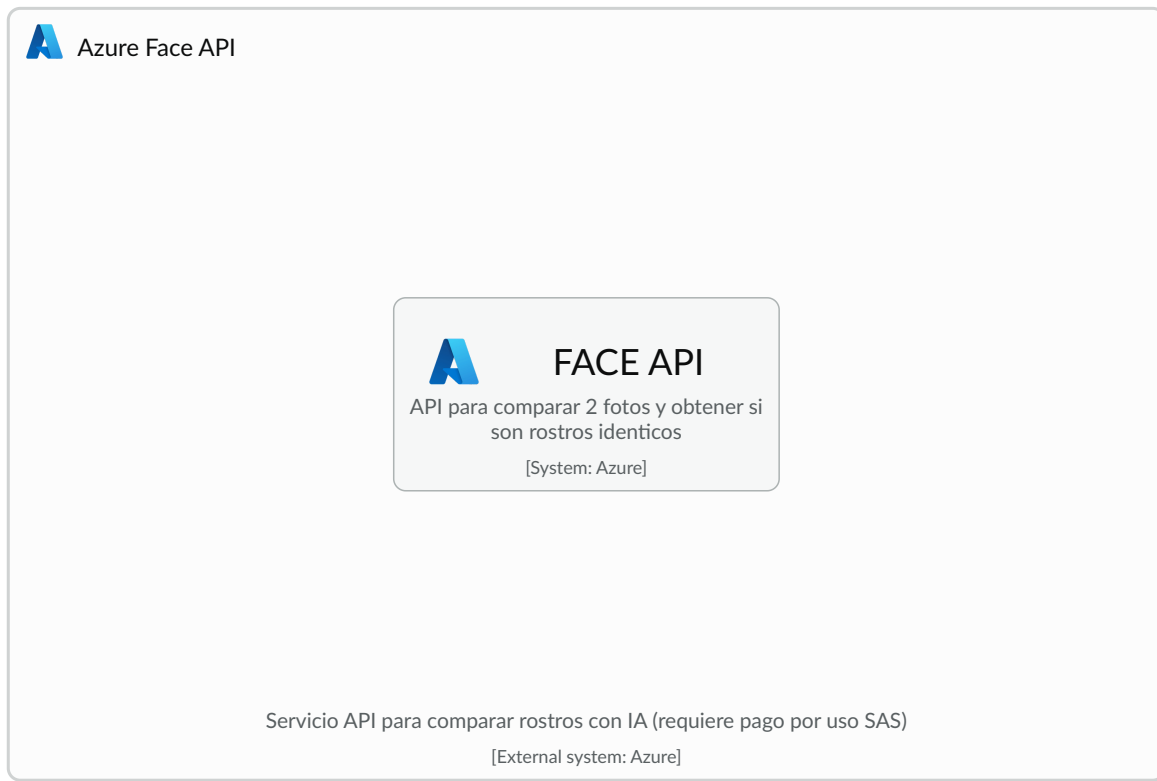
2.1. Azure Active Directory App Diagram

Azure Active Directory app diagram in the Default domain domain.



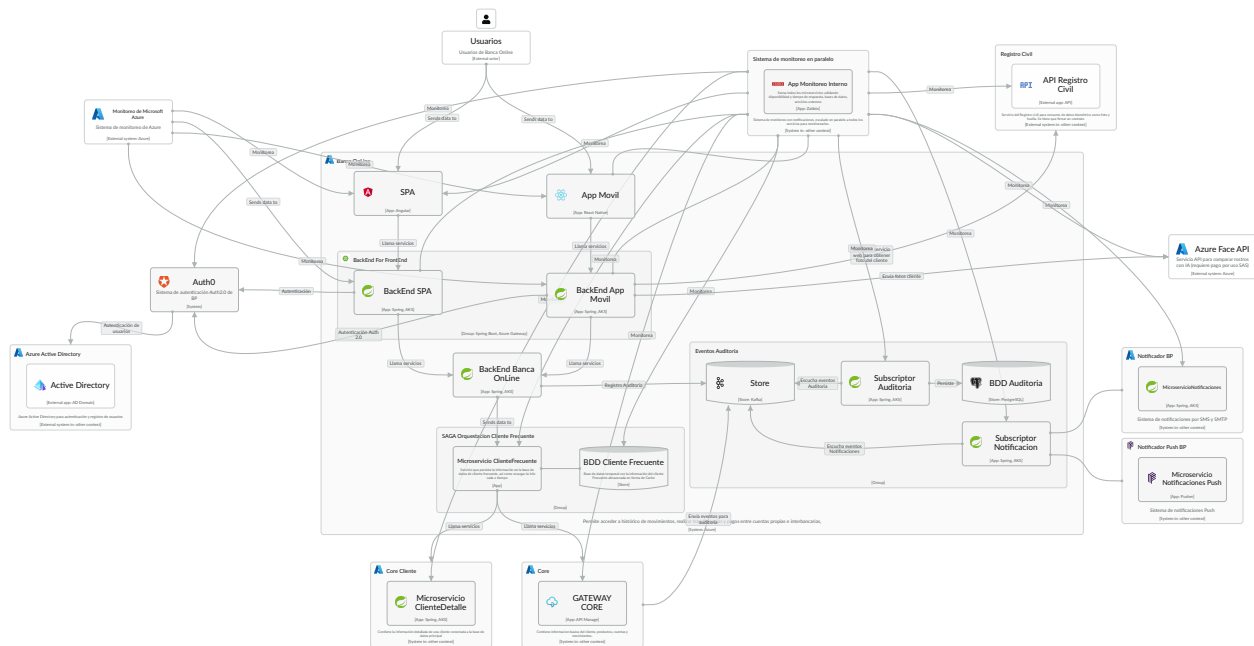
2.2. Azure Face API App Diagram

Azure Face API app diagram in the Default domain domain.



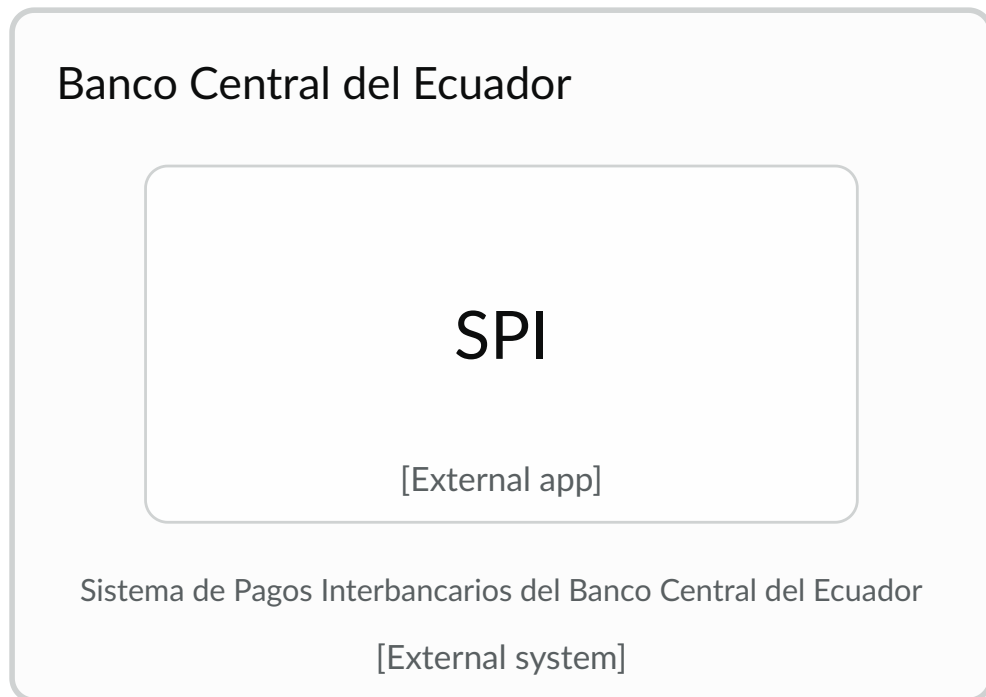
2.3. Banca OnLine App Diagram

Banca OnLine app diagram in the Default domain domain.



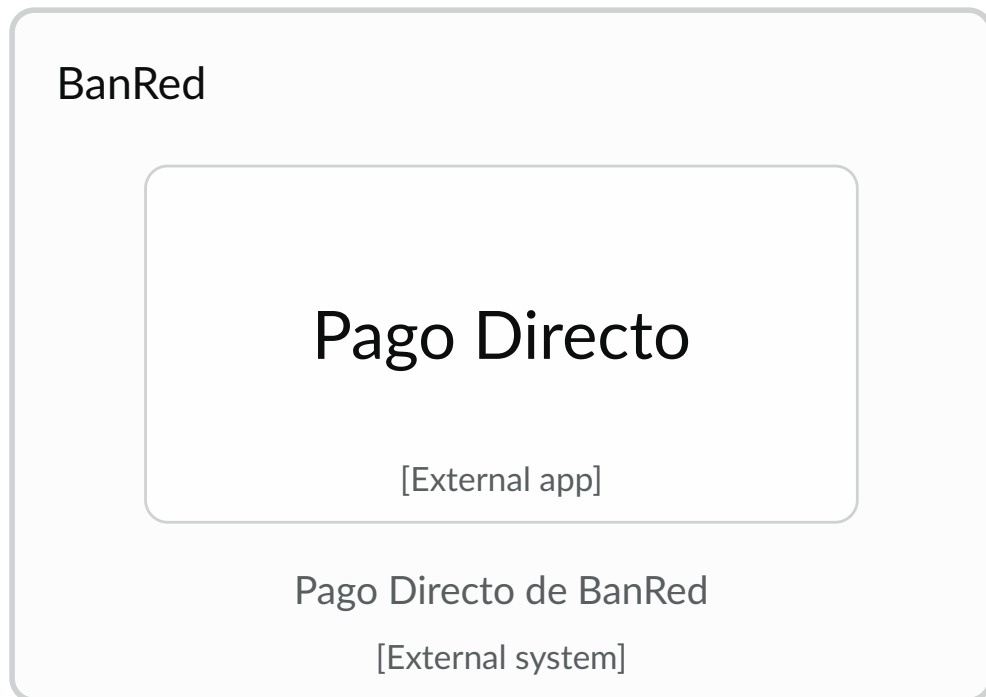
2.4. Banco Central del Ecuador App Diagram

Banco Central del Ecuador app diagram in the Default domain domain.



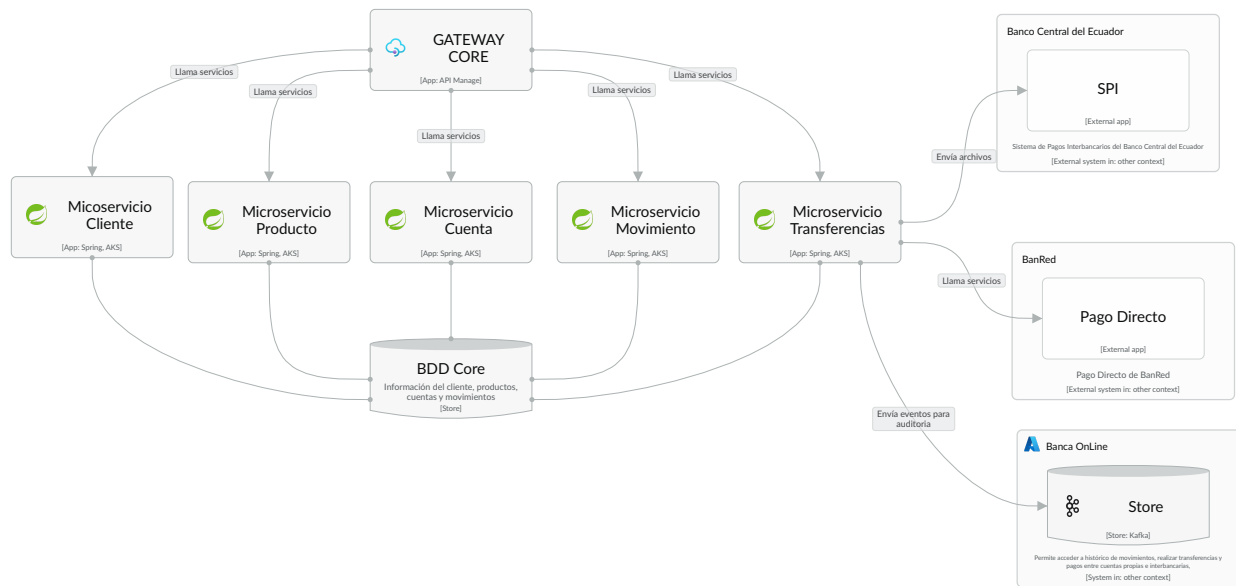
2.5. BanRed App Diagram

BanRed app diagram in the Default domain domain.



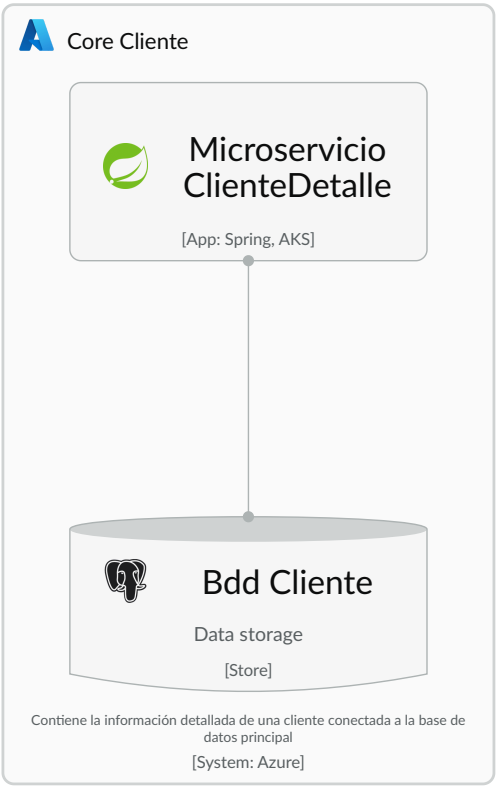
2.6. Core App Diagram

Core app diagram in the Default domain domain.



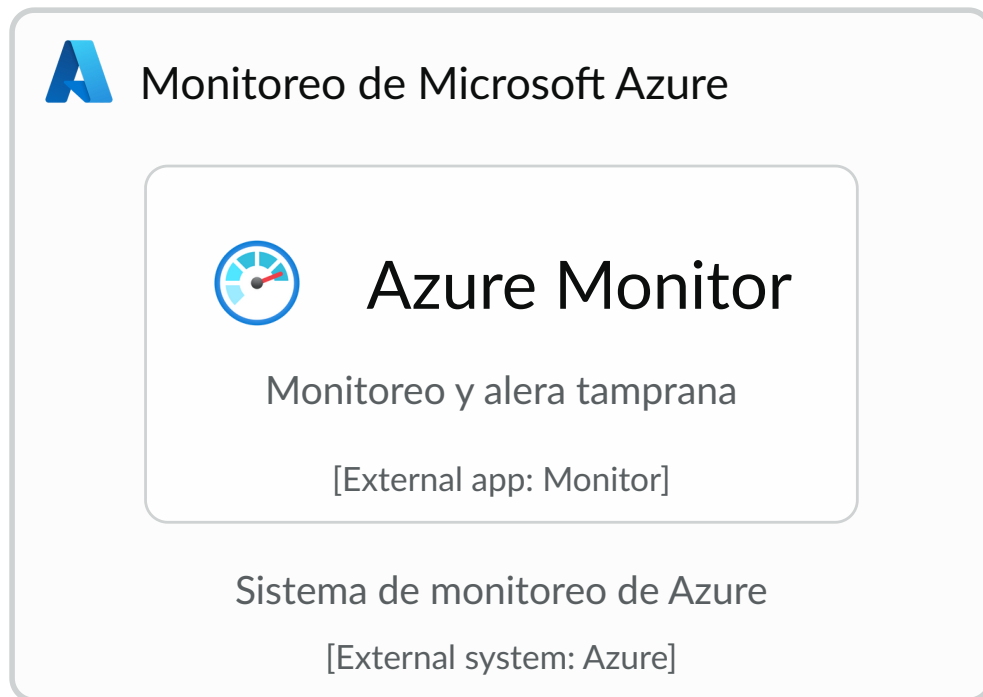
2.7. Core Cliente App Diagram

Core Cliente app diagram in the Default domain domain.



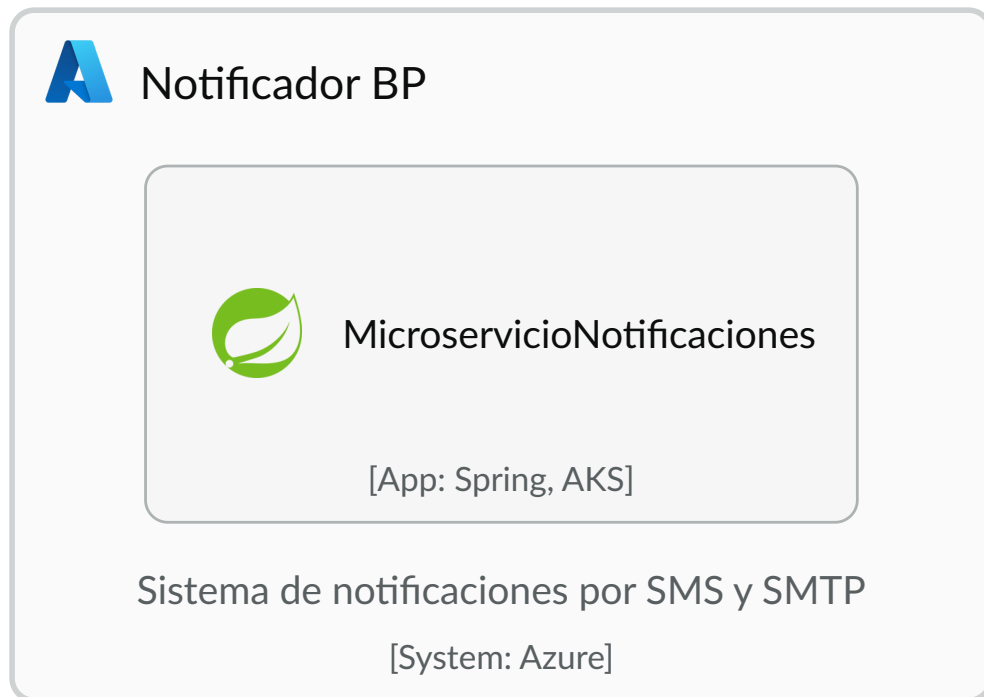
2.8. Monitoreo de Microsoft Azure App Diagram

Monitoreo de Microsoft Azure app diagram in the Default domain domain.



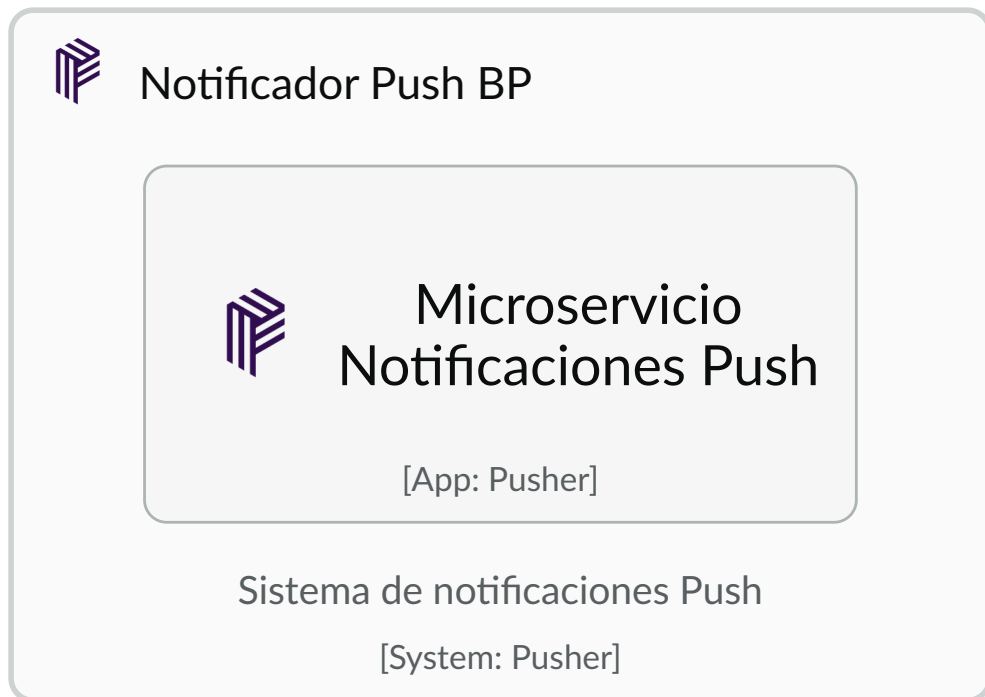
2.9. Notificador BP App Diagram

Notificador BP app diagram in the Default domain domain.



2.10. Notificador Push BP App Diagram

Notificador Push BP app diagram in the Default domain domain.



2.11. Registro Civil App Diagram

Registro Civil app diagram in the Default domain domain.



2.12. Sistema de monitoreo en paralelo App Diagram

Sistema de monitoreo en paralelo app diagram in the Default domain domain.

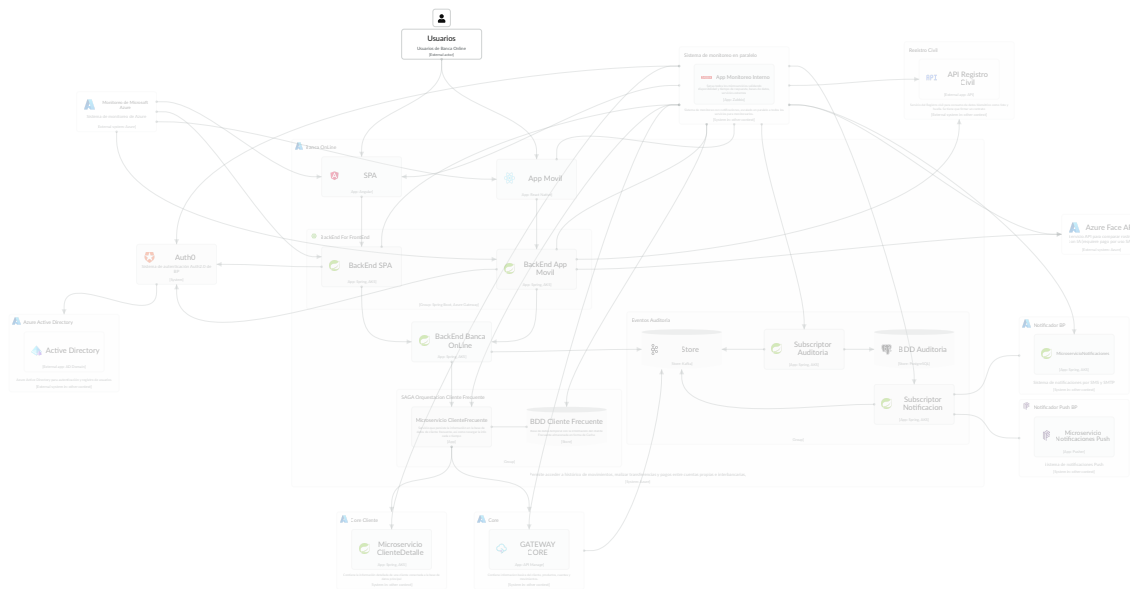


3. App diagram flows

3.1. New customer flow

App diagram flow shown on the Banca OnLine App Diagram diagram in the Default domain domain.

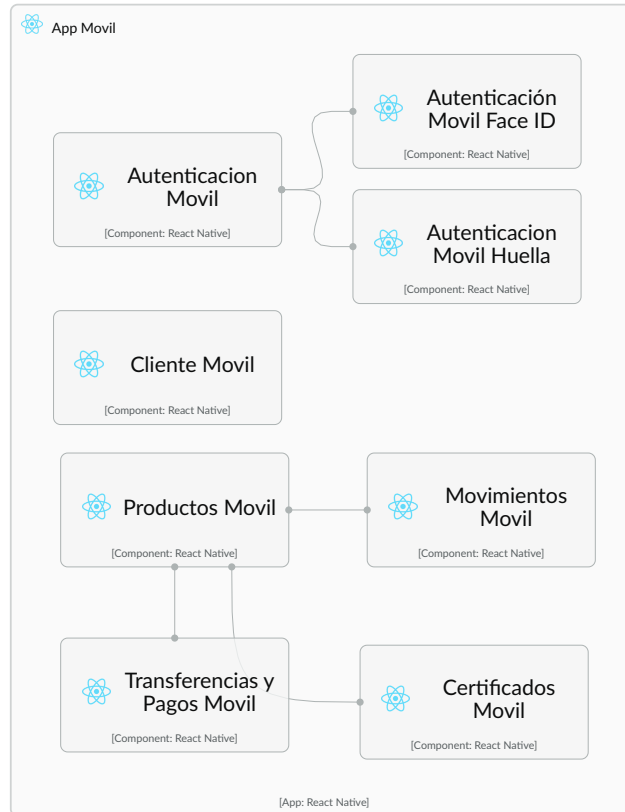
Steps:



4. Component diagrams

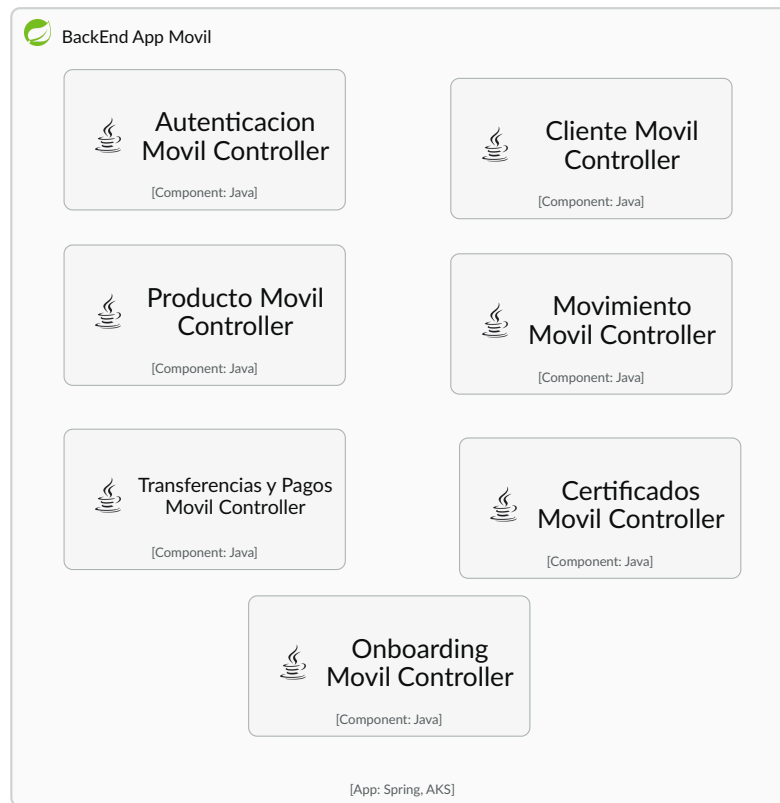
4.1. App Movil Component Diagram

App Movil component diagram in the Default domain.



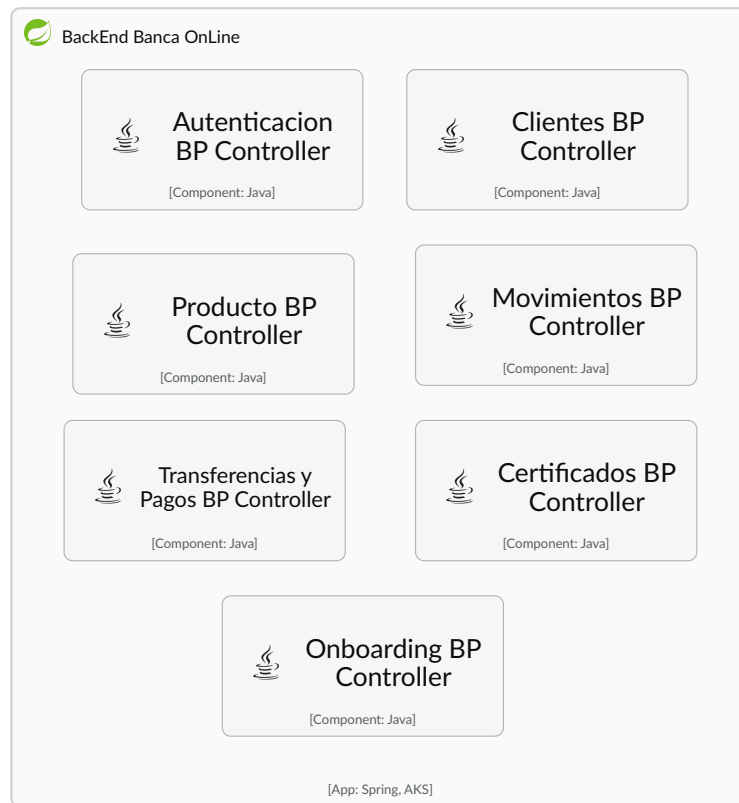
4.2. BackEnd App Movil Component Diagram

BackEnd App Movil component diagram in the Default domain domain.



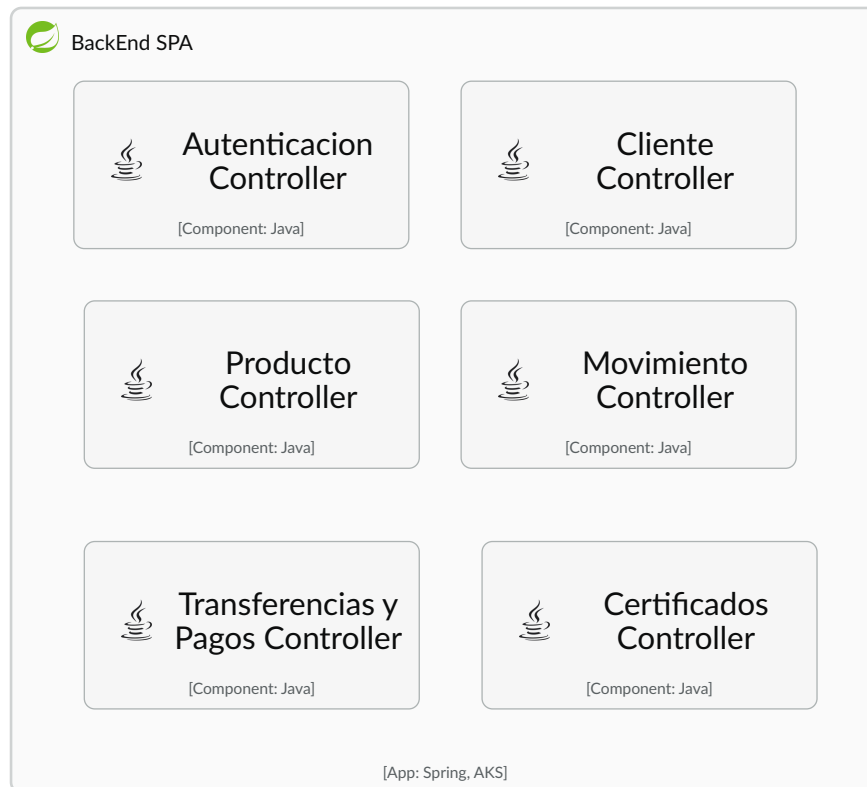
4.3. BackEnd Banca OnLine Component Diagram

BackEnd Banca OnLine component diagram in the Default domain domain.



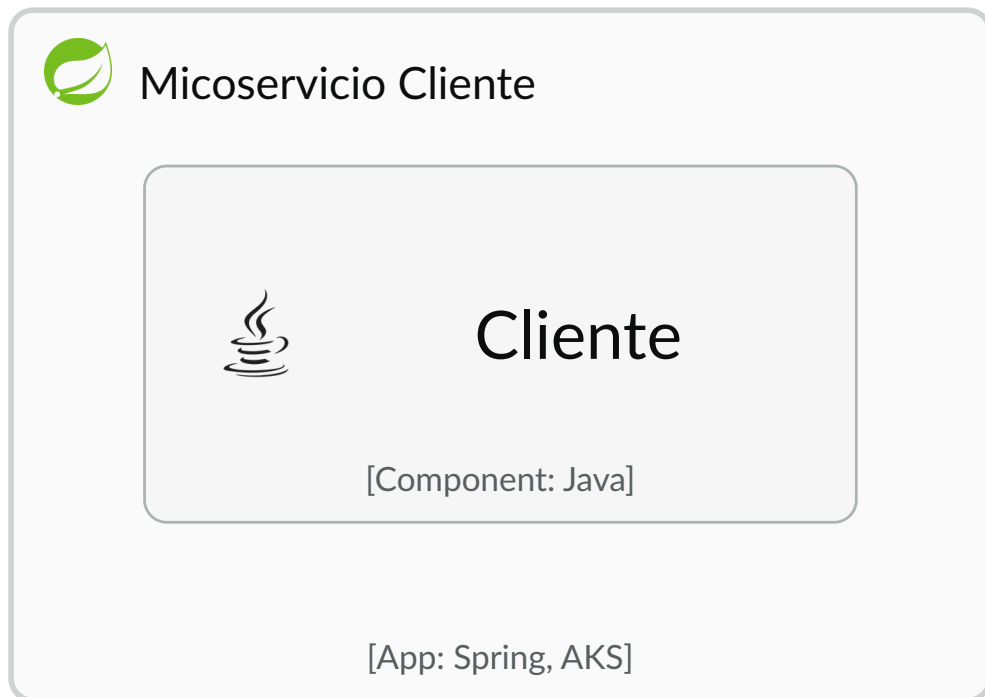
4.4. BackEnd SPA Component Diagram

BackEnd SPA component diagram in the Default domain domain.



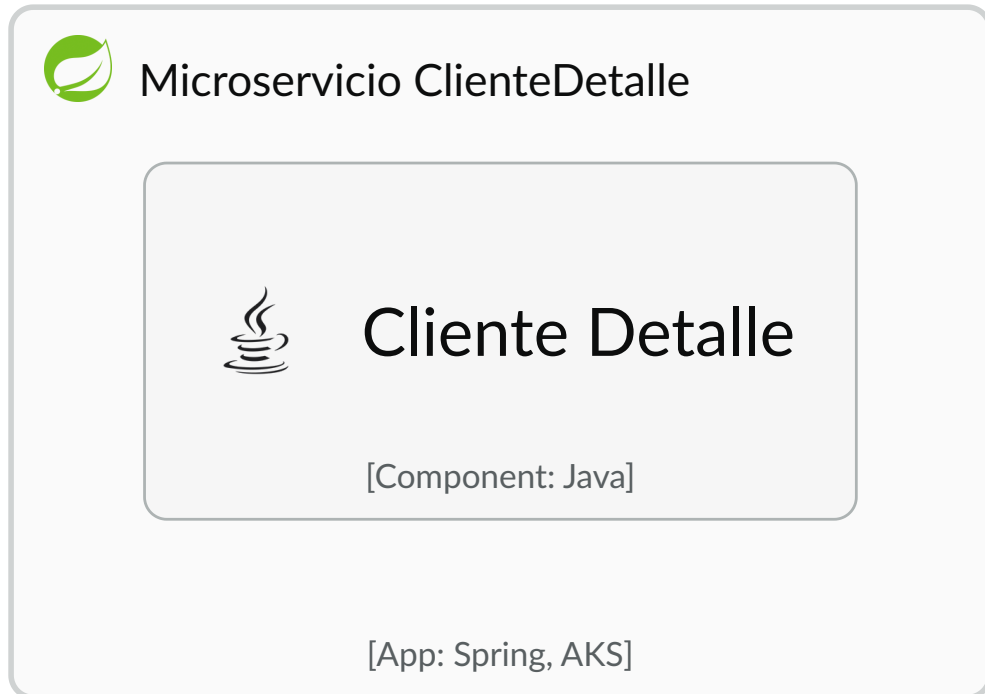
4.5. Micoservicio Cliente Component Diagram

Micoservicio Cliente component diagram in the Default domain domain.



4.6. Microservicio ClienteDetalle Component Diagram

Microservicio ClienteDetalle component diagram in the Default domain domain.



4.7. Microservicio ClienteFrecuente Component Diagram

Microservicio ClienteFrecuente component diagram in the Default domain domain.



4.8. Microservicio Cuenta Component Diagram

Microservicio Cuenta component diagram in the Default domain domain.



4.9. Microservicio Movimiento Component Diagram

Microservicio Movimiento component diagram in the Default domain domain.



4.10. Microservicio Producto Component Diagram

Microservicio Producto component diagram in the Default domain domain.



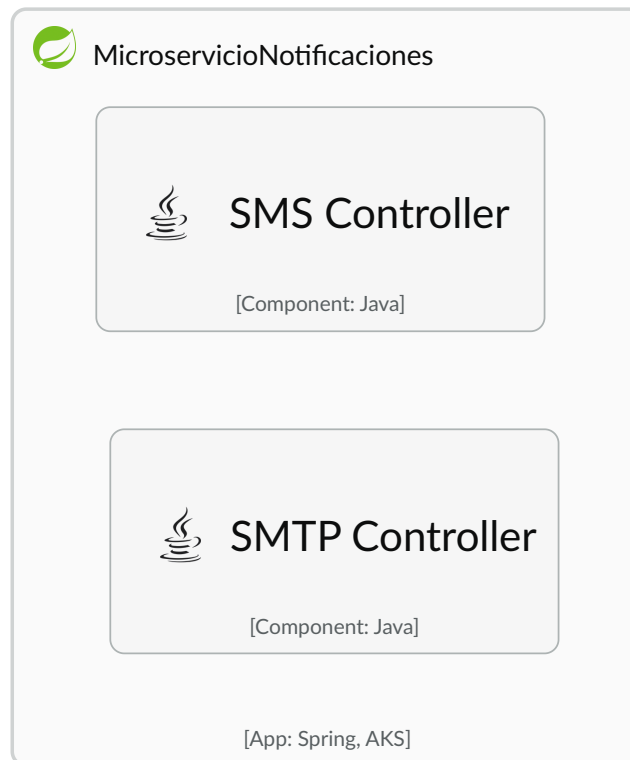
4.11. Microservicio Transferencias Component Diagram

Microservicio Transferencias component diagram in the Default domain domain.



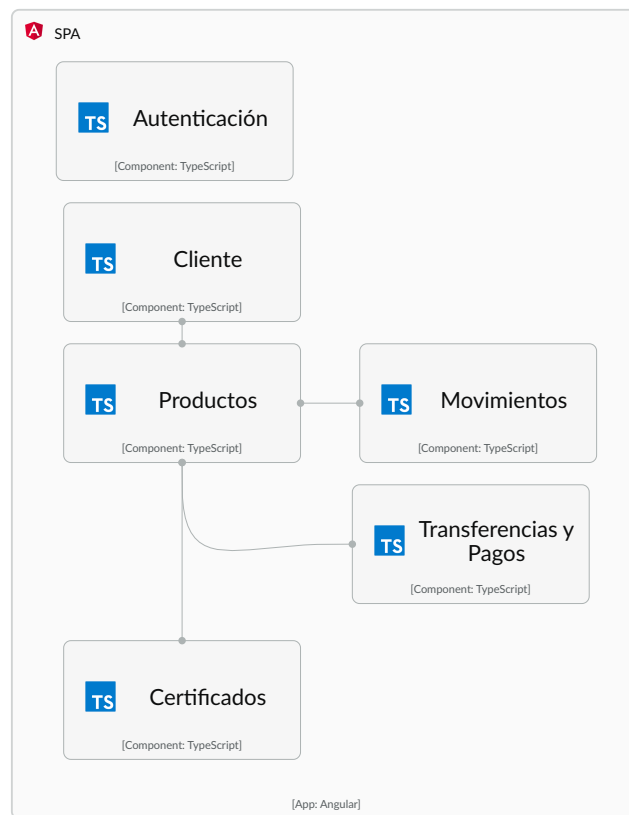
4.12. MicroservicioNotificaciones Component Diagram

MicroservicioNotificaciones component diagram in the Default domain domain.



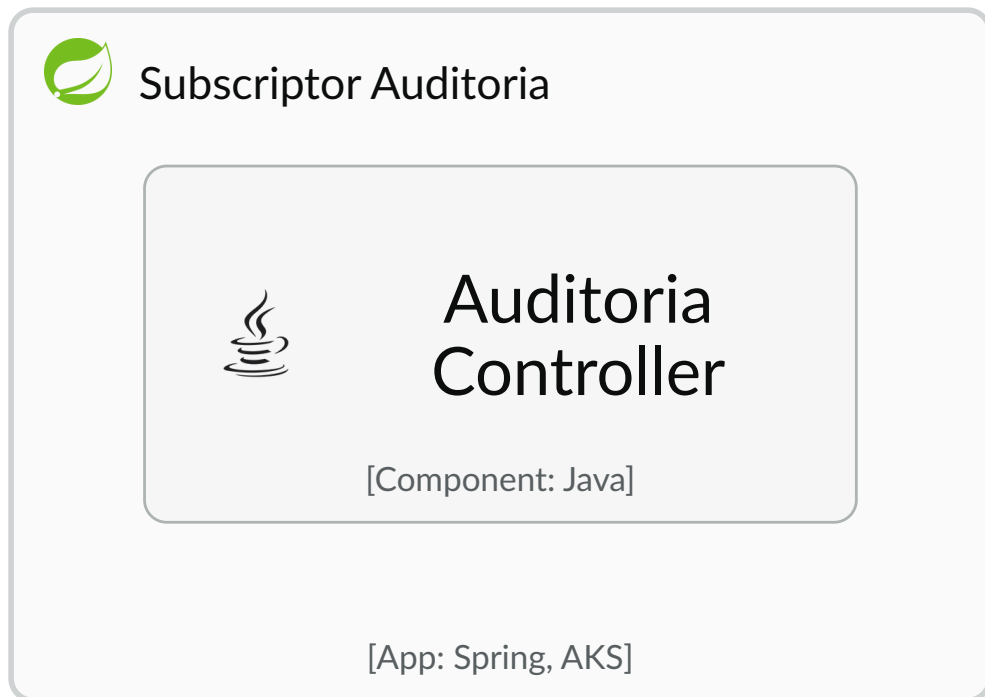
4.13. SPA Component Diagram

SPA component diagram in the Default domain domain.



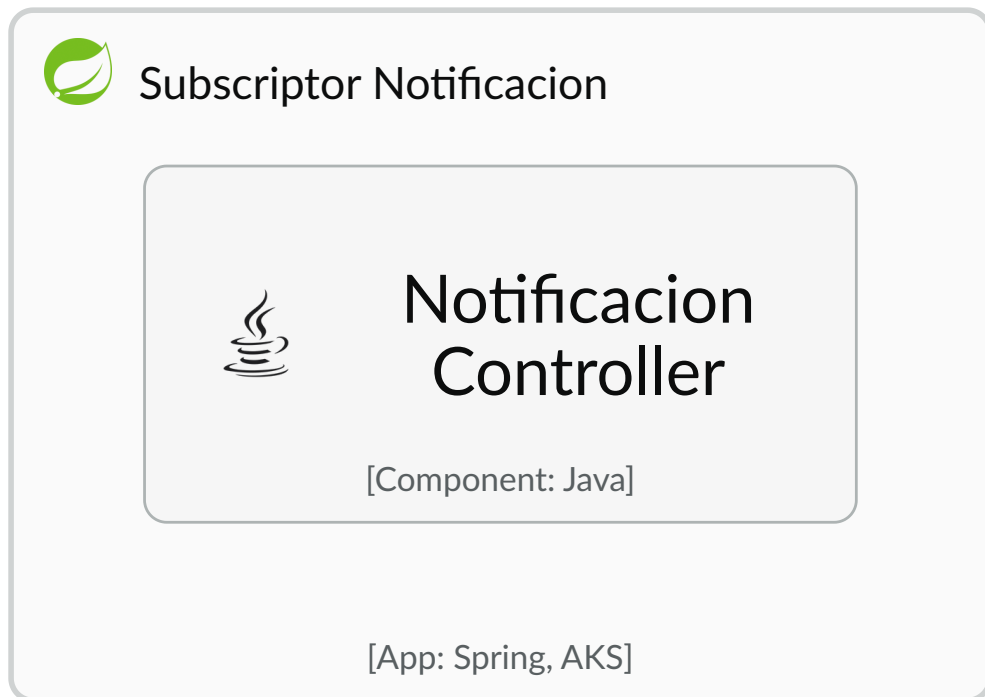
4.14. Subscriptor Auditoria Component Diagram

Subscriptor Auditoria component diagram in the Default domain domain.



4.15. Subscriptor Notificacion Component Diagram

Subscriptor Notificacion component diagram in the Default domain domain.



5. Actors

5.1. Usuarios

External actor in the Default domain domain.

Status: live

Usuarios de Banca Online

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)

In diagrams:

- [New customer flow](#)

6. Groups

6.1. BackEnd For FrontEnd

Internal group in the Default domain domain.

Status: live

Technologies:

- [Spring Boot](#)
- [Azure Application Gateway](#)

In diagrams:

- [Banca OnLine App Diagram](#)

6.2. Eventos Auditoria

Internal group in the Default domain domain.

Status: live

In diagrams:

- [Banca OnLine App Diagram](#)

6.3. SAGA Orquestacion Cliente Frecuente

Internal group in the Default domain domain.

Status: live

In diagrams:

- [Banca OnLine App Diagram](#)

7. Systems

7.1. Auth0

Internal system in the Default domain domain.

Status: live

Sistema de autenticación Auth2.0 de BP

Technologies:

- [Auth0](#)

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)

Descripción

- Manejara la autenticación por MFA, roles y permisos.
- Seguridad contra ataques de bots y fuerza bruta.
- El flujo de código de autorización con PKCE (Authorization Code Flow with PKCE) se usa para evitar el ataque de interceptación de código de autorización podemos utilizar PKCE que amplía el flujo de código de autorización impidiendo [CSRF \(opens new window\)](#). PKCE, pronunciado "pixy" es un acrónimo de *Proof Key for Code Exchange*. Este flujo incluye nuevos elementos PKCE (*code verifier*, *code challenge* y *code challenge method*) en varios pasos del flujo encargados de proteger la comunicación entre el cliente y el servidor de autenticación.

7.2. Azure Active Directory

External system in the Default domain domain.

Status: live

Azure Active Directory para autenticación y registro de usuarios

Technologies:

- [Microsoft Azure](#)

In diagrams:

- [Azure Active Directory App Diagram](#)
- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)

7.3. Azure Face API

External system in the Default domain domain.

Status: live

Servicio API para comparar rostros con IA (requiere pago por uso SAS)

Technologies:

- [Microsoft Azure](#)

In diagrams:

- [Azure Face API App Diagram](#)
- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)

7.4. Banca OnLine

Internal system in the Default domain domain.

Status: live

Permite acceder a histórico de movimientos, realizar transferencias y pagos entre cuentas propias e interbancarias,

Technologies:

- [Microsoft Azure](#)

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)
- [Core App Diagram](#)

A brief description of your system

7.5. Banco Central del Ecuador

External system in the Default domain domain.

Status: live

Sistema de Pagos Interbancarios del Banco Central del Ecuador

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banco Central del Ecuador App Diagram](#)
- [Core App Diagram](#)

7.6. BanRed

External system in the Default domain domain.

Status: live

Pago Directo de BanRed

In diagrams:

- [Banca OnLine - Contexto](#)
- [BanRed App Diagram](#)
- [Core App Diagram](#)

7.7. Core

Internal system in the Default domain domain.

Status: live

Contiene informacion basica del cliente, productos, cuentas y movimientos.

Technologies:

- [Microsoft Azure](#)

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)

7.8. Core Cliente

Internal system in the Default domain domain.

Status: live

Contiene la información detallada de una cliente conectada a la base de datos principal

Technologies:

- [Microsoft Azure](#)

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)
- [Core Cliente App Diagram](#)

7.9. FACE API

Internal system in the Default domain domain.

Status: live

API para comparar 2 fotos y obtener si son rostros identicos

Technologies:

- [Microsoft Azure](#)

In diagrams:

- [Azure Face API App Diagram](#)

7.10. Monitoreo de Microsoft Azure

External system in the Default domain domain.

Status: live

Sistema de monitoreo de Azure

Technologies:

- [Microsoft Azure](#)

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)
- [Monitoreo de Microsoft Azure App Diagram](#)

7.11. Notificador BP

Internal system in the Default domain domain.

Status: live

Sistema de notificaciones por SMS y SMTP

Technologies:

- [Microsoft Azure](#)

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)
- [Notificador BP App Diagram](#)

7.12. Notificador Push BP

Internal system in the Default domain domain.

Status: live

Sistema de notificaciones Push

Technologies:

- [Pusher](#)

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)
- [Notificador Push BP App Diagram](#)

7.13. Registro Civil

External system in the Default domain domain.

Status: live

Servicio del Registro civil para consumo de datos biométrico como foto y huella. Se tiene que firmar un contrato

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)
- [Registro Civil App Diagram](#)

7.14. Sistema de monitoreo en paralelo

Internal system in the Default domain domain.

Status: live

Sistema de monitoreo con notificaciones, escalado en paralelo a todos los servicios para monitorearlos.

In diagrams:

- [Banca OnLine - Contexto](#)
- [Banca OnLine App Diagram](#)
- [Sistema de monitoreo en paralelo App Diagram](#)

8. Apps

8.1. Active Directory

External app in the Default domain domain.

Status: live

Technologies:

- [AD Domain Services](#)

In diagrams:

- [Azure Active Directory App Diagram](#)
- [Banca OnLine App Diagram](#)

8.2. API Registro Civil

External app in the Default domain domain.

Status: live

Technologies:

- [API](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [Registro Civil App Diagram](#)

8.3. App Monitoreo Interno

Internal app in the Default domain domain.

Status: live

Sensa todos los microservicios validando disponibilidad y tiempo de respuesta, bases de datos, servicios externos

Technologies:

- [Zabbix](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [Sistema de monitoreo en paralelo App Diagram](#)

8.4. App Movil

Internal app in the Default domain domain.

Status: live

Technologies:

- [React Native](#)

In diagrams:

- [App Movil Component Diagram](#)
- [Banca OnLine App Diagram](#)

8.5. Azure Monitor

External app in the Default domain domain.

Status: live

Monitoreo y alera temprana

Technologies:

- [Azure Monitor](#)

In diagrams:

- [Monitoreo de Microsoft Azure App Diagram](#)

Description

-

Responsibilities

-

Technical decisions

-

8.6. BackEnd App Movil

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In groups:

- [BackEnd For FrontEnd](#)

In diagrams:

- [BackEnd App Movil Component Diagram](#)
- [Banca OnLine App Diagram](#)

8.7. BackEnd Banca OnLine

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In diagrams:

- [BackEnd Banca OnLine Component Diagram](#)
- [Banca OnLine App Diagram](#)

8.8. BackEnd SPA

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In groups:

- [BackEnd For FrontEnd](#)

In diagrams:

- [BackEnd SPA Component Diagram](#)
- [Banca OnLine App Diagram](#)

8.9. core.bp

Internal app in the Default domain domain.

Status: live

Backend REST API

In diagrams:

- [New customer flow](#)

8.10. Face API

Internal app in the Default domain domain.

Status: live

Technologies:

- [Azure Face](#)

8.11. GATEWAY CORE

Internal app in the Default domain domain.

Status: live

Technologies:

- [API Management Services](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [Core App Diagram](#)

8.12. Micoservicio Cliente

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In diagrams:

- [Core App Diagram](#)
- [Micoservicio Cliente Component Diagram](#)

8.13. Microservicio ClienteDetalle

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [Core Cliente App Diagram](#)
- [Microservicio ClienteDetalle Component Diagram](#)

8.14. Microservicio ClienteFrecuente

Internal app in the Default domain domain.

Status: live

Servicio que persiste la información en la base de datos de cliente frecuente, así como recargar la info cada x tiempo

In groups:

- [SAGA Orquestacion Cliente Frecuente](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [Microservicio ClienteFrecuente Component Diagram](#)

8.15. Microservicio Cuenta

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In diagrams:

- [Core App Diagram](#)
- [Microservicio Cuenta Component Diagram](#)

8.16. Microservicio Movimiento

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In diagrams:

- [Core App Diagram](#)
- [Microservicio Movimiento Component Diagram](#)

8.17. Microservicio Notificaciones Push

Internal app in the Default domain domain.

Status: live

Technologies:

- [Pusher](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [Notificador Push BP App Diagram](#)

8.18. Microservicio Producto

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In diagrams:

- [Core App Diagram](#)
- [Microservicio Producto Component Diagram](#)

8.19. Microservicio Transferencias

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In diagrams:

- [Core App Diagram](#)
- [Microservicio Transferencias Component Diagram](#)

8.20. MicroservicioNotificaciones

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [MicroservicioNotificaciones Component Diagram](#)
- [Notificador BP App Diagram](#)

8.21. Pago Directo

External app in the Default domain domain.

Status: live

In diagrams:

- [BanRed App Diagram](#)
- [Core App Diagram](#)

8.22. SPA

Internal app in the Default domain domain.

Status: live

Technologies:

- [Angular](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [SPA Component Diagram](#)

8.23. SPI

External app in the Default domain domain.

Status: live

In diagrams:

- [Banco Central del Ecuador App Diagram](#)
- [Core App Diagram](#)

8.24. Subscriptor Auditoria

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In groups:

- [Eventos Auditoria](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [Subscriptor Auditoria Component Diagram](#)

8.25. Subscriptor Notificacion

Internal app in the Default domain domain.

Status: live

Technologies:

- [Spring](#)
- [Azure Kubernetes Service \(AKS\)](#)

In groups:

- [Eventos Auditoria](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [Subscriptor Notificacion Component Diagram](#)

9. Stores

9.1. Store

Internal store in the Default domain domain.

Status: live

Technologies:

- [Apache Kafka](#)

In groups:

- [Eventos Auditoria](#)

In diagrams:

- [Banca OnLine App Diagram](#)
- [Core App Diagram](#)

9.2. BDD Auditoria

Internal store in the Default domain domain.

Status: live

Technologies:

- [PostgreSQL](#)

In groups:

- [Eventos Auditoria](#)

In diagrams:

- [Banca OnLine App Diagram](#)

9.3. Bdd Cliente

Internal store in the Default domain domain.

Status: live

Data storage

In diagrams:

- [Core Cliente App Diagram](#)

In diagrams:

- [New customer flow](#)

9.4. BDD Cliente Frecuente

Internal store in the Default domain domain.

Status: live

Base de datos temporal con la información del cliente Frecuente almacenada en forma de Cache

In groups:

- [SAGA Orquestacion Cliente Frecuente](#)

In diagrams:

- [Banca OnLine App Diagram](#)

9.5. BDD Core

Internal store in the Default domain domain.

Status: live

Información del cliente, productos, cuentas y movimientos

In diagrams:

- [Core App Diagram](#)

10. Components

10.1. Auditoria Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [Subscriber Auditoria Component Diagram](#)

10.2. Autenticación

Internal component in the Default domain domain.

Status: live

Technologies:

- [TypeScript](#)

In diagrams:

- [SPA Component Diagram](#)

10.3. Autenticacion BP Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd Banca OnLine Component Diagram](#)

10.4. Autenticacion Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd SPA Component Diagram](#)

10.5. Autenticacion Movil

Internal component in the Default domain domain.

Status: live

Technologies:

- [React Native](#)

In diagrams:

- [App Movil Component Diagram](#)

10.6. Autenticacion Movil Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd App Movil Component Diagram](#)

10.7. Autenticación Movil Face ID

Internal component in the Default domain domain.

Status: live

Technologies:

- [React Native](#)

In diagrams:

- [App Movil Component Diagram](#)

10.8. Autenticacion Movil Huella

Internal component in the Default domain domain.

Status: live

Technologies:

- [React Native](#)

In diagrams:

- [App Movil Component Diagram](#)

10.9. Certificados

Internal component in the Default domain domain.

Status: live

Technologies:

- [TypeScript](#)

In diagrams:

- [SPA Component Diagram](#)

10.10. Certificados BP Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd Banca OnLine Component Diagram](#)

10.11. Certificados Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd SPA Component Diagram](#)

10.12. Certificados Movil

Internal component in the Default domain domain.

Status: live

Technologies:

- [React Native](#)

In diagrams:

- [App Movil Component Diagram](#)

10.13. Certificados Movil Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd App Movil Component Diagram](#)

10.14. Cliente

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [Micoservicio Cliente Component Diagram](#)

10.15. Cliente

Internal component in the Default domain domain.

Status: live

Technologies:

- [TypeScript](#)

In diagrams:

- [SPA Component Diagram](#)

10.16. Cliente Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd SPA Component Diagram](#)

10.17. Cliente Detalle

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [Microservicio ClienteDetalle Component Diagram](#)

10.18. Cliente Frecuente Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [Microservicio ClienteFrecuente Component Diagram](#)

10.19. Cliente Movil

Internal component in the Default domain domain.

Status: live

Technologies:

- [React Native](#)

In diagrams:

- [App Movil Component Diagram](#)

10.20. Cliente Movil Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd App Movil Component Diagram](#)

10.21. Clientes BP Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd Banca OnLine Component Diagram](#)

10.22. Cuentas

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [Microservicio Cuenta Component Diagram](#)

10.23. Movimiento Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd SPA Component Diagram](#)

10.24. Movimiento Movil Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd App Movil Component Diagram](#)

10.25. Movimientos

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [Microservicio Movimiento Component Diagram](#)

10.26. Movimientos

Internal component in the Default domain domain.

Status: live

Technologies:

- [TypeScript](#)

In diagrams:

- [SPA Component Diagram](#)

10.27. Movimientos BP Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd Banca OnLine Component Diagram](#)

10.28. Movimientos Movil

Internal component in the Default domain domain.

Status: live

Technologies:

- [React Native](#)

In diagrams:

- [App Movil Component Diagram](#)

10.29. Notificacion Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [Subscriber Notificacion Component Diagram](#)

10.30. Onboarding BP Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd Banca OnLine Component Diagram](#)

10.31. Onboarding Movil Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd App Movil Component Diagram](#)

10.32. Producto

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [Microservicio Producto Component Diagram](#)

10.33. Producto BP Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd Banca OnLine Component Diagram](#)

10.34. Producto Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd SPA Component Diagram](#)

10.35. Producto Movil Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd App Movil Component Diagram](#)

10.36. Productos

Internal component in the Default domain domain.

Status: live

Technologies:

- [TypeScript](#)

In diagrams:

- [SPA Component Diagram](#)

10.37. Productos Movil

Internal component in the Default domain domain.

Status: live

Technologies:

- [React Native](#)

In diagrams:

- [App Movil Component Diagram](#)

10.38. SMS Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [MicroservicioNotificaciones Component Diagram](#)

10.39. SMTP Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [MicroservicioNotificaciones Component Diagram](#)

10.40. Transferencias

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [Microservicio Transferencias Component Diagram](#)

10.41. Transferencias y Pagos

Internal component in the Default domain domain.

Status: live

Technologies:

- [TypeScript](#)

In diagrams:

- [SPA Component Diagram](#)

10.42. Transferencias y Pagos BP Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd Banca OnLine Component Diagram](#)

10.43. Transferencias y Pagos Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd SPA Component Diagram](#)

10.44. Transferencias y Pagos Movil

Internal component in the Default domain domain.

Status: live

Technologies:

- [React Native](#)

In diagrams:

- [App Movil Component Diagram](#)

10.45. Transferencias y Pagos Movil Controller

Internal component in the Default domain domain.

Status: live

Technologies:

- [Java](#)

In diagrams:

- [BackEnd App Movil Component Diagram](#)