# PRUEBA PRACTICA DEVSU

Nombre: Jorge Luis Renjifo
Ci: 1713149480
PROBLEMA:

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, movimientos, productos y un sistema independiente que complementa la información del cliente cuando los datos se requieren en detalle.

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.

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).

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.

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.

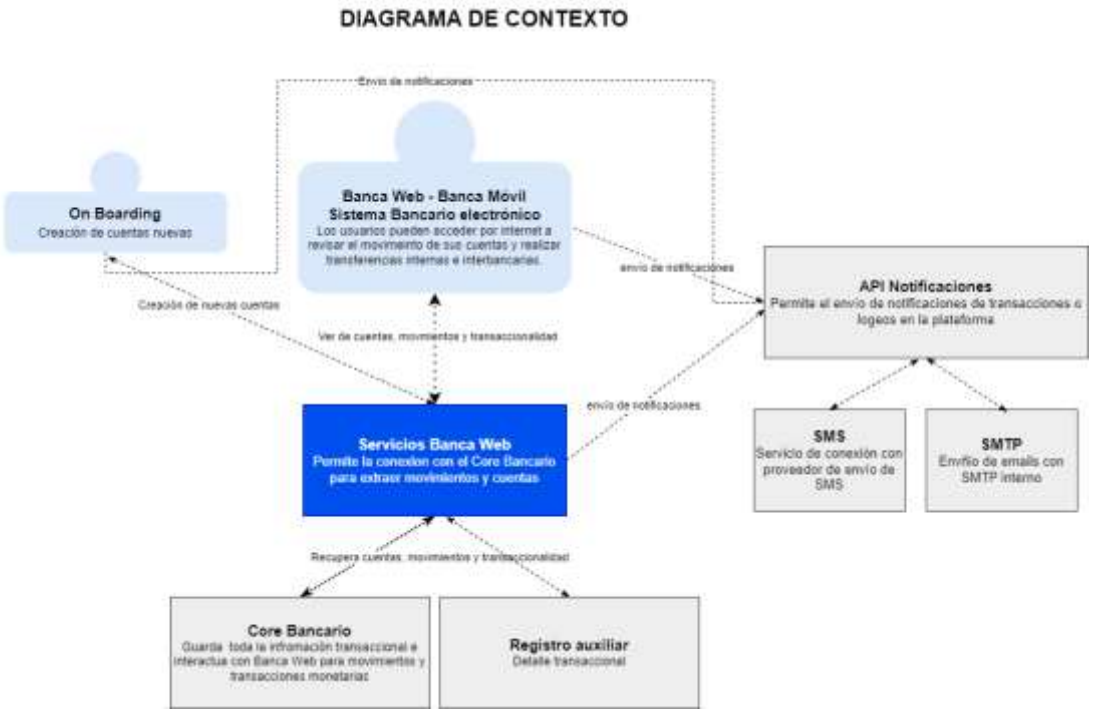
El sistema utiliza una base de datos de auditoría que registra todas las acciones del cliente y cuenta con un mecanismo de persistencia de información para clientes frecuentes, para este caso proponga una alternativa basada en patrones de diseño que relacione los componentes que deberían interactuar para conseguir el objetivo.

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 el rendimiento de su arquitectura o agregar más servicios para mejorar la repuesta de información a sus clientes, es libre de hacerlo.

SOUCION AL PROBLEMA – DIAGRAMAS

Se ha generado mediante un modelo C4 de diagramación los siguientes gráficos que a continuación se explica:

1. MODELO DE CONTEXTO DE LA SOLUCION



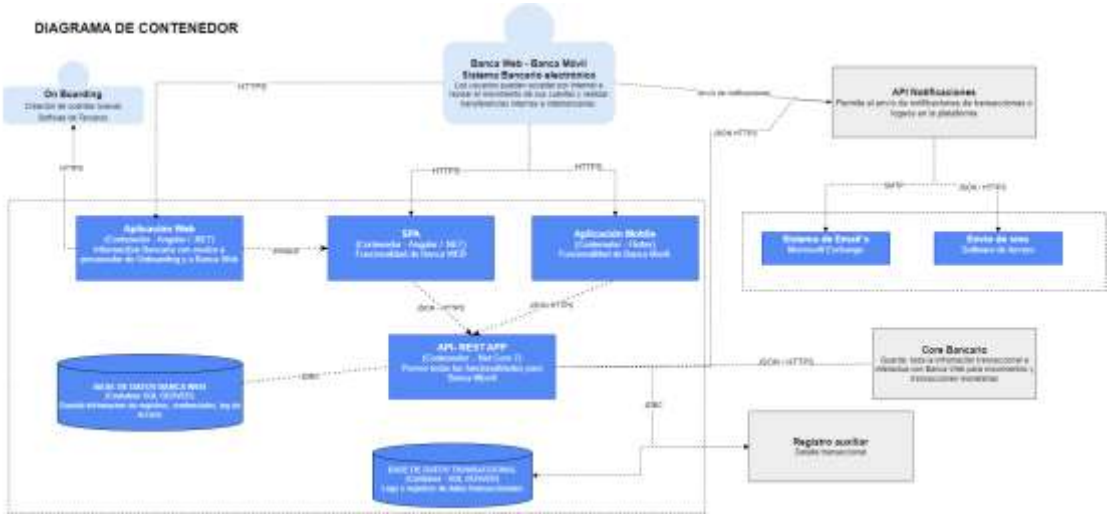

Aquí se puede observar a la Banca electrónica (web y móvil) que es el enlace directo con los usuarios que desean realizar transferencias internas, interbancarias y consultar el saldo de movimientos de sus Cuentas.

Como eje central está un API Gateway de Servicios que permite la interconexión entre la Banca Web con el Core Bancario y un registro auxiliar de auditoria y transaccionalidad.

Hay que tomar en cuenta que hay un servicio de Onboarding que sirve para la creación de cuentas nuevas , si bien esta servicio es de un tercero, interactua con nuestro api Gateway, de servicios de Banca web para controlar, registrar la autenticación, prueba de vida o garantía biométrica del solicitante. Es importante ese registro auxiliar transaccional, tipo auditoria, ya que según la norma de la Super Intendencia de Bancos en Ecuador, esta exige que existe registro y control de todas las identificaciones biométricas al crear cuentas o logearse en banca web con el fin de evitar la suplantación de identidad.

Además, dentro de la norma establece que debe existir un doble factor de autenticación, se crea un motor de notificaciones que funciona independiente de todo la Banca web. Esto con el motivo que el usuario, además de la contraseña, huella biométrica pueda tener como autenticación un token de seguridad que le llega al mail o  por sms al celular registrado.

2. MODELO DE CONTENEDOR




Aquí ya se puede observar desacoplada la solución.

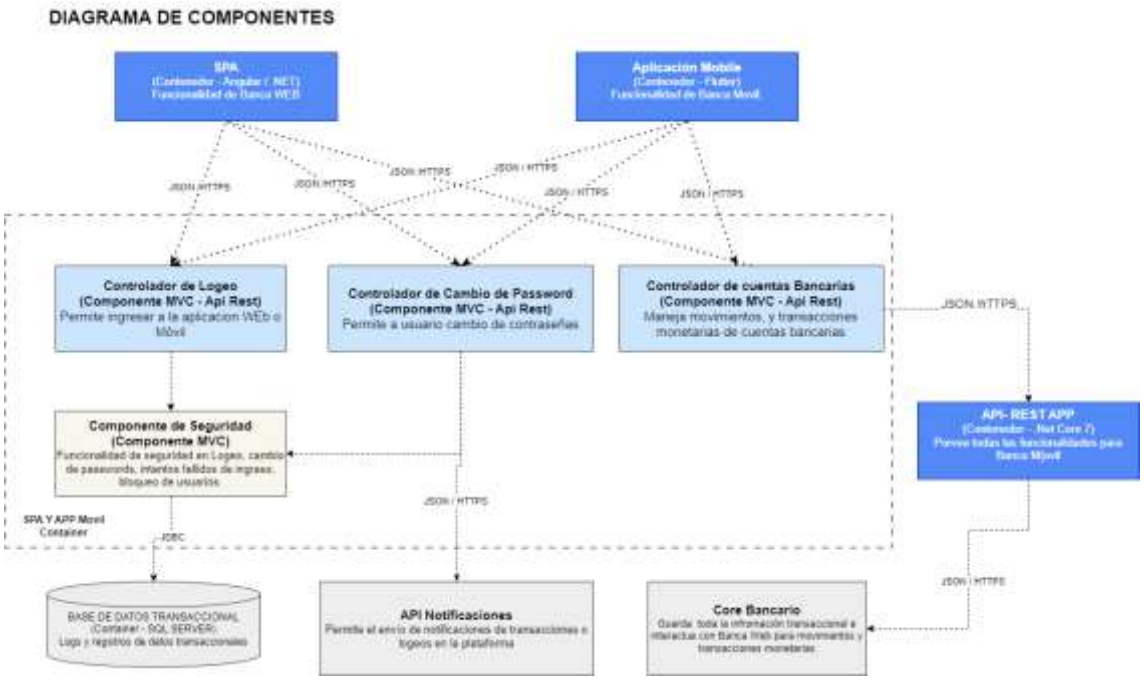
El usuario de Banca electrónica (web o móvil) mediante protocolo https podrá comunicarse con el FRONT SPA de banca web (utilizará Angular y .NET) o la aplicación Móvil, generada en FLUTTER, que nos permite tener versiones para Android y IOS

Se podrá acceder al Front de Banca Web a través de una aplicación WEB (Pagina web informativa) y que también nos puede redirigir hacia el sistema de ONBARDING de un tercero contratado como servicio.

La banca electrónica, se comunica con un API Gateway (REST) realizado con .Net Core tipo Micro servicios (json – https) los cuáles contactarán al Core Bancario para toda la transaccionalidad y grabará su información en una base de datos transaccional y en uno de registro auxiliar de auditoria. Esta conexión se realizará por el respectivo jdbc

Como contenedor individual esta la API de notificaciones que contiene dos servicios, uno de sms que conecta con un servicio terciarizado de envío de mensajes y el de mail que se ocupa de enviar por medio de smtp los correos a los clientes.

3. DIAGRAMA DE COMPONENTES
    a. Banca Electrónica.

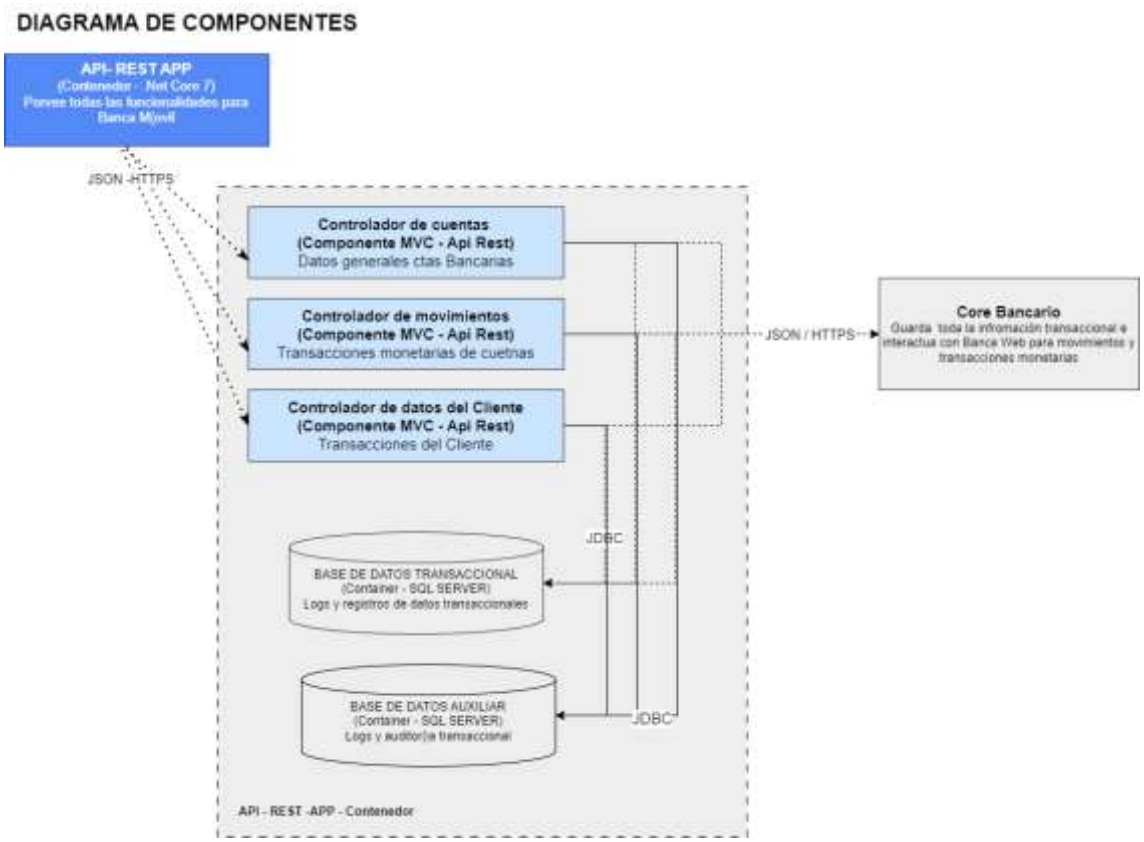



La aplicación web y la app móvil tendrán como componentes, uno de Logeo, otro de cambio de password y otro de cuentas bancarias.

El de Logeo y de manejo de Password se comunicará al componente de seguridad con el fin de transaccionar intentos fallidos, falsificación de identidad, bloqueo de usuarios, logeo, autenticidad, auditoria de ingreso. Y también se comunicarán de forma directa con el motor de notificaciones para manejar procesos de doble factor de autenticidad y comunicaciones mandatorias por la Super Intendencia de Bancos.

Para el proceso transaccional bancario se comunicará con el Api Gateway que esta descrito en el siguiente diagrama.

b. API – REST GATEWAY



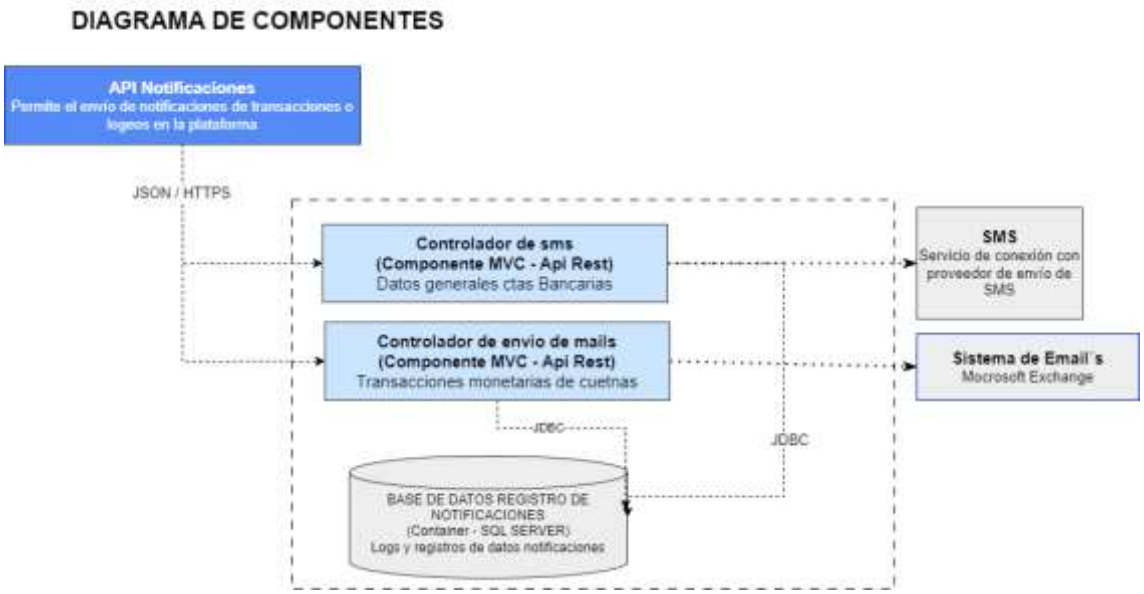

El Componente API REST GATEWAY se comunicará mediante JSON y protocolo HTTPS y tendrá micro servicios de cuentas, movimientos y clientes.

Contará con una base de datos transaccional donde se podrá registra logs transaccionales y registros de negocio propio de Banca Web

Ademas contará con una base de datos de auditoria y logs detallado de las actividades de Banca web que se realizan.

La comunicación con el core Bancario se pretende que sea también por API mediante JSON y HTTPS.

c. API – NOTIFICACIONES



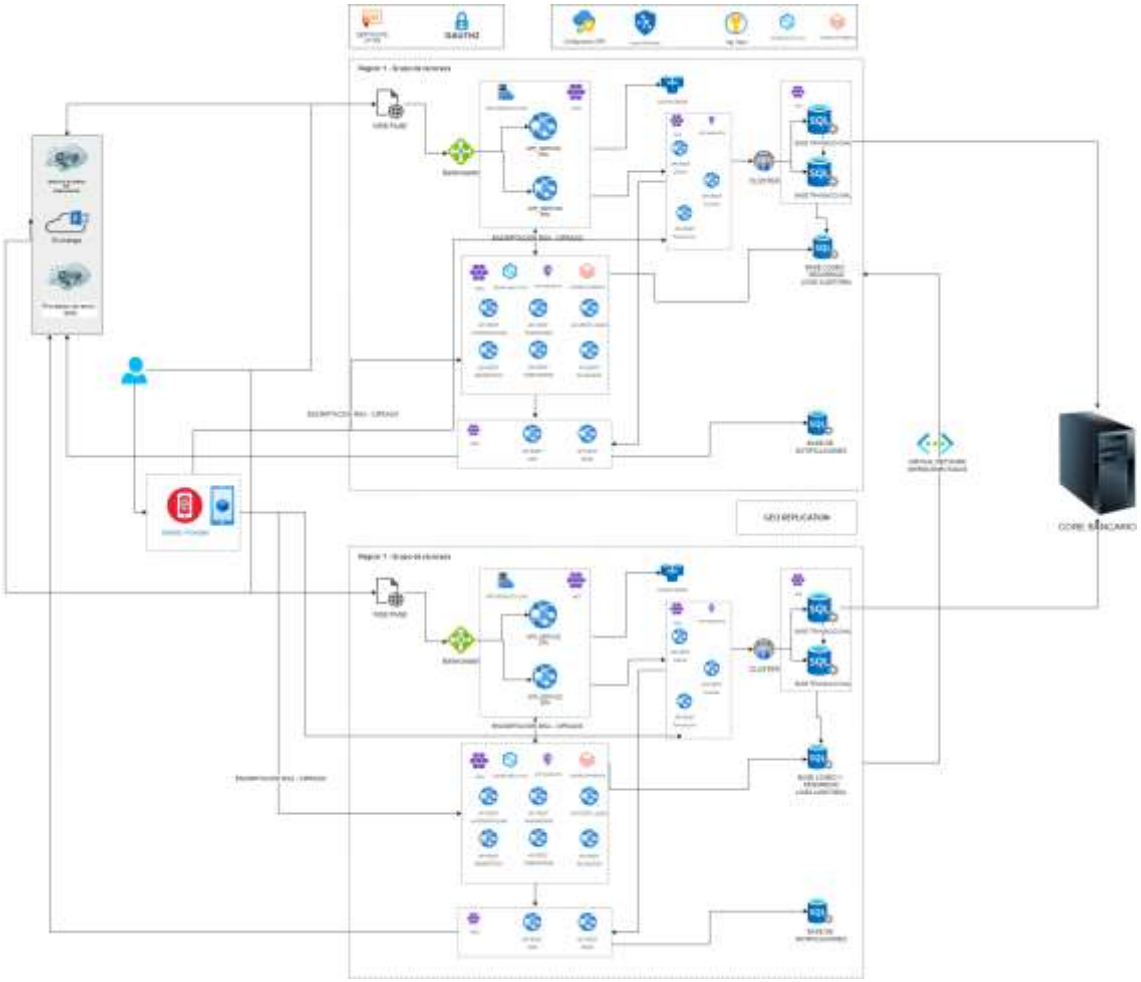
LA APIS de notificaciones es un motor independiente que debe ser llamado de forma asíncrona por sus clientes que lo necesiten y contará con dos controladores.

El controlador de sms, mediante una configuración de plantillas, guardará en la base de datos una cola de mensajes a enviar y un JOB de la base de datos, estará permanentemente en escucha de esta cola para armar la plantilla con los datos del cliente y enviarlos por medio del consumo de una API de un tercero.

En el caso de los emails, también mediante plantillas de acuerdo a quien le consume a este motor, guardará en la base de datos una cola de mensajes a enviar y otro JOB se encargará de enviar por medio de SMPT de Microsoft Exchange los correos a los clientes

4. ARQUITECTURA – INFRAESTRUCTURA

Esta realizado para utilizar la nube de AZURE, mediante un pre proceso de DEVOPS con Integración Continua y despliegue continuo utilizando pipelines que nos permitan desplegar Contenedores utilizando Kubernetes y Docker.



Con el fin de conseguir la respectiva replica de información que se necesita y así también tener procesos de levantamiento de backup contra fallos se propone tener la infraestructura completa desplegada en dos regiones diferentes las cuales deben configurarse para tener una geo replicación conectada estas por una Virtual Network de conexión.

Como especifica la norma de la Súper Intendencia de Banco y en la ISO 27001 los datos personales y transaccionales deben viajar encriptados se utilizará una encriptación de clave RSA de 2048 bits y un hash SHA-256 al generar certificados.

La comunicación entre API, Servicios siempre será mediante JSON y con protocolo https.

Dentro del diagrama se tiene componentes fuera de nuestro alcance, pero se encuentran graficados para conocer el nivel de conexión o dependencia que se tiene con estos.

a. CORE BANCARIO.- Es el core ya adquirido por la institución donde se asentará las transacciones monetarias estarán los datos personales del cliente.
   Cabe indicar que en todo el proceso de manejo de información se utilizará el código del core y toda la información personal estará encriptada como por ejemplo, nombres, cedula, teléfonos, cuentas, direcciones y más.
b. ONBOARDING.- Es un software proporcionado por terceros que se accedería de un enlace WEB y la única relación directa que tiene es con el motor de notificaciones.
c. Microsoft Exchange.- Permite el envío de mail por medio del smtp
d. Envío de SMS.- Software de terceros que permite por medio del consumo de una API el envío de notificaciones a los clientes.

***Descripción de componentes***

**Cache Redis**.- Permite optimizar la respuesta a actividades e información recurrente y está atado a la Banca Web.

**Balanceador**.- Nos permite tener un balanceo de uso para la Banca WEB.

**Cluster**.- Con el fin de tener la mayor tolerancia a fallos se establece un clúster para la base de datos transaccional de Banca Web estableciendo configuraciones de réplica entre activo y pasivo.

**Certificate Https.-** Indica que se generará certificados para las páginas expuestas y para los servicios internos.

**Oauth2.0.-** Autenticación multifactor que se utiliza para la comunicación entre todas las App – web

**Configuration SPA.**- Guarda todas las configuraciones que tiene la Banca Web.

**Key Vault**.- Permite guardar de forma segura las cadenas de conexión, usuarios, contraseñas y url de cada una de las API utilizadas.

**Azure Defender**.- Se utiliza las herramientas proporcionadas por Azure defender para proteger todos nuestro sitios y para monitorear su funcionamiento.

**Logs Insights, Azure Analytics y Databricks.-** Permite tener un registro, conectar y analizar estos datos para dar una respuesta mejor y más rápida al cliente, mediante el uso de BI e IA.

Se puede observar de forma clara los componentes independientes utilizados con AKS para el despliegue de banca web, Micro servicios transaccionales (cuentas, clientes, transacciones (, micro servicios de notificaciones (email, sms),micro servicios de seguridad (Autenticacion, password, logeo, biométrico, on boarding, bloqueos).También se identifica los servidores de Base de datos que son todos SLQ server.