

Diagramas de Arquitectura

Software BP Banking

Diagrama de Contexto

El diagrama de contexto muestra un nivel de abstracción a alto nivel. Los clientes acceden directamente a la solución de banca por internet para atender sus necesidades.

El sistema se encuentra alojado en un servidor web de alta capacidad que le da el soporte requerido. Asimismo envía mensajes de notificación a un sistema externo que se encargará de su propagación. Para lograr la atención requerida se conecta a tres sistemas: de datos ampliados, de core banking y de auditoría.

Para la autenticación de usuarios se hace uso del software NA-AT que se especializa en servicios de cloud recognition, garantizando la identidad de los mismos. El servicio Amazon Simple Notificator se encargará de la entrega de mensajes SMS mientras que la plataforma EmBlue podrá hacerse cargo de la mensajería por correo.

Audiencia: Stakeholders, Product Owners, Architects, Developers, Technical Support

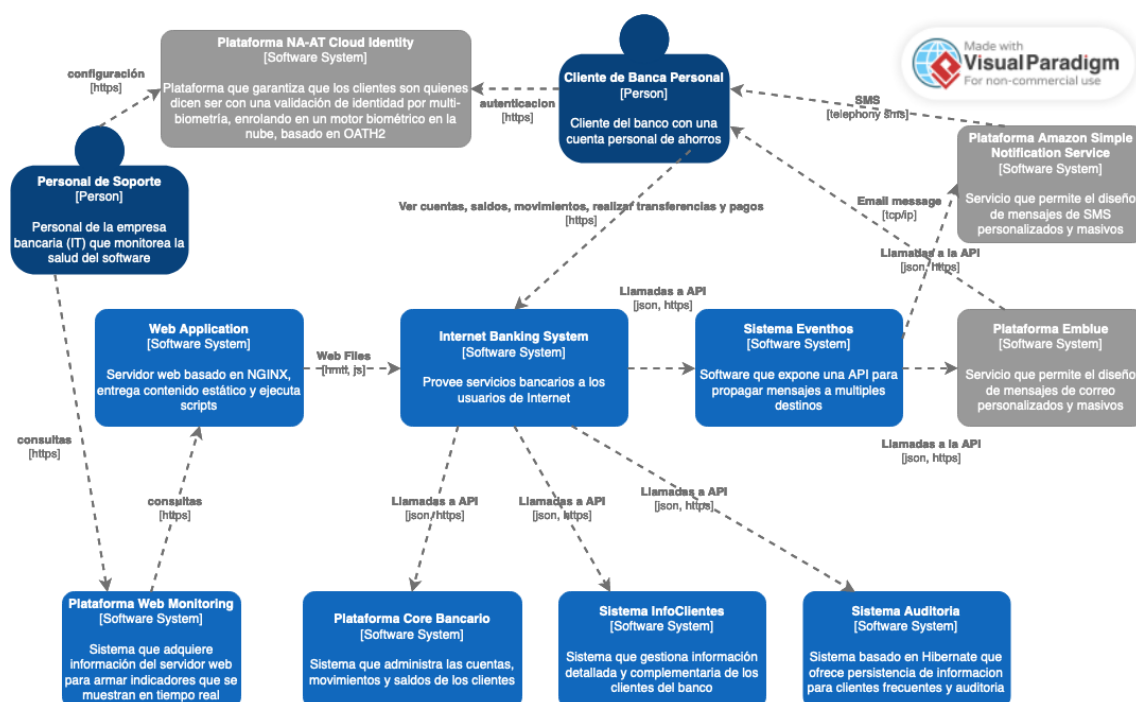


Diagrama de Contenedores

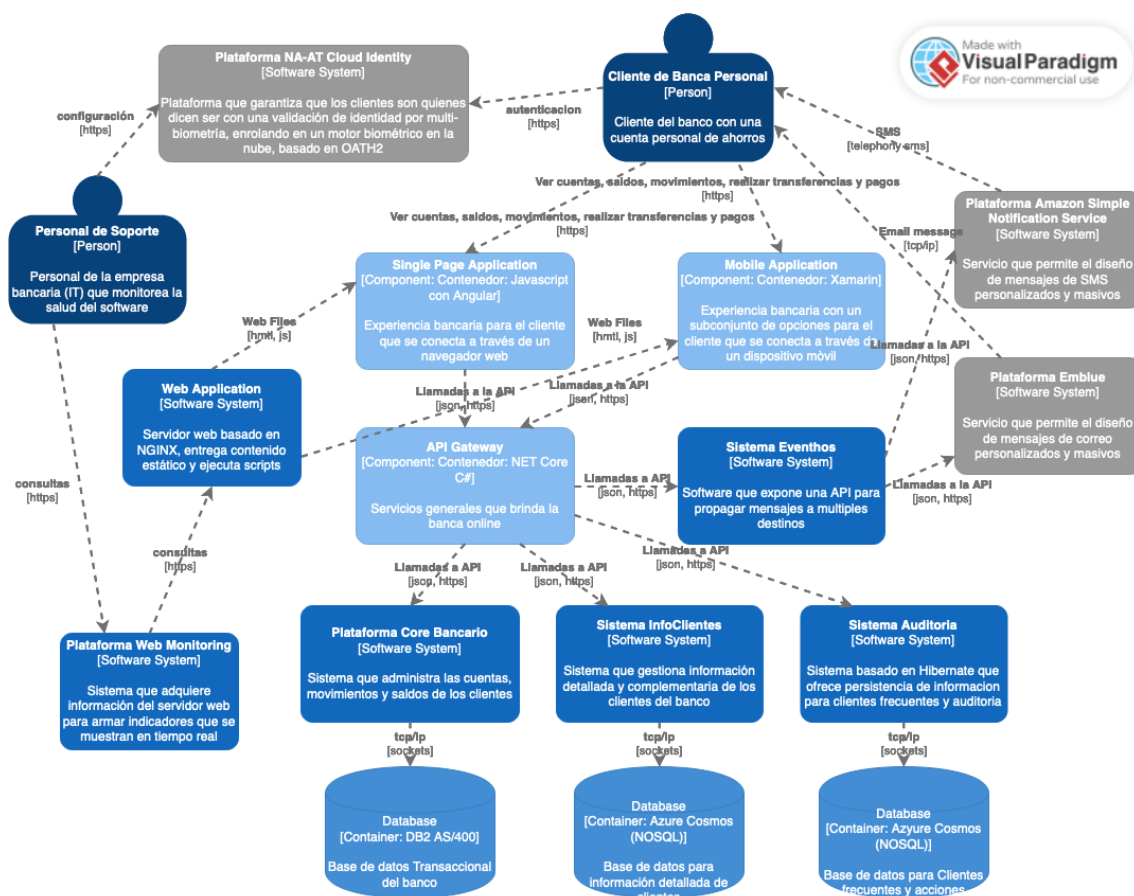
En este segundo nivel de los diagramas C4 podemos ver los tres principales componentes de la Banca por Internet: La aplicación SPA está hecha con Javascript y Angular, mientras que la aplicación mobile está hecha con Xamarin.

El motivo de elegir Xamarin es porque el código .NET que usa puede ser deployado en código nativo en las diferentes plataformas móviles como Android y iOS con el mismo esfuerzo.

Ambos consumen el API Gateway REST realizado con NET Core el cual podrá consumir los servicios de la Banca por Internet. Éste se encarga de propagar las notificaciones al sistema Eventhos que puede redistribuir un mismo mensaje a varios proveedores de mensajería, como es el caso de Amazon o Emblue.

Del otro lado tenemos el Sistema de Web Monitoring que se encarga de consumir los eventos generados por el servidor Web y los transforma en gráficas de rendimiento y reportes de posibles fallos en la latencia o rendimiento de la aplicación.

Se eligen las bases de datos NOSQL para las consultas de cuentas y movimientos puesto que son más veloces para las operaciones de lectura. El core bancario se mantiene en COBOL con AS/400.



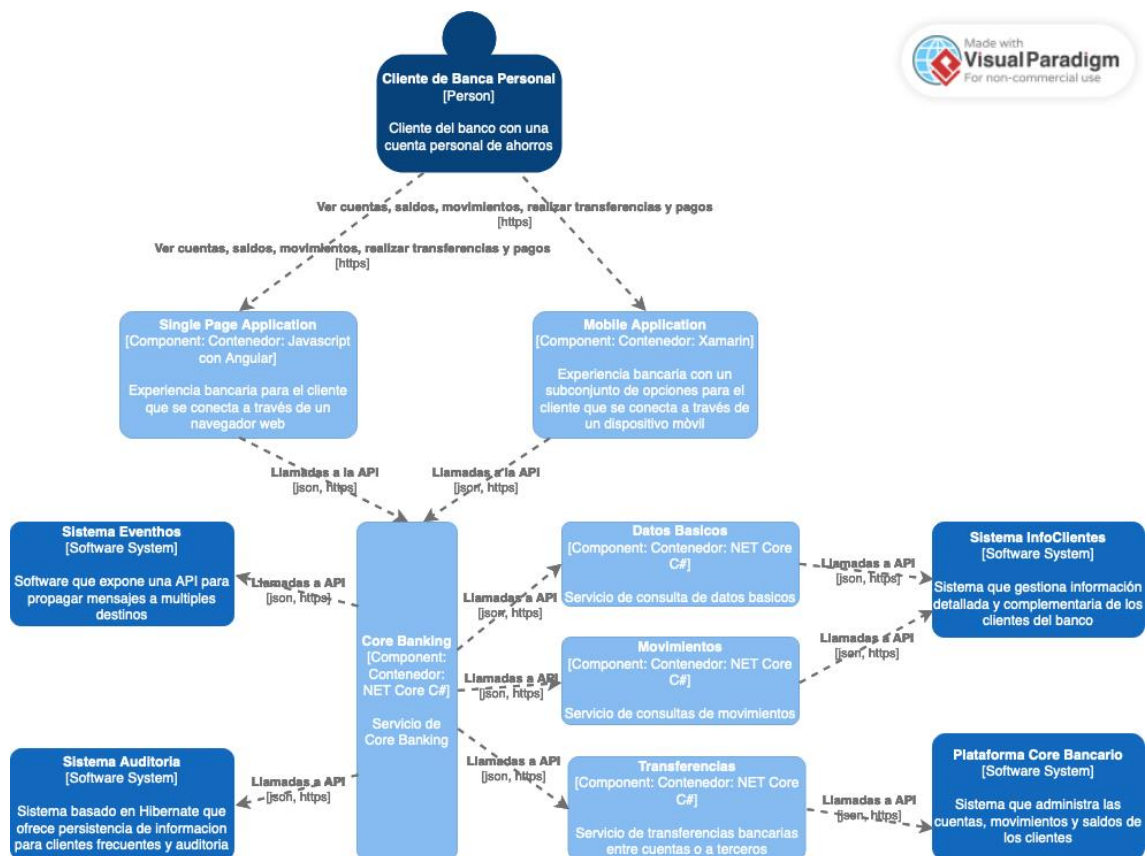
Audiencia: Architects, Developers, Technical Support

Diagrama de Componentes

En este nivel podemos observar que los servicios se empiezan a dividir para atacar diferentes temas que conciernen a la atención del usuario, en especial la división por consultas, movimientos o transferencias.

Lo importante es que pasan por el Core banking API el cual actúa como una especie de “Service Bus” para redirigir el pedido al módulo correspondiente según sea el caso, además por ser la única puerta hacia el resto de servicios, es el lugar idóneo para colocar la grabación de registros de auditoría o notificaciones.

Audiencia: Architects, Developers, Technical Support



Ruta del documento en GIT HUB:

<https://github.com/alessandrochipulina/api/C4-Arquitectura.pdf>