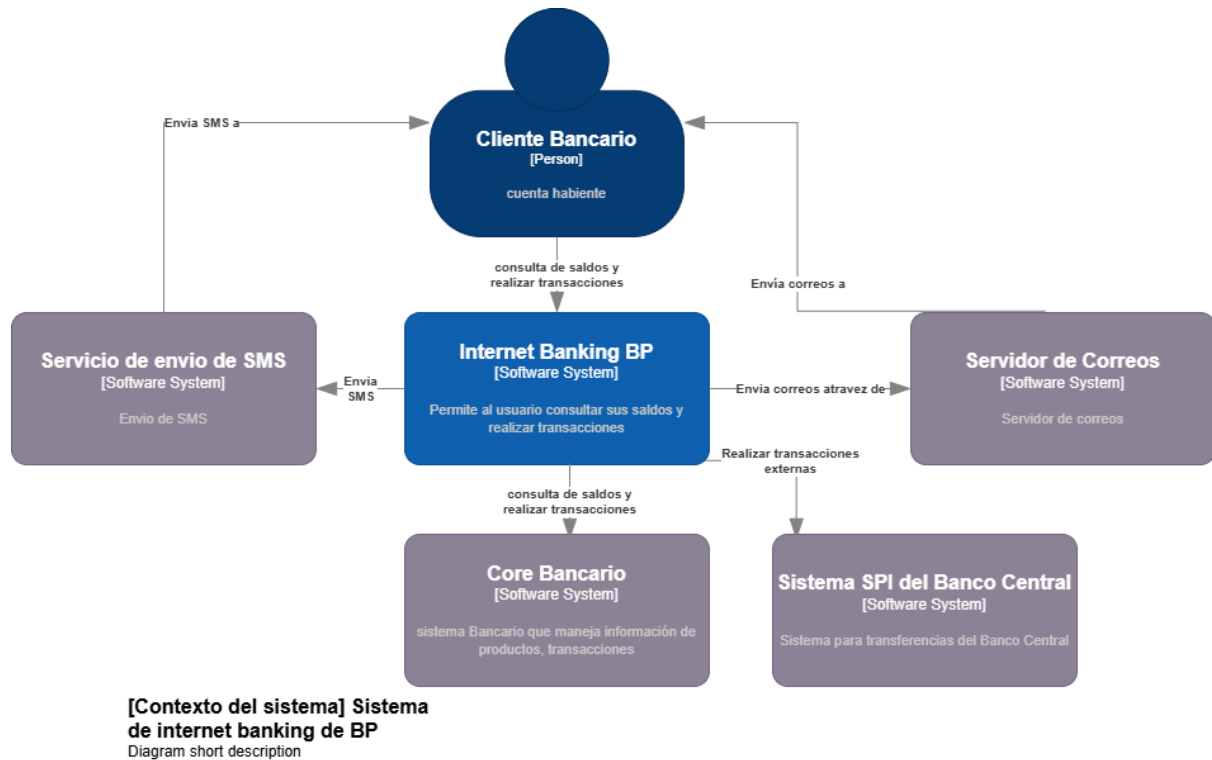
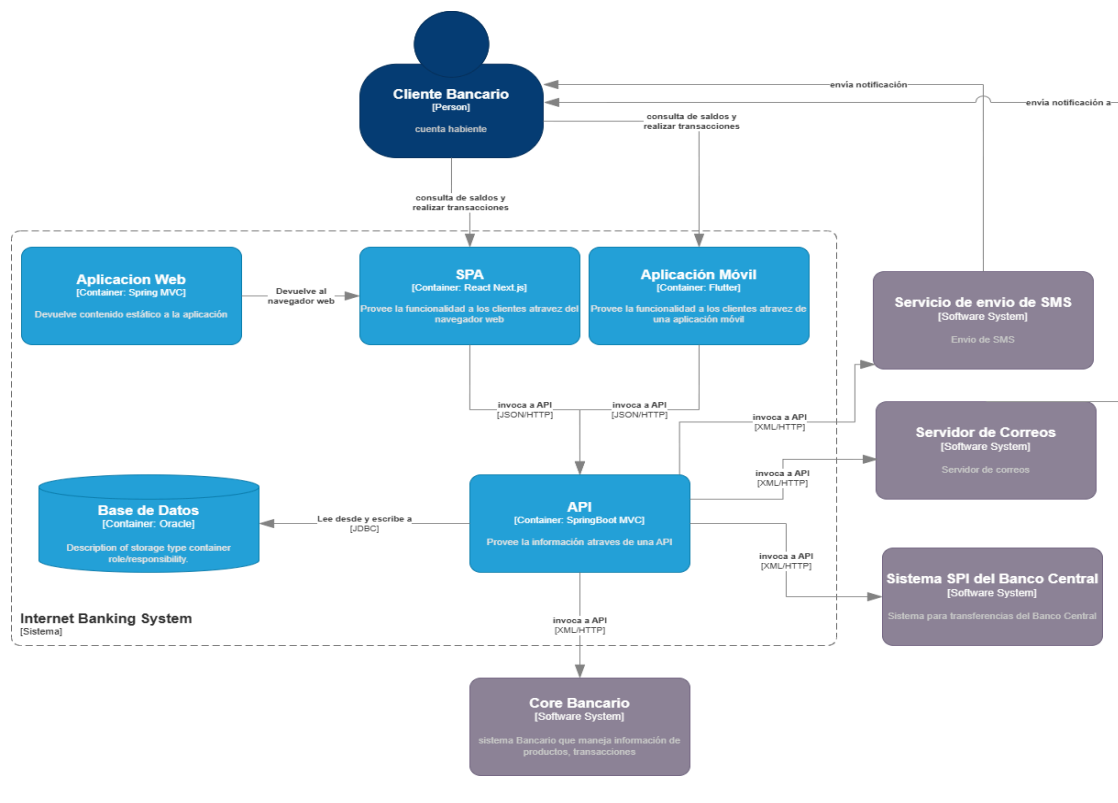


## Diagramas C4

### Diagramas de Contexto

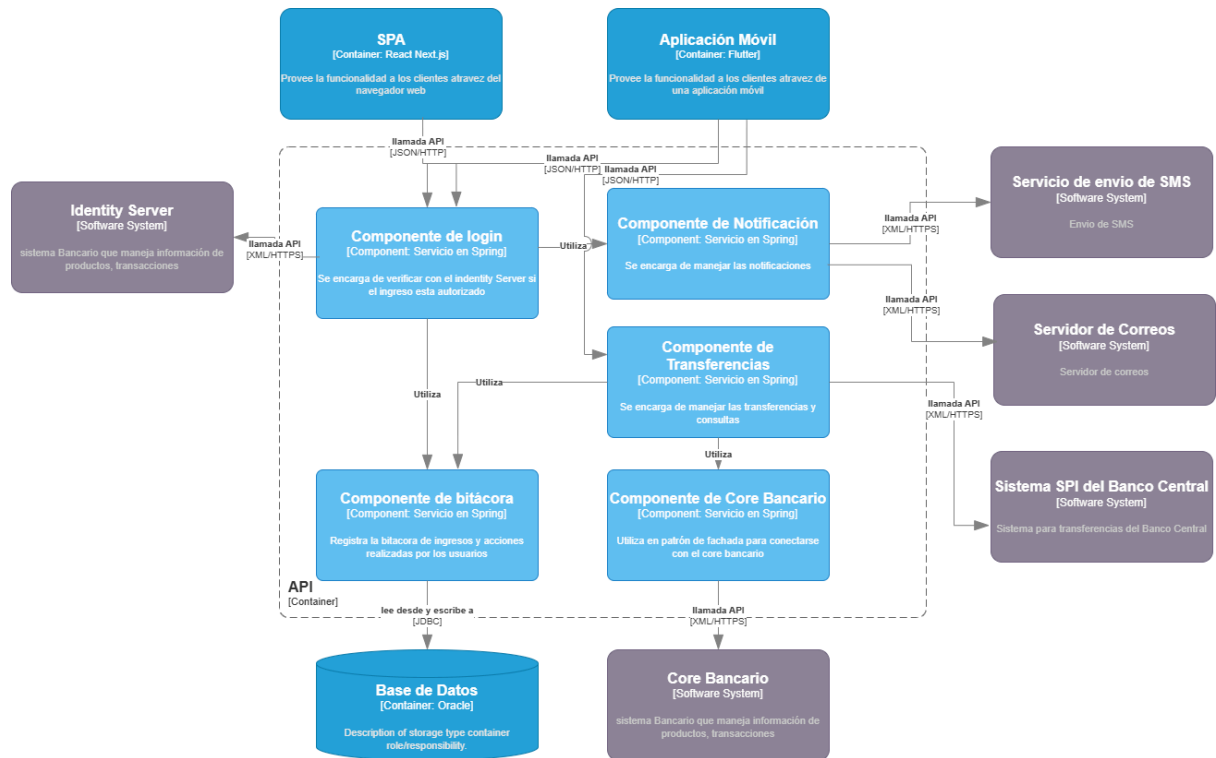


### Diagrama de contenedores

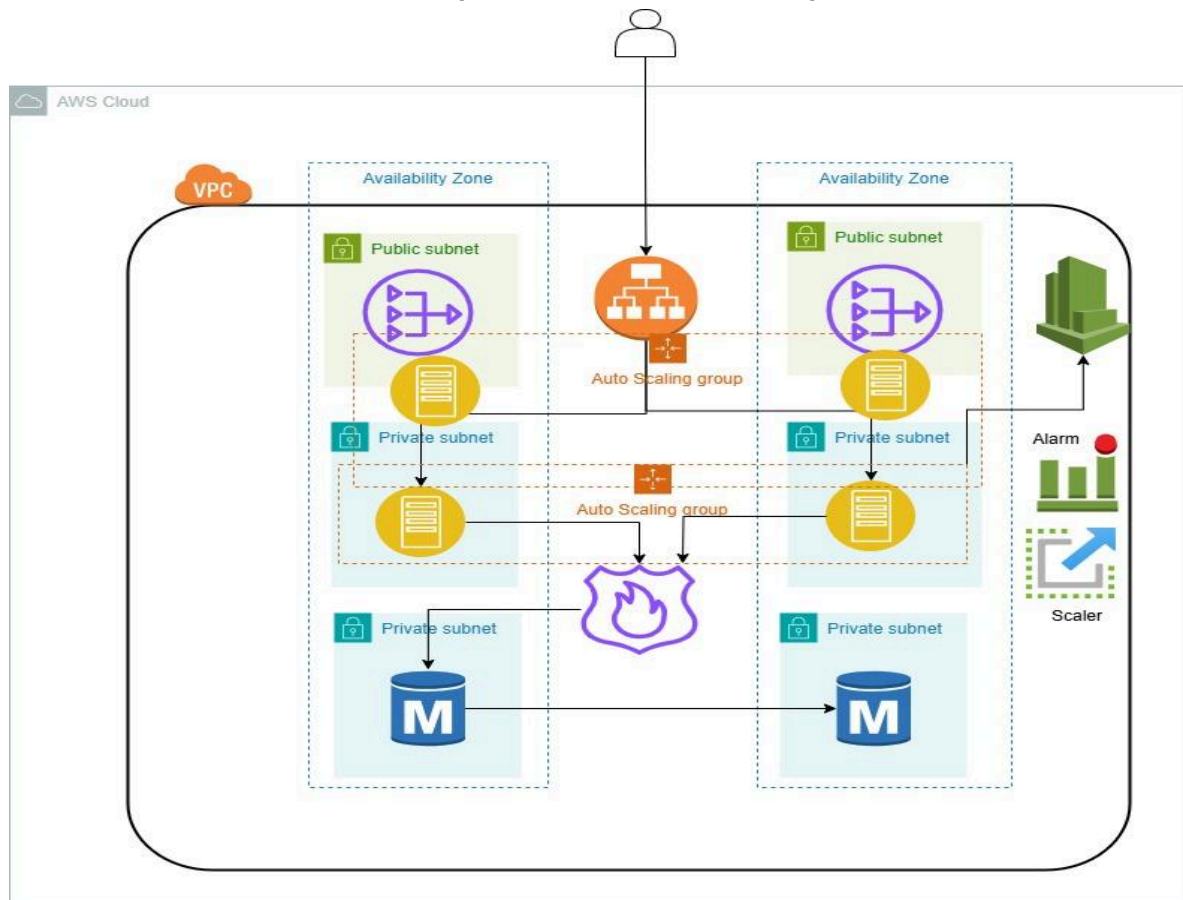


## Diagrama de componentes

Se sugiere un componente que maneje los accesos desde la aplicación móvil y la SPA mismo que enviará las notificaciones mediante un componente de notificaciones, para la funcionalidad de la aplicación se tiene un componente que se conecta con el sistema del banco central para manejar transferencias externas y se conecta con el core bancario para consulta de saldos y transferencias internas, la integración con servicios externos a la aplicación se hará con el patrón de diseño facade o fachada que enmascara la funcionalidad para la integración.



## Arquitectura Alta disponibilidad y Resistente a desastres y tolerante a fallos



En esta arquitectura se obtiene resistencia a desastres al implementar dos availability zones o zonas de disponibilidad en AWS. las dos zonas de disponibilidad deben estar al menos 50 km alejadas las unas de la otra para que en caso de desastre natural no afecte a una de ellas. Para obtener alta disponibilidad se implementa un auto scaling group con la aplicación web a deployar misma que aumentará el número de instancias dependiendo de las alarmas que son definidas en el recurso de cloud watch de la arquitectura. De esta manera la arquitectura de la aplicación es tolerante a fallos.

Una alternativa a esta arquitectura es utilizar EKS pues este servicio facilita al usuario el manejo de un cluster de Kubernetes y simplemente se necesita manejar la infraestructura como código implementando el proxy las políticas de acceso, conceptos como rate limiting, circuit breaking etc pueden ser manejados con plantillas helm facilitando el deployment de la aplicación y la escalabilidad de la misma.

## Arquitectura de Autorización

El protocolo de autorización a utilizar será OAuth y se aplicará como se muestra en el siguiente diagrama.

