

Diseño de Sistemas.

Actividad Nº 3:

Diseño​ ​de​ ​Arquitectura.

Baltazar, Cristian.

Segovia, Alejandro.

Mansilla, Adrián.

**Caso 1 : Sistema de archivos distribuido**

En un contexto descentralizado, el objetivo es que todos los nodos que integren la red

mantengan sincronizados un conjunto de recursos en común. Cada nodo debe poder actuar

tanto en modo cliente como servidor. La solución debe ser flexible para escalar

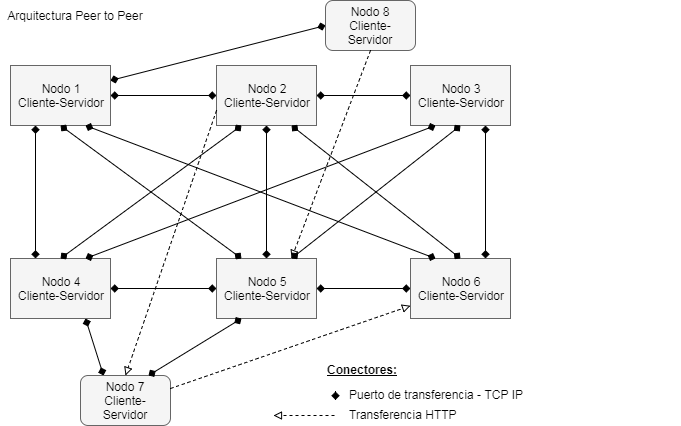
horizontalmente y soportar eventuales fallos de conectividad entre nodos.

Se pide:

● Diseñar y documentar una arquitectura.

● Mencionar las posibles alternativas de conectores de integración indicando ventajas

y desventajas de cada una.



Se puede escalar horizontalmente agregando más nodos a la red que por el protocolo de transferencia se unirán la red.

La ventaja de los conectores es que son estándar y los básicos de internet. La desventaja es que dependen del ancho de banda de cada nodo.

**Caso 2 : Terminales NFC**

Un proveedor de servicios de cobranza necesita integrar las terminales lectores NFC

ubicadas en las tiendas de sus clientes, con el servidor central. Cada vez que se genera

una transacción de compra, la terminal debe transmitir la operación asegurando su

atomicidad. Se desea que en ningún momento la terminal quede bloqueada a la espera de

una confirmación de recepción por parte del servidor de modo que, además, tolere alta

frecuencia de operaciones. Asimismo, la solución debe ofrecer la suficiente flexibilidad para

integrar diferentes fabricantes de terminales y soportar múltiples formatos de representación

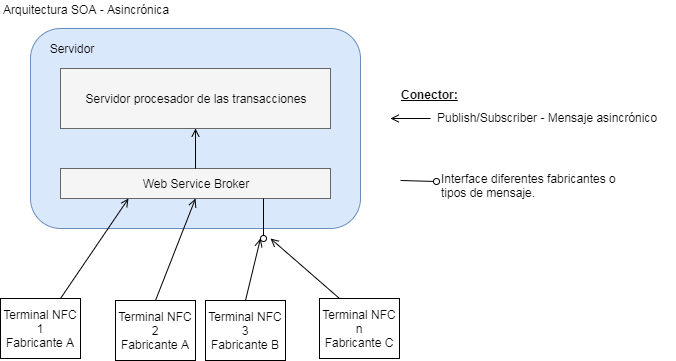
de los datos.

Se pide:

● Diseñar y documentar una arquitectura.

● Indicar el o los patrones de integración recomendados y justificar mencionando

aspectos críticos del caso.



El patrón de integración con el servidor recomendado es asincrónica para que los clientes no esperen respuesta del servidor. Los diferentes terminales NFC envían las transacción asincrónicamente y no se detienen a esperar confirmación, el servidor las recibe a través del orquestador (broker) que posee interfaces para soportar diferentes fabricantes.

**Caso 3 : Integración de sistemas empresariales**

Se precisa establecer interoperabilidad entre sistemas de tecnologías diversas. Si bien

algunos de estos sistemas están basados en tecnología antigua (sistemas legacy), todos

cuentan con acceso a Internet a través de HTTP. En algunos casos puede ser necesario

realizar algunas conversiones en el soporte de representación de los datos. La solución

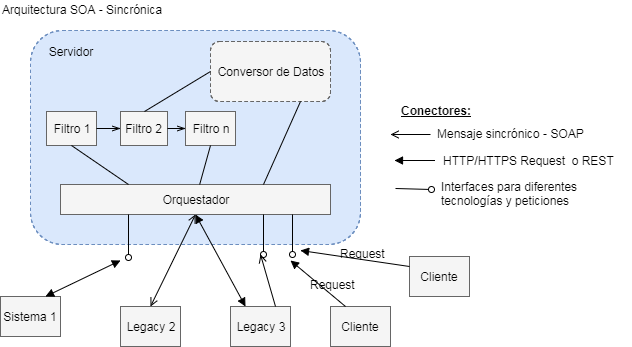
deberá permitir administrar y re-direccionar flujos de acuerdo a reglas de negocio específicas para diferentes servicios. Es deseable que también de soporte al envío de mensajes entre nodos.

Se pide:

● Diseñar y documentar una arquitectura.

● Mencionar las posibles alternativas de conectores de integración indicando ventajas

y desventajas de cada una.



La arquitectura elegida es SOA - Sincrónica, con una pequeña implementación de arquitectura de flujo de datos en el servidor.

Arquitectura SOA porque se requiere interconectar diferentes sistemas con diferentes tecnologías y esta arquitectura provee los conectores adecuados, además de permitir la administración, redirección y procesamiento (aplicar reglas, filtros y conversión de datos) en el lado del servidor. También permite la intercomunicación entre los sistemas conectados (nodos).

La pequeña parte de arquitectura de flujo de datos es para cubrir el pedido de aplicación de reglas y filtros.

Sincrónica porque no hay un requerimiento que indique que no se puede esperar una confirmación de las operaciones y siempre es preferible confirmar el éxito o no de una operación.

**Caso 4 : Tráfico aéreo**

En la consola de control de tráfico aéreo donde el operador desarrolla su actividad analítica,

reside gran parte del procesamiento de datos recuperados de un servidor. Dada la criticidad

del sistema, es tolerable que ocurran bloqueos en la terminal cuando debe sincronizar la

información local con la remota. Al tratarse de una solución integral donde todo el software

es desarrollado por el mismo fabricante, no es necesario ofrecer representación

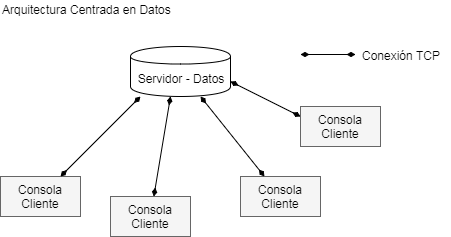
estandarizada de los datos que intercambien.

Se pide:

● Indicar a qué patrón de arquitectura está haciendo referencia el enunciado.

● Recomendar el patrón de integración más apropiado, explicando sus principales

fortalezas y también mencionar aspectos críticos de su implementación



El patrón de integración más apropiado es Centrada en Datos, los componentes se comunican a través del repositorio compartido.

Las ventajas/fortalezas:

* Proporciona integridad.
* Proporciona escalabilidad.
* Ofrece capacidad de reutilización de los componentes (no tienen

comunicación entre sí).

* Reduce la sobrecarga de datos entre los componentes.

**Caso 5 : Marina mercante**

Una compañía naviera precisa actualizar el modo como sus buques notifican a la base

acerca de las condiciones meteorológicas de la región oceánica que están atravesando. A

su vez, el sistema deberá suministrar a cada buque de reportes meteorológicos regionales

que sean de su interés. Como la calidad del canal de comunicación satelital podría verse

afectado por lluvias intensas, es crítico que, tanto en las mediciones [buque → base] como

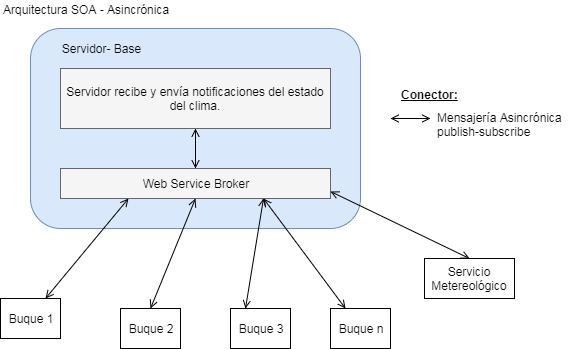
los informes [base → buque], esté garantizada su integridad.

Se pide:

● Diseñar y documentar una arquitectura.

● Indicar el o los patrones de integración recomendados y justificar mencionando

aspectos críticos del caso.



Patrón SOA - Asincrónico publish subscribe ya que el canal de comunicación puede no ser estable y la comunicación es bidireccional.

Un buque publica una mensaje a la base (suscribe) y recibe respuesta cuando el canal de comunicación se restablezca.

Por otro lado la base puede recibir notificaciones directamente del servicio meteorológico y redireccionarlas a los buques.

**Caso 6 : FinTech**

Un emprendimiento que surgió como una pequeña iniciativa “de garage” está

incrementando su cartera de clientes. El núcleo de su actividad consiste en suministrar

información a sistemas de inversores acerca del desempeño histórico de diferentes

compañías que cotizan en la bolsa. Como en algunos casos estos servicios de información

son personalizados, es restricción obligatoria (mandatory) que las fuentes estén aisladas

entre sí (al menos desde un enfoque lógico). En algunos casos, estos informes pueden ser

una agregación de servicios propios o de terceros. Es deseable que la solución permita alta

flexibilidad en función de cambios en los requerimientos de los clientes.

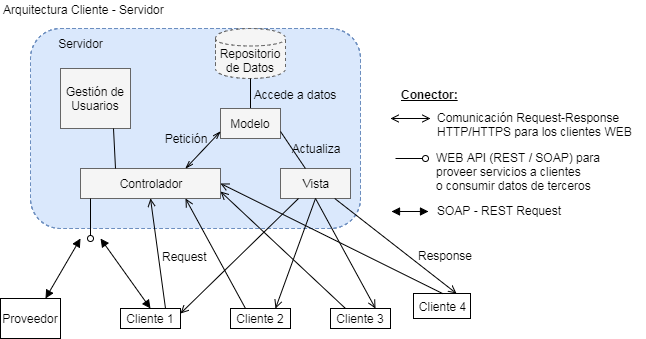
Más info: <https://www.pwc.com/us/en/financial-services/publications/viewpoints/assets/pwc-fsi-what-is-fintech.pdf>

Se pide:

● Diseñar y documentar una arquitectura.

● Mencionar las posibles alternativas de conectores de integración indicando ventajas

y desventajas de cada una.



La arquitectura elegida es cliente servidor para la comunicación con los clientes e internamente en el servidor se aplica la arquitectura Modelo Vista Controlador para darle estructura.

Los usuarios del sistema son identificados e interactúan con el sistema a través de una interfaz web. Acceden a sus propios datos almacenados en el repositorio de datos. También pueden acceder a servicios de terceros provistos a través de las APIs.

Internamente en el servidor la arquitectura MVC brinda flexibilidad y mantenibilidad. Permite el desacople entre la vista y el modelo.

Es fácil reemplazar cualquiera de los tres componentes.