



Asignatura..... **SISTEMAS INFORMÁTICOS II** ..... Grupo..... **231**  
 Apellidos ..... Nombre.....  
 Ejercicio del día..... **26 de febrero de 2015. Examen parcial.**.....

Teoría 1 (2.5)	Teoría 2 (2.5)	Teoría 3 (2.5)	Teoría 4 (2.5)	Total Teoría (10)

**1.- TEORÍA (10 puntos).** Contesta de modo claro y conciso a las siguientes cuestiones.

1. Explica el proceso de desarrollo y funcionamiento de una aplicación cliente servidor basada en SUN-RPC.

2. Explica las componentes de un sistema distribuido basado en servicios web y describe cada uno de los estándares que utilizan para comunicarse e interaccionar.



Asignatura..... **SISTEMAS INFORMÁTICOS II** ..... Grupo..... **231**  
 Apellidos ..... Nombre.....  
 Ejercicio del día..... **26 de febrero de 2015. Examen parcial.**.....

3. Explica el uso de un Bus de Servicios de Empresa (Enterprise Service Bus) para integrar aplicaciones heterogéneas a nivel empresarial. Describe cuáles son sus ventajas y desventajas.

4. Explica para qué se utilizan los acuerdos a nivel de servicio en un sistema distribuido y pon un ejemplo de requisito funcional y no funcional del sistema. Nombra al menos 2 mecanismos que podrían utilizarse para estimar el rendimiento del mismo.



**SISTEMAS INFORMÁTICOS II**

Asignatura..... Grupo..... **231**

Apellidos ..... Nombre.....

Ejercicio del día..... **26 de febrero de 2015. Examen parcial.**

2.1 (6)	3.1 (3)	3.2 (1)	Total Problema (10)

**2. PROBLEMA.** Considerar los siguientes casos particulares de sistemas distribuidos:

1. Servidor de información sobre empleados. El servidor proporciona información sobre los datos de los distintos empleados de la empresa, incluyendo datos personales, tareas asignadas, y ventas realizadas. Los datos asociados a cada empleado pueden obtenerse y modificarse de forma remota. Los clientes que usarán el servicio serán homogéneos, pero habrá que pasar a través del corta-fuegos de la empresa. El ancho de banda de la empresa no es grande y se desearía que el tamaño de los mensajes no fuese excesivo.
2. Servidor de información sobre ubicaciones. Clientes heterogéneos acceden a través de una interfaz de programación a diversos servicios relativos a una ubicación especificada: estado del tráfico, restaurantes cercanos, información meteorológica, etc. Se espera que se faciliten nuevos servicios en el futuro. Se querría que los clientes pudiesen acceder a estos nuevos servicios fácilmente. El coste de procesar los mensajes y el tamaño de los mismos no son factores críticos.
3. Sistema de procesamiento de documentos xml. Un documento en formato xml enviado por un cliente pasa a través de varios nodos del sistema que realizan ciertas tareas de procesamiento, añadiendo, eliminando y modificando elementos. Habrá documentos que deberán ser procesados antes que otros. Habrá documentos no válidos que el sistema deberá ignorar. El tiempo de procesamiento no es crítico.
4. Servidor de tiempo. Clientes homogéneos consultan al servidor. El servidor al recibir el mensaje responde con ciertas marcas de tiempo. Los clientes utilizan esta información para actualizar su reloj interno.

**2.1. (6 puntos)** Para cada uno de ellos se pide elegir razonadamente el mecanismo de comunicación más adecuado: comunicación no orientada a conexión (UDP), colas de mensajes, servicios web basados en REST y servicios web basados en SOAP.

**No se tendrán en cuenta respuestas sin justificación.**

1. El mecanismo más adecuado sería servicios web basados en REST. Cada usuario y las ventas y tareas asociadas se podrían exportar como recursos que se podrían acceder y modificar mediante los métodos del protocolo HTTP (GET; PUT, POST, DELETE). Los servicios web pasarían a través de cortafuegos, y clientes y servidores podrían ponerse de acuerdo en las representaciones de los recursos, para que fuesen algo más eficiente que los mensajes SOAP de los servicios web estándar. Descartamos colas de mensajes ya que querríamos funcionalidad síncrona, y UDP por que el tamaño de los datos puede ser mayor que lo que cabe en un datagrama y no aporta fiabilidad de entrega.
2. Los clientes son heterogéneos y acceden al servicio a través de una interfaz de programación. Esto favorece a los servicios web basados en SOAP. Además, podríamos usar UDDI y WSDL para localizar e invocar nuevos servicios que hayan aparecido. También

**SISTEMAS INFORMÁTICOS II****231**

Asignatura..... Grupo.....

Apellidos..... Nombre.....

Ejercicio del día..... **26 de febrero de 2015. Examen parcial.**

porque el tamaño y coste de procesado de los mensajes no es crítico. No usaríamos colas de mensajes ni UDP por las razones del caso anterior.

3. En este caso la solución más interesante es colas de mensajes porque el formato de los datos es homogéneo (documentos en xml) y es un caso típico de workflow (ciclo de trabajo). Las consultas no serían bloqueantes, y los clientes podrían seguir trabajando mientras se procesa el documento que han mandado. También se podrían filtrar documentos y establecer prioridades. No habría problema tampoco debido a que alguno de los nodos del sistema se cayese. Los mensajes no se perderían y quedarían almacenados en la cola correspondiente.
4. En este caso parece crítico que los mensajes se envíen rápidamente. Las interacciones serían sincronicas. Además la información a enviar es muy pequeña, solo marcas de tiempo. Los clientes también son homogéneos con lo que no habría que traducir datos. Si un mensaje se pierde se podría re-enviar sin problema, ya que las interacciones son del tipo petición respuesta e idempotentes. Todo esto apunta a utilizar UDP.



Asignatura..... **SISTEMAS INFORMÁTICOS II** ..... Grupo..... **231**  
 Apellidos ..... Nombre.....  
 Ejercicio del día..... **26 de febrero de 2015. Examen parcial.**.....

**3. PROBLEMA** Se quiere prestar un servicio que permita gestionar la asignación docente de la Escuela Politécnica Superior utilizando servicios web REST. Dicho servicio podrá ser útil para consultar los datos e información de contacto de cada profesor, para fijar y consultar las asignaturas que imparte cada profesor, y para fijar las asignaturas impartidas en la escuela. Cada profesor se identifica por su nombre y cada asignatura por su código.

**3.1. (3 puntos)** Indicar de forma razonada una posible URI (Uniform Resource Identifier) y el posible método HTTP (GET, POST, DELETE, PUT) a utilizar para ofrecer las siguientes funciones:

- a) Modificar los datos personales del profesor “Juan Pérez”.

URI: <http://www.eps.uam.es/profesores/juanperez>

Método: POST

- b) Indicar que el profesor “Juan Pérez” imparte la asignatura “Proyecto de Programación” con código 17823.

URI: <http://www.eps.uam.es/profesores/juanperez/asignaturas/17823>

Método: PUT

- c) Obtener las asignaturas impartidas por el profesor “Juan Pérez”.

URI: <http://www.eps.uam.es/profesores/juanperez/asignaturas>

Método: GET

**3.2 (1 punto)** Dar un ejemplo razonado de una posible representación del profesor “Juan Pérez”.

La representación podría tener cualquier formato, pero debería incluir sus datos personales y enlaces a recursos relacionados. Elegimos XML.

```
<profesor>
  <nombre> Juan Pérez </nombre>
  <email>juan.perez@uam.es </email>
  <despacho> B-432 </despacho>
  <asignaturas>http://www.eps.uam.es/juanperez/asignaturas</asignaturas>
</profesor>
```