**CUESTIONARIO DE SISTEMAS**

**DISTRIBUIDOS**

1-DEFINA EN FORMA APROXIMADA QUE ES LO QUE ENTIENDE POR SISTEMA DISTRIBUIDO  
 Colección de computadoras independientes que le dan al usuario la idea de constituir un único sistema coherente.   
 2-COMO TIENEN QUE SER LOS SISTEMAS DISTRIBUIDOS, (DOS CUALIDADES)?   
Los sistemas distribuidos debe ser abierto y debe ser escalable.  
3-A LOS SIST DISTRIBUIDOS, COMO TAMBIEN SE LOS CONOCE?   
Los sistemas distribuidos son conocidos también como Middleware.

4-COMO TIENEN QUE PERMANECER EL HARDWARE Y EL SOFTWARE?

Las diferencias entre los distintos tipos de hardware y sistemas operativos deben permanecer ocultas.

5-EXPLIQUE MIGRACION, REPLICACION, REUBICACION Y CONCURRENCIA   
Migración: que un recurso pudiera moverse a otra ubicación  
Replicación: del número de copias de un recurso  
Reubicación: que un recurso pudiera moverse a otra ubicación mientras está en uso  
Concurrencia: que un recurso pueda ser compartido por varios usuarios que compiten por él.

6-EL GRADO DE APERTURA QUE CARACTERISTICAS TIENE?   
Interoperabilidad: las implementaciones de sistemas y componentes de fabricantes pueden coexistir y trabajar juntos.  
Extensible: debe ser fácil de configurar para componentes diferentes, a parte de agregar nuevos o reemplazar los existentes sin afectar a los que permanecen en su lugar  
Flexibilidad: los componentes son relativamente pequeños y fáciles de reemplazar o adaptar

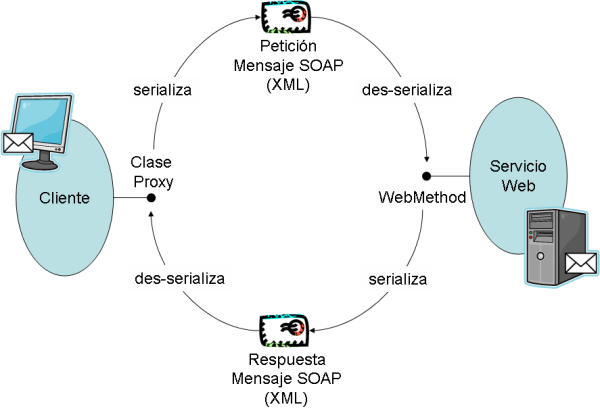
7-y RESPECTO DE LA ESCALABILIDAD?   
Se puede medir desde varios puntos de vista:  
-Respecto a su tamaño: podemos agregar fácilmente recursos y usuarios. Hay que evitar los algoritmos centralizados.   
-Desde el punto de vista geográfico: usuario y recursos pueden radicar muy lejos uno de los otros  
-Escalabilidad administrativa: puede ser fácil de manejar incluso involucrando muchas organizaciones administrativas diferentes.  
  
8-LAS TECNICAS DE ESCALAMIEMTO PARA QUE ME SIRVEN?   
La tecnicas de escalamiento sirven para disminuir el impacto de los problemas de escalar un sistema, las cuales son:  
-Ocultar las latencias de comunicación  
-Distribución: dividir un componente en partes más pequeñas y distribuir dichas partes a lo largo del sistema  
-Replicación: replicar los componentes a lo largo del sistema distribuido

9-QUE CARACTERISTICAS TENDRIA UN SISTEMA DISTRIBUIDO IDEAL??   
Un sistema distribuido ideal debe tener una red confiable, segura y homogénea por lo que la topología no debe cambiar, su latencia debería ser cero, el ancho de banda infinito, el costo de transporte igual a cero, que exista un administrador, etc.

10-QUE COSAS SE TIENEN EN CUENTA EN TODOS LOS TIPOS DE SISTEMAS DISTRIBUIDOS?   
Se tienen en cuenta la arquitectura, procesos, comunicación, asignación de nombres, sincronización, consistencia y replicación, tolerancia a fallas, seguridad  
11-COMO FUNCIONAN LOS SISTEMAS DISTRIBUIDOS BASADOS EN LA COORDINACION?   
Los mensajes son direccionados por tema, por lo que no tienen la dirección de su receptor. Los procesos que desean recibir mensajes deben suscribirse a un tema específico. El middleware se encarga de que los mensajes se encaucen desde los editores hasta los suscriptores.   
12-EXPLIQUE SIST DIDTRIBUIDOS DE COMPUTO   
En cluster: El HW consta de una colección de estaciones de trabajo similares o PC, conectadas cercanamente por medio de una LAN de alta velocidad. Cada nodo ejecuta el mismo Sistema Operativo.   
En malla (grid): consta de sistemas distribuidos construidos generalmente como un conjunto de sistemas de cómputo en donde cada sistema podría caer dentro de un dominio administrativo distinto, y obviamente distintas sus características de HW, SW y de tecnología de red.

13-A QUE SE CONOCE COMO RPC TRANSACCIONAL?   
RPC: llamada a procedimiento remoto, el cual es un programa que utiliza una computadora para ejecutar código en otra máquina remota sin tener que preocuparse por las comunicaciones entre ambas, de forma que parezca que se ejecuta local. Los clientes usan procesos que se ejecutan en CPUs remotas, pueden ser síncronas o asíncronas.

14-CUALES SON LOS DESAFIOS DE LOS SISTEMAS DISTRIBUIDOS?   
La heterogeneidad de redes, HW de computadores, sistemas operativos, lenguajes de programación e implementación de distintos desarrolladores.

15-COMO GRAFICARIA UN SERVICIO WEB?   


16-DESCRIBA WSDL, UDDI Y SOAP   
WSDL (Web Service Description Language): lenguaje formal similar a los lenguajes de interfaz usados para soportar la comunicación basada en RPC. Puede ser transformado en forma automática en Resguardo del lado del cliente y del lado del servidor.  
SOAP (Simple Object Access Protocol): estructura donde una gran parte de la comunicación entre dos procesos puede ser estandarizada.

17-QUE ASPECTOS SE EVALUAN EN LOS WEB SERVICES? DETALLELOS   
**Sincronización**: es necesario que los servidores se sincronicen entre sí y que sus acciones sean coordinadas.   
**Consistencia y Replicación**: se idean estrategias para guardar en caché y replicar contenido tanto estático como dinámico, es decir, documentos generados a consecuencia de una solicitud. El almacenamiento en caché del lado del cliente ocurre generalmente en 2 lugares:  
 -Los browsers cuentan con un almacenamiento en la caché que se pueda configurar indicando en que momento se deberá verificar la consistencia  
 -El sitio de un cliente ejecuta con frecuencia un proxy web. Un proxy web acepta solicitudes de clientes locales y las transfiere a servidores web. Cuando llega una respuesta, el resultado se transfiere al cliente.   
La replicación sirve para que el sitio sea continuo y de fácil acceso.   
**Algunas métricas**:  
 -Métrica de latencia: mide el tiempo requerido para realizar una acción  
 -Métrica de uso de red: se relaciona con el ancho de banda consumido en función del número de bytes que se van a transferir  
 -Métrica espacial: mide la distancia entre nodos en función del número de conexiones intermedias de enrutamiento a nivel de red o de conexiones intermedias entre sistemas autónomos  
-Métricas de consistencia: informa en qué medida o grado se está desviando una réplica de su copia maestra.   
-Métrica financiera: forma otra clase para medir que tan bien está funcionando una CDN. Está estrechamente relacionada con la infraestructura real de Internet.   
**Tolerancia a fallas:** se logra principalmente por medio de almacenamiento en caché del lado del cliente y replicación de servidores. Los problemas se agravan para servicios diseñados para manejar fallas bizantinas, un servicio que tolere este tipo de fallas (BTS) puede necesitar actuar como cliente de otro servicio no replicado.  
**Seguridad**: el método para establecer un canal seguro en la web es usar una capa de socket seguro (SSL), que fue implementada originalmente por Netscape, hoy es conocido como protocolo de seguridad en la capa de transporte (TLS)

18- QUE PROPONEN CASTRO Y LISKOV?   
Una solución al sistema BFT debe manejar:  
-Los clientes de un servicio BTF deberán ver al servicio como a cualquier otro servicio web  
-Deberá garantizar la consistencia interna cuando actúa como cliente  
-Deberán tratarse como un servicio BFT que actúa como cliente y como una sola entidad.

19-UN CANAL SEGURO SE ESTABLECE EN DOS FASES. EXPLIQUELO   
El establecimiento de un canal seguro se realiza en dos fases:  
-El cliente informa al servidor acerca de algoritmos criptográficos que puede manejar así como también, cualquier método de compresión que soporte.  
-La segunda fase es la autenticación: siempre se requiere que el servidor se autentique a sí mismo, por lo que el cliente pasa un certificado que contiene una clave pública firmada por una autoridad de certificación. Si el servidor requiere que el cliente se autentifique, también el cliente deberá enviar un certificado al servidor.

20-CUALES SON LOS MODELOS ARQUITECTONICOS?   
Estilo arquitectónico se formula en términos de componentes, el intercambio entre componentes y cómo estos elementos se pueden configurar juntos en un sistema. Los estilos arquitectónicos que tenemos son en capas, basadas en objetos, centradas en datos y basadas en eventos.

21-DEFINA LAS INVOCACIONES PARA RPC, RMI Y CORBA Y COMPARELOS EN TODOS SUS ASPECTOS

MUY IMPORTANTE   
RPC: lógica de negocios escrita como procedimientos que se invoca en forma remota  
RMI: lógica de negocios escrita como objetos cuyos métodos son invocados en forma remota  
CORBA: lógica de negocios escrita como objetos que son invocados en forma remota.   
Comparación:  
-RMI es más lento que RPC por la interpretación byte-code en la JVM, pero se puede mejorar con el compilador JIT  
-RMI tiene una ventaja adicional sobre RPC porque pertenece al mundo orientado a objetos de Java. Los sistemas RPC tradicionales son lenguajes neutrales, de modo que no pueden proveer la funcionalidad que no estén disponibles en sus plataformas destino. RMI por su naturaleza Java se conecta a los sistemas usando métodos nativos, por lo que puede proveer tecnología distribuida en forma directa, natural y potente.

22-COMO SE HACE PARA CONSTRUIR UN WEB SERVICE Y COMO SE CONSTRUYE UNA APLICACIÓN EN CORBA? EXPLIQUE LAS DIFERENCIAS.   
Construcción de un Web Service:  
Archivo--Nuevo--Proyecto--Sitio web vacío de ASP.NET--En la solución se agrega un elemento--Servicio Web asmx--Para invocarlo tenemos que agregar referencia de servicio-- Servicio Web de esta solución.

Construcción de una aplicación CORBA: