|  |  |
| --- | --- |
| INFORMACIÓN BÁSICA | |
| NOMBRE DE LA PRÁCTICA: Construcción de una arquitectura Cliente/Servidor Remota | PRÁCTICA No.: 1 |
| **ASIGNATURA:**  Sistemas Distribuidos | |
| **TEMA DE LA PRÁCTICA:**  Comunicación en Sistemas Distribuidos, Programación de Sockets, Arquitectura Cliente Servidor, Replicación o tolerancia a fallas | |
| LABORATORIO A UTILIZAR: GO 310 | |
| TIEMPO*: (4 horas).*  TRABAJO GRUPAL: TRABAJO INDIVIDUAL: x | |
| **CONTENIDO DE LA GUÍA** | |
| **COMPETENCIAS DISCIPLINARES.**  Controla la comunicación de mensajes en una arquitectura cliente-servidor en forma unidireccional y bidireccional aplicando el protocolo TCP/IP  **INVESTIGATIVAS.**  Trabaja en la creación de un modelo cliente / servidor usando un middleware (SOCKET, PROTOCOLO TCP/IP, ServerSocket, Programción concurrente, Threads) | |
| **MARCO TEORICO**  En el mundo de las comunicaciones la mayor parte se rige por un modelo llamado Cliente-Servidor, el cual permite características como:   * Usabilidad * Flexibilidad * Interoperabilidad y * Escalabilidad en las comunicaciones.   Funcionamiento de un modelo Cliente/ Servidor  cs.gif  Tomado de : ““ARQUITECTURA CLIENTE/SERVIDOR. Recuperado de: <http://nereida.deioc.ull.es/~cleon/doctorado/tic02/cs.html>”  **Características del modelo cliente/servidor**   1. El equipo que actúa como [servidor](http://www.alegsa.com.ar/Dic/servidor.php) (host), ejecuta los programas que contestan los requerimientos de los clientes. 2. La interacción entre cliente y servidor es representada usando diagramas de secuencia 3. Hay una centralización de los recursos, datos y accesos. 4. Facilita el mantenimiento y actualización del lado del servidor 5. Toda la información es almacenada en el lado del servidor, que suele tener mayor seguridad que los clientes. | |
| **CONSULTA PREVIA**   1. Transacciones de datos replicados: Los objetos en sistemas transaccionales pueden estar replicados para mejorar la disponibilidad y el rendimiento. Esto se logra a través de distintos esquemas de replicación (REFERENCIA 1):    1. **Replica de copia primaria**: Un gestor primario se encarga de la concurrencia (FIFO). Para consumar una transacción, el primario comunica con las réplicas y después responde el cliente, o bien, responde inmediatamente y realiza las copias en segundo plano (a esto se le conoce como estrategia exhaustiva y estrategia perezosa respectivamente). Si el primario cae, el reemplazo puede no tener la última versión de los datos.    2. **Uno lee, todos escriben**: Un cliente puede leer de cualquiera de las réplicas, ya que todas están actualizadas. Por otro lado, un gestor de réplicas escribe y envía al resto la actualización. Cuando todos han realizado dicha actualización, termina la transacción.    3. **Consenso por quorum**: El quorum es un subgrupo de gestores de réplicas autorizado a efectuar operaciones. Es un esquema diseñado para reducir el número de réplicas que deben ejecutar las actualizaciones. En esta transacción tenemos:       1. **Operaciones de lectura**: Recuperar un quorum de lectura (cualquier conjunto de r copias). De las r copias, seleccionar la copia con el nº de versión más alto. Entonces se retorna el valor de dicha copia.       2. **Operación de escritura**: Tomar un quorum de escritura (cualquier conjunto de w copias). De las w copias, obtener el nº de versión más alto. Se incrementa el nº de versión y se escribe el nuevo valor y el nuevo nº de versión en todas las w copias del quorum. 2. Comunicaciones unidireccionales y bidireccionales Full-dúplex en una arquitectura cliente servidor aplicando el protocolo TCP/IP (REFERENCIA 2).   El modo de operación de un enlace de comunicación es una característica para considerar desde la perspectiva de un sistema distribuido. Usualmente para transmitir datos por un enlace de comunicación se ocupan tres modos:   * 1. **Comunicación unidireccional**: También conocida como *simplex*, se presenta cuando los datos viajan en una sola dirección.   2. **Comunicación bidireccional**: También conocida como *half-duplex*, permite que los datos viajen en dos direcciones, una a la vez.   3. **Comunicación Full-duplex**: En este modo los datos viajan simultáneamente en ambas direcciones.   Por lo general, en una arquitectura cliente-servidor se debe garantizar que la información llegue a ambos extremos del servicio, por lo que en estos modos de operación debe utilizarse un protocolo que garantice esto. El protocolo TCP/IP es adecuado en esta situación debido a que define cuidadosamente cómo se mueve la información desde el remitente hasta el destinatario. En primer lugar, los programas de aplicación envían mensajes o corrientes de datos a uno de los protocolos de la capa de transporte de Internet, en este caso, a la capa de TCP (Transmission Control Protocolo). Este protocolo recibe los datos de la aplicación, los dividen en partes más pequeñas llamadas paquetes, añaden una dirección de destino y, a continuación, pasan los paquetes a la siguiente capa de protocolo, la capa de red de Internet. La capa de red de Internet pone el paquete en un datagrama de IP (Internet Protocol), pone la cabecera y la cola de datagrama, decide dónde enviar el datagrama (directamente a un destino o a una pasarela) y pasa el datagrama a la capa de interfaz de red. La capa de interfaz de red acepta los datagramas IP y los transmite como tramas a través de un hardware de red específico, por ejemplo redes Ethernet o de Red en anillo. | |
| **PALABRAS CLAVE**  Consulta la definición y el campo de aplicación   1. Transacciones: Son secuencias de una o más operaciones que tienen que ocurrir de manera atómica. En sistemas distribuidos se consideran como unidades indivisibles de manipulación de información, y se comprenden como mecanismos de alto nivel que ocultan los aspectos técnicos de sincronización, como la exclusión mutua, interbloqueos y recuperación de fallas (REFERENCIA 2). 2. Concurrencia: Es la realización simultanea de procesos dentro de un sistema; estos procesos pueden ser ejecutados por uno o más programas según hayan sido planificados. La concurrencia facilita la programación de aplicaciones, agiliza la ejecución de procesos, posibilita el uso multiusuario y optimiza el uso de recursos. 3. Modelo arquitectónico: Es aquel que se caracteriza por el reparto de responsabilidades entre componentes del sistema y la ubicación de dichos componentes (REFERENCIA 3). Entre estos se destacan:    1. Cliente-servidor.    2. P2P. 4. Modelo Fundamental: Son aquellos que comparten ciertas propiedades fundamentales (REFERENCIA 3):    1. Procesos que se comunican por paso de mensajes a través de una red de computadores.    2. Se debe definir y clasificar la comunicación entre elementos del sistema (Modelo de interacción).    3. Se debe definir y clasificar los fallos que pueden darse en el sistema (Modelo de fallos).    4. Se debe definir y clasificar los fallos que pueden afectar al sistema (Modelo de seguridad). 5. Objetos distribuidos e invocación remota: Los objetos distribuidos son aquellos gestionados por un servidor y sus clientes invocan sus métodos utilizando un “método de invocación remota”. El cliente invoca el método mediante un mensaje al servidor que gestiona el objeto, se ejecuta el método del objeto en el servidor y el resultado se devuelve al cliente en otro mensaje (REFERENCIA 2). | |
| **METODOLOGÍA.**   * El taller es individual y para trabajar en tres sesiones de clase. * Cada estudiante debe ubicarse en un equipo y asegurarse de tener instaladas las aplicaciones requeridas. * En el laboratorio el estudiante debe proceder a realizar la práctica y posteriormente puede enviar la solución del taller para ser revisada por el profesor. * El estudiante puede formular inquietudes al profesor durante el desarrollo de la práctica.   **PROCEDIMIENTO A UTILIZAR**  **Procedimiento para la sesión 1.**   1. Revisar y comprender los requerimientos de la guía de laboratorio. Pedir al profesor las aclaraciones del caso: 2. Leer marco teórico. 3. Desarrollar consulta previa. 4. Análisis de los requerimientos del informe de laboratorio. 5. Diligenciar los datos básicos del informe. 6. Formular las habilidades que desea adquirir o desarrollar a través de la práctica. 7. Elaborar mapa conceptual   **Procedimiento para la sesión 2.**   1. Construya una aplicación cliente servidor utilizando java, python o visual studio.NET. El proyecto se debe socializar.   La guía esta para trabajar en tres sesiones de Clase. En la primera sesión se debe trabajar en la parte los 3 primeros puntos del cuestionario. En la sesión dos se debe adelantar los puntos del 4 al 7 del cuestionario y en la sesión tres los puntos finales del cuestionario  **INFORMACION BASICA**  1) El estudiante debe ingresar al laboratorio y ubicarse en un computador.  2) Descargar la guía de laboratorio.  3) Leer y consultar los temas que hace referencia el marco teórico.  4) Leer, analizar y resolver el problema propuesto en consulta previa.  5) Diligenciar cada una de las secciones de la guía de laboratorio: nombre, asignatura, grupo, programa, tipo de trabajo, objetivos, mapa conceptual, cuestionario, causas de error, conclusiones, aplicación profesional, bibliografía.  6) Resolver y responder el cuestionario de la guía.  7) Preparar documento y soportes de entrega.  8) Enviar documento (archivo) respuesta mediante Aula Virtual No se reciben trabajos por correo electrónico. | |
| **MATERIALES, EQUIPOS Y VIRTUALIDAD A UTILIZAR**   |  |  |  | | --- | --- | --- | | **Materiales y Equipos** | **Virtualidad** | **Materiales Estudiante** | | 1 computadora (Windows 7, 8,10 con un programa de packet Tracer  2. Linux Debian 10 | Máquinas virtuales virtual Box o Vmware | Unidad de almacenamiento externa. USB | | Conexión a Internet | N/A | N/A | | Cables de alimentación | N/A | N/A | | |
| **BIBLIOGRAFÍA RECOMENDADA.**  Tanenbaum, A.S.; van Steen, Maarten; “Distributed Systems: Principles and Paradigms". Prentice Hall, 2012.  (Colección Biblioteca USB)  Sukumar Ghosh, Distributed Systems, 2nd Edition**,** Publisher: Chapman and Hall/CRC Release Date: July 2014, ISBN: 9781466552975.  **TANENBAUM, ANDREW S. y MAARTEN VAN STEEN Sistemas Distribuidos. Principios y Paradigmas**  Segunda edición PEARSON EDUCACIÓN, México, 2008 ISBN: 978-970-26-1280-3 Área: Computación. Link:  <https://fdocuments.in/document/sistemas-distribuidos-principios-y-paradigmas-andrew-s-tanenbaum-maarten-van-steen-2da-edicion.html>  López Fuentes, Francisco de Asís Sistemas distribuidos / Francisco de Asís López Fuentes. -- México: UAM, Unidad Cuajimalpa, c2015. 200 p. : il., diagramas, col. ; 24 cm. (Una década de la Unidad Cuajimalpa de la Universidad Autónoma Metropolitana) ISBN de la Colección Una Década: 978-607-28-0452-4 ISBN de este libro: 978-607-28-0476-0.  <http://ilitia.cua.uam.mx:8080/jspui/bitstream/123456789/173/1/X56.pdf>  Enric Martínez Gomáriz, Diseño de Sistemas Distribuidos, Introducción a los Sistemas Distribuidos, 2010.  <http://www.essi.upc.edu/~gomariz/index_archivos/IntroduccionSD-EnricMartinez.pdf>  **Webgrafías**  Cliente-Servidor, Catarina, México, recuperado  de <http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/marquez_a_bm/capitulo5.pdf>  Edu Salguero, médium corporatión, médium, recuprado de <https://medium.com/@edusalguero/sistemas-distribuidos-transacciones-y-control-de-concurrencia-1f5845edd5f>  Abel Santín González, Sistemas operativos distribuidos, dit , recuperado de  <https://www.dit.upm.es/~joaquin/so/p2p/p2p.pdf>   |  |  |  | | --- | --- | --- | | **ELABORÓ** | **REVISÓ** | **APROBÓ** | | **Emilio Barajas L.**  **Fecha: 15-03-2021** | **Director(a) de Programa de:**  Ingeniería de Sistemas  **Fecha:** | **Programa de Ingeniería de Sistemas (USB)**  **Fecha:** | |  |  |  | | |
|  | |
|  | |
| **INFORME DE LABORATORIO**  (*Para elaborar por el Estudiante)* | |
| ESTUDIANTES:  * Juan David Gonzalez Dimaté * Luis Felipe Velasco Tao | **ASIGNATURA: SISTEMAS DISTRIBUIDOS** |
| **GRUPO: 4351** |
| **NOTA:** |
| **FORMULE TRES HABILIDADES QUE DESEE ADQUIRIR O DESARROLLAR A TRAVÉS DE LA PRÁCTICA DE LABORATORIO.** *El estudiante formulará desde la revisión de la guía las habilidades que desarrollará a través*  *de la práctica*   * ***Entender las arquitecturas que se pueden utilizar al momento de plantear o desarrollar un sistema distribuido.*** * ***Reconocer el tipo de arquitectura más adecuado al momento de implementar un sistema distribuido.*** * ***Implementar adecuadamente un sistema distribuido en un sistema con otra arquitectura.*** | |
| **Elabore un Mapa conceptual del tema a tratar en la Práctica de Laboratorio.** *El estudiante construirá un mapa conceptual a partir de las palabras claves suministradas por el docente.* | |
| **RESULTADOS.** *Incluir tablas, diagramas en la que los estudiantes puedan ir consignando los datos y observaciones obtenidos en la práctica y generar un análisis de los mismos mediante comparación con la literatura actualizada disponible.* | |
| **CUESTIONARIO.**   1. Explicar 5 aplicaciones que funcionan en el modelo cliente/servidor:    1. Servidor de correo: Un usuario puede utilizar un cliente para realizar diversas operaciones para otros clientes dentro del servidor en la red.    2. Servidor de archivos: Un cliente puede almacenar un archivo dentro del servidor para poder distribuirlo con los demás clientes.    3. Servidor de bases de datos: Desde un cliente se pueden almacenar datos en la base de datos, mientras que en otro cliente se pueden acceder a esos datos, siempre y cuando correspondan al mismo servidor.    4. Servidor web: Las páginas web tienen como cliente al navegador mientras que este utiliza un servidor Apache.    5. Servidor de impresión: Los usuarios de una red de impresoras pueden realizar una solicitud para imprimir un documento en una determinada impresora al servidor, que le transfiere dicha solicitud, junto al documento a la impresora. 2. Cuál es el concepto, función de socket: Es un túnel de comunicación que ayuda a que 2 aplicaciones se comuniquen. Son los puntos finales de enlaces de comunicaciones entre procesos (REFERENCIA 4). 3. Cómo funciona un socket:  Los procesos tratan a los sockets como descriptores de ficheros, de forma que se puedan intercambiar datos con otros procesos, transmitiendo y recibiendo a través de ellos. Dependiendo de su tipo se describe una forma distinta para transferir información (REFERENCIA 4):    1. Stream Socket: Son un servicio orientado a conexión donde los datos se transfieren sin encuadrarlos en registros o bloques. El protocolo de comunicaciones con streams es un protocolo orientado a conexión, ya que para establecer una comunicación utilizando el protocolo TCP, hay que establecer en primer lugar una conexión entre un par de sockets. Mientras uno de los sockets atiende peticiones de conexión (servidor), el otro solicita una conexión (cliente).    2. Datagram Socket: Son un servicio de transporte sin conexión. Es más eficiente que TCP, pero no garantiza la fiabilidad. Los datos se envían y reciben en paquetes, cuya entrega no está garantizada. Los paquetes pueden ser duplicados, perdidos o llegar en un orden diferente al que se envió. Debido a que es un protocolo sin conexión, cada vez que se envían datagramas es necesario enviar el descriptor del socket local y la dirección del socket que debe recibir el datagrama. 4. Que es un modelo peer to peer, cita un ejemplo aplicativo:   Se refiere a una red que no tiene clientes y servidores fijos, sino una serie de nodos que se comportan a la vez como clientes y servidores de los demás nodos de la red (REFERENCIA 5). Un ejemplo aplicativo de este modelo es la telefonía IP, como Skype, que permite aprovechar los recursos de la red para establecer una conexión entre dos nodos, siendo uno de ellos el cliente y el servidor del otro al mismo tiempo y viceversa.   1. Explica la estructura por medio de un ejemplo aplicativo del modelo hibrido o centralizado:   La primera generación de P2P (Napster) empleaba una estructura de red cliente-servidor. El servidor central mantiene una base de datos con información de los ficheros servidos por cada par. Cada vez que un cliente se conecta o desconecta de la red, la base de datos se actualiza. Todos los mensajes de búsqueda y control son enviados al servidor centralizado. El servidor centralizado compara la solicitud de sus clientes con el contenido de su base de datos y envía las correspondencias al cliente en cuestión. Una vez que es informado de las correspondencias, el cliente contacta con el par directamente y accede al recurso solicitado.   1. Explica la estructura por medio de un ejemplo aplicativo del modelo puro o totalmente descentralizado.   La segunda generación de P2P (Gnutella) usa un modelo distribuido donde no existe ningún servidor central y todos los nodos tienen el mismo estatus. Cada nodo actúa como servidor y como clientes en la red. Como es evidente cada par dentro de esta arquitectura trata de mantener un cierto número de conexiones con otros pares durante todo el tiempo. Este conjunto de pares conectados transporta el tráfico de red, que está conformado esencialmente por peticiones y respuestas a esas peticiones, y varios mensajes de control que facilitan el descubrimiento de otros nodos (mensajes ping en Gnutella). A pesar de que el número de saltos de la red es potencialmente infinito, permanece limitado por un tiempo de vida máximo o TTL, relacionado con el máximo número de saltos que puede dar un mensaje. Por cada nodo o par por el que circula el mensaje de petición se decrementa en una unidad el TTL descartándose la petición si esta llega a cero. El modelo P2P puro es más robusto al no depender de un servidor central, además es más económico. La principal desventaja es el elevado tiempo y sobrecarga de ancho de banda que suponen las búsquedas de información en la red. Además, puede ser que el recurso buscado y existente ni siquiera pueda ser encontrado.   1. Explica la estructura por medio de un ejemplo aplicativo del modelo mixto o semi descentralizado:   Dentro de este modelo, ciertos pares de la red son seleccionados como superpares y ayudan a gestionar el tráfico dirigido hacia otros pares. Los superpares cambian dinámicamente a medida que nuevos pares se conectan.  En este modelo cada nodo cliente mantiene sólo un pequeño número de conexiones abiertas y cada una de esas conexiones es a un superpar. Así mismo los superpares están conectados entre sí. Esta nueva topología virtual tiene el efecto de hacer la red escalable, mediante la reducción del número de nodos involucrados en el encaminamiento y manejo de los mensajes, así como la disminución del volumen de tráfico entre ellos. Además, la velocidad de respuesta a las solicitudes dentro de un entorno mixto es comparable al de un entorno P2P centralizado.   1. Realice un Comparativo entre un sistema cliente Servidor de Dos capas Vs. Uno de tres capas.  |  |  |  | | --- | --- | --- | | Arquitectura. | Dos niveles | Tres niveles | | Definición | Sistemas cliente/servidor en donde el cliente solicita recursos y el servidor responde directamente a la solicitud con sus propios recursos. | Sistemas cliente/servidor en donde existe un intermediario. El cliente solicita recursos, el intermediario proporciona los recursos solicitados mediante otro servidor y el servidor de datos proporciona al intermediario los datos que requiere.  Cada servidor realiza una tarea especializada (un servicio). Por lo tanto, un servidor puede utilizar los servicios de otros servidores para proporcionar su propio servicio. | | Ventajas | Es polivalente.  Puede responder directamente a todas las solicitudes de recursos del cliente. | Mayor grado de flexibilidad.  Mayor seguridad.  Mejor Rendimiento. |  1. ¿Qué características debe tener el motor de base de datos al montar el sistema de información?   Se debe tener en cuenta que una base de datos en un sistema distribuido debe estar formada por varias computadoras en donde se deben encontrar fragmentos o réplicas de la base de datos principal, además de poseer el sistema gestor de la base de datos y el software requerido para la comunicación con los otros nodos. Sin mencionar que debe existir la red de comunicaciones para este propósito. Cada uno de los ordenadores que forman parte de la red deben tener autonomía local. Finalmente, ningún nodo debe tener un rango superior a los demás, para evitar una centralización del sistema (REFERENCIA 6).   1. ¿Si se tiene una base de datos es distribuida cuantos servidores de aplicaciones se deben tener? Argumente la respuesta. 2. Crea un cuadro comparativo entre los diferentes modelos de cliente servidor cuales son las ventajas y desventajas de los diferentes modelos (REFERENCIA 2)  |  |  |  |  | | --- | --- | --- | --- | | Arquitectura. | Dos niveles | Tres niveles | N Niveles | | Definición | Sistemas cliente/servidor en donde el cliente solicita recursos y el servidor responde directamente a la solicitud con sus propios recursos. | Sistemas cliente/servidor en donde existe un intermediario. El cliente solicita recursos, el intermediario proporciona los recursos solicitados mediante otro servidor y el servidor de datos proporciona al intermediario los datos que requiere. | Cada servidor realiza una tarea específica, por lo que un servidor del sistema puede usar los servicios de otro servidor para poder proporcionar su propio servicio. | | Ventajas | Es polivalente.  Puede responder directamente a todas las solicitudes de recursos del cliente. | Mayor grado de flexibilidad.  Mayor seguridad.  Mejor Rendimiento. | Facilita la integración entre diferentes sistemas.  Mayor interacción con el usuario. | | Desventajas | Poca seguridad. | Costes elevados. | Dificultad de reparación.  Costes elevados. | | |
| **CAUSAS DE ERROR Y ACCIONES PARA OBTENER MEJORES RESULTADOS.** *El estudioso describirá las variables que a su consideración dificultaron el alcance de las habilidades propuestas* | |
| **CONCLUSIONES.** *El estudiante consignará en este espacio la síntesis de los logros alcanzados a través del desarrollo de la práctica de Laboratorio.*  **Los sistemas distribuidos facilitan la interacción dentro de una misma red al utilizar el modelo cliente-servidor. Esto se debe a que mediante un sistema distribuido se puede acceder a un servicio que es independiente de la plataforma por la cual se conecte el cliente. Esto quiere decir que, si quisiera acceder a un servicio que solo está disponible en una plataforma Linux, debido a que el servidor se encuentra alojado en ese sistema operativo, se puede realizar una distribución del sistema para que exista un cliente que pueda ser ejecutado en Windows, con lo que se puede acceder a dicho servicio sin tener que depender del sistema operativo por el cual se quiera acceder.** | |
| **APLICACIÓN PROFESIONAL DE LA PRÁCTICA REALIZADA.** *Exponga en este espacio la relación de las competencias y habilidades desarrolladas con el ejercicio de su futura profesión.*  En el ámbito de las bases de datos, los sistemas distribuidos que permiten el acceso a la información desde determina- dos servidores está cada vez más extendido. Un ejemplo de esto puede ser los servicios de búsqueda que ofrece Google, ya que por medio de su navegador “Google Académico”, permite acceder a las distintas bases de datos aloja- das en distintos servidores alrededor del mundo, desde un solo cliente en uso. | |
| **BIBLIOGRAFIA UTILIZADA.** *Utilizando las normas APA*  **Referencia 1:** [**http://vis.usal.es/rodrigo/documentos/sisdis/teoria/8-replicacion.pdf**](http://vis.usal.es/rodrigo/documentos/sisdis/teoria/8-replicacion.pdf) **Universidad de Salamanca, España**  **Referencia 2:** López Fuentes, Francisco de Asís Sistemas distribuidos / Francisco de Asís López Fuentes. -- México: UAM, Unidad Cuajimalpa, c2015. 200 p. : il., diagramas, col. ; 24 cm. (Una década de la Unidad Cuajimalpa de la Universidad Autónoma Metropolitana) ISBN de la Colección Una Década: 978-607-28-0452-4 ISBN de este libro: 978-607-28-0476-0.  <http://ilitia.cua.uam.mx:8080/jspui/bitstream/123456789/173/1/X56.pdf>  **Referencia 3:** <https://docentes.uaa.mx/guido/wp-content/uploads/sites/2/2015/10/Sistemas-Distribuidos-modelos.pdf> Universidad autónoma de Aguascalientes, México.  **Referencia 4:** [**https://docs.oracle.com/javase/tutorial/networking/sockets/definition.html**](https://docs.oracle.com/javase/tutorial/networking/sockets/definition.html)  **Referencia 5:** [**https://www.academia.edu/5263286/Peer\_2\_Peer\_Sistemas\_Operativos\_Distribuidos**](https://www.academia.edu/5263286/Peer_2_Peer_Sistemas_Operativos_Distribuidos)  **Referencia 6: C. . Date, Introducción a los sistemas de bases de datos, 7th ed., vol. 1. 2001**  Rúbrica para la evaluación de la práctica   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **INDIVIDUAL** | | **CRITERIOS DE EVALUACIÓN** | | | | | **NOTA** | | **Habilidad** | **Estudiante** | **0 – 1,5** | **1,6 - 2,9** | **3,0 - 3,9** | **4,0 - 4,5** | **4,6 - 5,0** | | **1. Entiende los servicios de la Interconexión de Redes (internetworking) de datos que involucra diversas redes con diferentes tecnologías de forma transparente para el usuario** | **No los identifica, no conoce las funciones de cada uno** | **Identifica algunos, no conoce las funciones** | **Identifica algunos conoce algunas funciones** | **Identifica todos, conoce algunas funciones** | **Identifica todos los mecanismos y conoce sus funciones** |  | | **2. Realizar investigaciones utilizando la bibliografía existente.** | **No utilizó bibliografía** | **Utilizó bibliografía pero no realizó las citaciones** | **Utilizó bibliografía no científica o educativa** | **La bibliografía reseñada no corresponde con las citas empleadas en la investigación** | **Uso bibliografía y realizó las citaciones correspondientes de forma adecuada** |  | | **3. Logra expresar ideas propias a partir los conocimientos que adquiere en la investigación.** | **No utiliza palabras propias ni ideas propias** | **Las ideas plasmadas son confusas, desordenas y no corresponden a al tema** | **Las ideas que expresa con coherentes, pero no corresponden al tema** | **Las ideas son coherentes pero están en desorden y no logran concluir** | **Las ideas son coherentes, ordenadas y pertenecen a la temática** |  | | **4. Forma conceptos utilizando las guías conceptuales de forma crítica.** | **No utiliza palabras propias ni ideas propias** | **No muestra una interpretación de las ideas investigadas** | **La interpretación que muestra no corresponde a la temática** | **La interpretación que muestra solo repite lo leído** | **La interpretación no se limita a los conceptos investigados, dejando claras sus ideas en torno al tema** |  | | **5. Comunica de forma verbal los resultados obtenidos en su investigación, siendo claros y concretos** | **No realiza presentación de su investigación** | **La presentación no contiene todos los conceptos involucrados en la investigación** | **La presentación está completa pero la expresión verbal no logra transmitir los conocimientos adquiridos** | **La presentación está completa pero la expresión verbal solo expresa el contenido de la presentación misma (lee la presentación)** | **La presentación está completa y la expresión verbal logra transmitir los conocimientos adquiridos** |  | | **Total** | **Total = (N1 + N2 + N3 + N4 + N5) / 5** | | | | |  | | |