TRABAJO INVESTIGATIVO LENGUAJE DE PROGRANACION 3

SANTIAGO RESTREPO ZULUAGA

UNIREMINGTON

RIONEGRO-ANTIOQUIA

**TABLA DE CONTENIDO**

[**MODELO OSI** 4](#_Toc129716789)

[**CAPA FÍSICA** 4](#_Toc129716790)

[**PROTOCOLOS** 4](#_Toc129716791)

[ **Ethernet** 4](#_Toc129716792)

[ **Token Ring** 4](#_Toc129716793)

[ **ATM** 4](#_Toc129716794)

[ **RS-232** 4](#_Toc129716795)

[ **USB:** 4](#_Toc129716796)

[**CAPA DE ENLACE DE DATOS** 5](#_Toc129716797)

[**PROTOCOLOS** 5](#_Toc129716798)

[ **PPP** 5](#_Toc129716799)

[ **HDLC** 5](#_Toc129716800)

[ **Ethernet** 5](#_Toc129716801)

[ **Wi-Fi** 5](#_Toc129716802)

[**CAPA DE RED** 5](#_Toc129716803)

[**PROTOCOLOS** 5](#_Toc129716804)

[ **IP** 5](#_Toc129716805)

[ **ICMP** 5](#_Toc129716806)

[ **ARP** 6](#_Toc129716807)

[ **OSPF** 6](#_Toc129716808)

[ **BGP** 6](#_Toc129716809)

[**CAPA DE TRANSPORTE** 6](#_Toc129716810)

[**PROTOCOLOS** 6](#_Toc129716811)

[ **TCP** 6](#_Toc129716812)

[ **UDP** 6](#_Toc129716813)

[ **SCTP** 6](#_Toc129716814)

[**CAPA DE SESIÓN** 6](#_Toc129716815)

[**PROTOCOLOS** 6](#_Toc129716816)

[ **NetBIOS** 6](#_Toc129716817)

[ **RPC** 6](#_Toc129716818)

[ **SSH** 7](#_Toc129716819)

[**CAPA DE PRESENTACIÓN** 7](#_Toc129716820)

[**PROTOCOLOS** 7](#_Toc129716821)

[ **SSL/TLS** 7](#_Toc129716822)

[ **JPEG** 7](#_Toc129716823)

[ **MPEG** 7](#_Toc129716824)

[ **ASCII** 7](#_Toc129716825)

[**CAPA DE APLICACIÓN** 7](#_Toc129716826)

[**PROTOCOLOS** 7](#_Toc129716827)

[ **HTTP** 7](#_Toc129716828)

[ **FTP** 7](#_Toc129716829)

[ **SMTP** 7](#_Toc129716830)

[ **DNS** 8](#_Toc129716831)

[**DEFINICION DE DNS** 8](#_Toc129716832)

[**DEFINICION DE URL** 8](#_Toc129716833)

[**DEFINICION DE SOCKET** 9](#_Toc129716834)

[**REFERENCIAS** 9](#_Toc129716835)

# **MODELO OSI**

El modelo OSI (Open Systems Interconnection) es un modelo de referencia de comunicación de red que se utiliza para describir cómo los datos se transfieren a través de una red de computadoras. El modelo OSI se divide en siete capas, cada una de las cuales se encarga de un conjunto específico de funciones de comunicación. A continuación, se describen brevemente cada una de las capas del modelo OSI:

## **CAPA FÍSICA:**

Esta capa se encarga de la transmisión de datos físicos a través de la red. Incluye la definición de los medios de transmisión (cable, fibra óptica, inalámbrico, etc.) y las especificaciones eléctricas, mecánicas y de señalización necesarias para la transmisión de datos.

### **PROTOCOLOS:**

* **Ethernet**: Es un protocolo de red que se utiliza comúnmente en redes de área local (LAN). Define los estándares para la transmisión de datos a través de cables de cobre y fibra óptica.
* **Token Ring*:*** Es un protocolo de red que utiliza un token para controlar el acceso a la red. Los dispositivos sólo pueden transmitir datos cuando tienen el token.
* **ATM:** Es un protocolo de red utilizado para transferir datos en alta velocidad a través de redes de área amplia (WAN).
* **RS-232:** Es un protocolo de comunicaciones serie utilizado para transmitir datos entre dispositivos, como ordenadores y módems.
* **USB:** Es un protocolo de comunicaciones utilizado para la transmisión de datos y energía eléctrica a través de cables USB.

## **CAPA DE ENLACE DE DATOS:**

Esta capa se encarga de la transmisión confiable de datos entre dispositivos conectados directamente a través de un enlace físico. Esta capa se divide en dos subcapas: control de enlace lógico (LLC) y control de acceso al medio (MAC).

### **PROTOCOLOS:**

* **PPP** (Protocolo de punto a punto): Es un protocolo de enlace de datos utilizado para establecer una conexión de red punto a punto. Se utiliza comúnmente en conexiones de línea telefónica y de banda ancha.
* **HDLC** (Control de enlace de datos de alto nivel): Es un protocolo de enlace de datos utilizado para la comunicación entre dispositivos en una red. Se utiliza comúnmente en redes de área amplia (WAN).
* **Ethernet**: Es un protocolo de red utilizado comúnmente en redes de área local (LAN). Define los estándares para la transmisión de datos a través de cables de cobre y fibra óptica.
* **Wi-Fi**: Es un conjunto de estándares de comunicación inalámbrica utilizados para la transmisión de datos a través de una red inalámbrica.
* **ARP** (Protocolo de resolución de direcciones): Es un protocolo utilizado para traducir direcciones IP en direcciones físicas de red, como direcciones MAC.

## **CAPA DE RED:**

Esta capa se encarga de enrutar los datos a través de la red. Esta capa utiliza direcciones lógicas para identificar los dispositivos en la red y utiliza algoritmos de enrutamiento para determinar la mejor ruta para los datos.

**PROTOCOLOS:**

* **IP** (Protocolo de Internet): Es un protocolo de red utilizado para la transferencia de datos entre redes. Define los estándares para la transmisión de paquetes de datos a través de redes de diferentes tipos.
* **ICMP** (Protocolo de mensajes de control de Internet): Es un protocolo utilizado para enviar mensajes de error y control en una red IP.
* **ARP** (Protocolo de resolución de direcciones): Es un protocolo utilizado para traducir direcciones IP en direcciones físicas de red, como direcciones MAC.
* **OSPF** (Protocolo de estado de enlace abierto): Es un protocolo de enrutamiento utilizado para determinar la mejor ruta para los paquetes de datos en una red.
* **BGP** (Protocolo de puerta de enlace de borde): Es un protocolo de enrutamiento utilizado para la transferencia de datos entre diferentes redes.

## **CAPA DE TRANSPORTE:**

Esta capa se encarga de la transferencia de datos extremo a extremo. Esta capa proporciona servicios de control de flujo y de error para garantizar que los datos se transmitan de manera fiable y eficiente.

### **PROTOCOLOS**:

* **TCP** (Protocolo de control de transmisión): Es un protocolo utilizado para la transferencia de datos confiable entre dispositivos en una red.
* **UDP** (Protocolo de datagramas de usuario): Es un protocolo utilizado para la transferencia de datos no confiable entre dispositivos en una red.
* **SCTP** (Protocolo de control de transporte de Stream): Es un protocolo utilizado para la transferencia de datos en tiempo real y multimedia.

## **CAPA DE SESIÓN:**

Esta capa se encarga de establecer, mantener y finalizar las conexiones entre aplicaciones que se ejecutan en dispositivos de red diferentes. Esta capa proporciona servicios para la gestión de sesiones y el control de diálogo.

**PROTOCOLOS**:

* **NetBIOS** (Servicio de soporte básico de entrada/salida de red): Es un protocolo utilizado para la comunicación entre dispositivos en una red de área local (LAN). Proporciona servicios de sesión, como el control de diálogo y el establecimiento y cierre de sesiones.
* **RPC** (Llamada a procedimiento remoto): Es un protocolo utilizado para permitir que los procesos de una red se comuniquen entre sí a través de diferentes sistemas operativos.
* **SSH** (Protocolo seguro de shell): Es un protocolo utilizado para la comunicación segura entre dispositivos de red a través de una conexión cifrada.

## **CAPA DE PRESENTACIÓN:**

Esta capa se encarga de la representación de los datos en un formato común para la comunicación entre dispositivos de red diferentes. Esta capa proporciona servicios para la conversión, la compresión y el cifrado de datos.

### **PROTOCOLOS**:

* **SSL/TLS** (Capa de sockets seguros/Protocolo de seguridad de transporte): Es un protocolo utilizado para proporcionar una conexión segura entre dispositivos en una red. Cifra los datos transmitidos para asegurar la privacidad y la integridad de los mismos.
* **JPEG** (Grupo de expertos en fotografía): Es un formato de archivo utilizado para la compresión de imágenes digitales.
* **MPEG** (Grupo de expertos en imágenes en movimiento): Es un formato de archivo utilizado para la compresión de vídeo digital.
* **ASCII** (Código estándar americano para el intercambio de información): Es un conjunto de caracteres utilizado para representar letras, números y otros símbolos en dispositivos informáticos.

## **CAPA DE APLICACIÓN:**

Esta capa se encarga de proporcionar servicios de red a las aplicaciones que se ejecutan en los dispositivos finales. Esta capa incluye protocolos de aplicación como HTTP, SMTP, FTP, etc.

### **PROTOCOLOS**:

* **HTTP** (Protocolo de transferencia de hipertexto): Es un protocolo utilizado para la transferencia de datos entre servidores web y navegadores web.
* **FTP** (Protocolo de transferencia de archivos): Es un protocolo utilizado para la transferencia de archivos entre dispositivos en una red.
* **SMTP** (Protocolo simple de transferencia de correo): Es un protocolo utilizado para la transferencia de correo electrónico entre servidores de correo electrónico.
* **DNS** (Sistema de nombres de dominio): Es un protocolo utilizado para la resolución de nombres de dominio en direcciones IP.

# **DEFINICION DE DNS:**

DNS son las siglas en inglés de "Domain Name System", que se traduce al español como "Sistema de Nombres de Dominio". Es un sistema informático que se encarga de traducir los nombres de dominio (como, por ejemplo, google.com) en direcciones IP (como 172.217.6.174), que son las direcciones numéricas que identifican a los servidores en internet.

El DNS actúa como una especie de "libro de teléfonos" de internet, permitiendo que los usuarios puedan acceder a sitios web o servicios en línea utilizando nombres de dominio fáciles de recordar en lugar de tener que recordar direcciones IP complejas. Cuando un usuario ingresa un nombre de dominio en su navegador web, el navegador envía una solicitud al sistema DNS para obtener la dirección IP correspondiente. Luego, el DNS responde con la dirección IP, permitiendo que el navegador se conecte al servidor web que aloja el sitio solicitado.

# **DEFINICION DE URL:**

URL son las siglas en inglés de "Uniform Resource Locator", que se traduce al español como "Localizador Uniforme de Recursos". Es una dirección web que identifica la ubicación de un recurso en internet, como una página web, una imagen, un archivo de video, un documento o cualquier otro tipo de contenido en línea.

Una URL está compuesta por varias partes, incluyendo el protocolo utilizado para acceder al recurso (como HTTP, HTTPS, FTP, etc.), el nombre de dominio del servidor donde se encuentra el recurso (como google.com), la ruta del recurso en el servidor (como /search?q=URL), y cualquier otro parámetro o fragmento necesario para acceder al recurso.

Por ejemplo, la URL completa para acceder a la página principal de Google es: https://www.google.com/.Las URL son esenciales para acceder a recursos en línea de manera efectiva y eficiente, y son utilizadas por navegadores web, aplicaciones y dispositivos en todo el mundo para acceder a contenidos en la web.

# **DEFINICION DE SOCKET:**

Un socket (en español, "conector" o "enchufe") es una abstracción de software utilizada para establecer una conexión de red entre dos programas o procesos que se ejecutan en diferentes dispositivos o en la misma máquina.

Básicamente, un socket es un punto final de una comunicación bidireccional entre dos programas, que permite enviar y recibir datos entre ellos. Los programas pueden usar sockets para establecer una conexión de red y comunicarse entre sí a través de un protocolo de red, como TCP (Protocolo de Control de Transmisión) o UDP (Protocolo de Datagramas de Usuario).

Cada socket está identificado por un número único, que se llama puerto, que puede estar asociado a una dirección IP. Los programas pueden conectarse a un socket específico para enviar y recibir datos, y pueden enviar y recibir mensajes de forma simultánea o en serie.

Los sockets son ampliamente utilizados en aplicaciones de red, como navegadores web, programas de correo electrónico, juegos en línea, mensajería instantánea, servidores web, entre otros.

# **REFERENCIAS**

"Information Technology - Open Systems Interconnection - Basic Reference Model: The Basic Model". International Organization for Standardization (ISO), ISO/IEC 7498-1:1994, 1994.

"TCP/IP Illustrated, Volume 1: The Protocols". Autor: W. Richard Stevens, Publicado por Addison-Wesley Professional, 1993.

"Computer Networks: A Top-Down Approach". Autores: James F. Kurose, Keith W. Ross, Publicado por Pearson Education, 2016.

# **CONCLUSIONES PRACTICA CHAT JAVA**

* El programa implementado es un chat en Java utilizando sockets y POO.
* En este programa, el servidor es responsable de escuchar y aceptar conexiones de múltiples clientes, manteniendo una lista de hilos de clientes activos. Cada hilo de cliente se encarga de manejar la comunicación bidireccional entre el servidor y el cliente correspondiente.
* Por otro lado, el cliente se encarga de enviar mensajes al servidor y recibir los mensajes enviados por otros clientes conectados al servidor.
* En resumen, este programa demuestra cómo se puede implementar un chat básico utilizando sockets y POO en Java, utilizando una arquitectura cliente-servidor, permitiendo que múltiples usuarios se comuniquen entre sí en tiempo real.