

Generación de servicios en red (Programación de servicios y procesos)

Los procesos en segundo plano que realizan distintas funciones, algunas relacionadas con el sistema operativo y otras no, se denominan servicios, y se están ejecutando permanentemente en el sistema.

Un servicio está formado por los siguientes elementos:

- Estructura: componentes hardware y software que lo forman.
- Función: aquello para lo que está pensado el sistema, es decir, para lo que sirve y/o se usa.

Los servicios se pueden iniciar, detener, pausar, reanudar, etc. Estas acciones, normalmente, sólo las puede realizar el administrador de forma local o remota. Un ejemplo de servicio de sistema es el servicio de escritorio remoto, que permite conectarse remotamente al equipo.

Protocolos estándar de comunicación en red a nivel de aplicación (telnet, ftp, http, pop3, smtp, entre otros)

La capa de aplicación define las aplicaciones de red y los servicios de Internet estándar que puede utilizar un usuario. Estos servicios utilizan la capa de transporte para enviar y recibir datos.

Existen varios protocolos de capa de aplicación. En la lista siguiente se incluyen ejemplos de protocolos de capa de aplicación, algunos ejemplos son:

- Servicios TCP/IP estándar como los comandos ftp, tftp y telnet.
- Servicios de nombres, como NIS o el sistema de nombre de dominio (DNS).
- Servicios de directorio (LDAP).

Librerías de clases y componentes

La programación utilizando clases y componentes dependen del sistema operativo y el lenguaje de programación, hay múltiples librerías de clases para programar aplicaciones en red.

Ejercicios

- [Realizar petición HTTP mediante el método GET](#)
 - [Realizar petición HTTP mediante el método POST](#)
 - [Realizar petición HTTP mediante el método POST utilizando librerías de Apache](#)
-

Utilización de objetos predefinidos

Dependiendo de las librerías que se utilicen se hará uso de objetos y métodos propios de las clases.

Ejercicio

- [Realizar una suma mediante una interfaz que contiene los métodos que se ejecutarán de forma remota \(RMI\)](#)
-

Establecimiento y finalización de conexiones

Las librerías de clases proporciona de manera transparente el proceso correcto para el establecimiento y finalización de conexiones.

Transmisión de información

Toda la transmisión de información es controlada por la biblioteca, liberando al programador de tareas de control.

Programación de aplicaciones cliente

Las aplicaciones cliente se pueden programar mediante interfaces de texto o interfaces gráficos.

Programación de servidores

Las aplicaciones servidores se pueden programar mediante interfaces de texto o interfaces gráficos.

Más información

- <https://www.jesusninoc.com/03/17/enviar-un-sonido-entre-un-cliente-y-un-servidor-con-powershell/>
- <https://www.jesusninoc.com/03/17/crear-una-tarea-programada-en-la-que-un-servidor-escucha-los-mensajes-que-son-mandados-por-parte-de-los-cliente-con-powershell/>
- <https://www.jesusninoc.com/03/17/crear-una-tarea-programada-en-la-que-un-servidor-escucha-los-mensajes-que-son-mandados-por-parte-de-los-cliente-con-powershell-la-comunicacion-entre-el-servidor-y-el-cliente-se-realiza->

Implementación de comunicaciones simultáneas

Mediante el uso de hilos se llevan a cabo comunicaciones simultáneas.