



CONTINGUTS

1. Programación multiproceso:

- Ejecutables. Procesos. Servicios.
- Estados de un proceso.
- Planificación de procesos por el sistema operativo.
- Hilos.
- Programación concurrente.
- Programación paralela y distribuida.
- Creación de procesos.
- Comunicación entre procesos.
- Gestión de procesos.
- Comandos para la gestión de procesos en sistemas libres y propietarios.
- Herramientas gráficas para la gestión de procesos en sistemas libres y propietarios.
- Sincronización entre procesos.
- Programación de aplicaciones multiproceso.
- Documentación.
- Depuración.

2. Programación multihilo:

- Recursos compartidos por los hilos.
- Estados de un hilo. Cambios de estado.
- Elementos relacionados con la programación de hilos. Librerías y clases.
- Gestión de hilos.
- Creación, inicio y finalización.
- Sincronización de hilos.
- Información entre hilos. Intercambio.
- Prioridades de los hilos.
- Gestión de prioridades.
- Programación de aplicaciones multihilo.
- Documentación.
- Depuración.

3. Programación de comunicaciones en red:

- Protocolos de comunicaciones.
- Comunicación entre aplicaciones.



- Roles cliente y servidor.
- Elementos de programación de aplicaciones en red. Librerías.
- Funciones y objetos de las librerías.
- Sockets.
- Tipos de sockets. Características.
- Puertos de comunicaciones.
- Creación de sockets.
- Enlazado y establecimiento de conexiones.
- Utilización de sockets para la transmisión y recepción de información.
- Programación de aplicaciones cliente y servidor.
- Optimización de sockets.
- Utilización de hilos en la programación de aplicaciones en red.
- Depuración.
- Monitorización de tiempos de respuesta.

4. Generación de servicios en red:

- Protocolos estándar de comunicación en red a nivel de aplicación (telnet, ftp, http, pop3, smtp, entre otros).
- Librerías de clases y componentes.
- Funciones y objetos de las librerías.
- Utilización de objetos predefinidos.
- Propiedades de los objetos predefinidos.
- Métodos y eventos de los objetos predefinidos.
- Establecimiento y finalización de conexiones.
- Transmisión de información.
- Programación de aplicaciones cliente.
- Programación de servidores.
- Implementación de comunicaciones simultáneas.
- Documentación.
- Depuración.
- Monitorización de tiempos de respuesta.

5. Utilización de técnicas de programación segura:

- Prácticas de programación segura.
- Control de accesos.
- Limitación de privilegios.
- Validación de entradas.
- Criptografía de clave pública y clave privada.
- Principios criptográficos.



- Principales aplicaciones de la criptografía.
- Protocolos criptográficos.
- Política de seguridad.
- Usuarios.
- Roles.
- Programación de mecanismos de control de acceso.
- Encriptación de información.
- Protocolos seguros de comunicaciones.
- Sockets seguros.
- Programación de aplicaciones con comunicaciones seguras.
- Prueba y depuración.

CRITERIS D'AVUACIÓ

1.- Desarrolla aplicaciones compuestas por varios procesos reconociendo y aplicando principios de programación paralela.

Criterios de evaluación:

- a. Se han analizado las características de los procesos y de su ejecución por el sistema operativo.
- b. Se han caracterizado los hilos de ejecución y descrito su relación con los procesos.
- c. Se han reconocido las características de la programación concurrente y sus ámbitos de aplicación.
- d. Se han identificado las diferencias entre programación paralela y programación distribuida, sus ventajas e inconvenientes.
- e. Se han utilizado clases para programar aplicaciones que crean subprocesos.
- f. Se han utilizado mecanismos para sincronizar y obtener el valor devuelto por los subprocesos iniciados.
- g. Se han desarrollado aplicaciones que gestionen y utilicen procesos para la ejecución de varias tareas en paralelo.
- h. Se han depurado y documentado las aplicaciones desarrolladas.

2.- Programa mecanismos de comunicación en red empleando sockets y analizando el escenario de ejecución.

Criterios de evaluación:

- a. Se han identificado escenarios que precisan establecer comunicación en red entre varias aplicaciones.
- b. Se han identificado los roles de cliente y de servidor y sus funciones asociadas.



- c. Se han reconocido librerías y mecanismos del lenguaje de programación que permiten programar aplicaciones en red.
- d. Se ha analizado el concepto de socket, sus tipos y características.
- e. Se han utilizado sockets para programar una aplicación cliente que se comunique con un servidor.
- f. Se ha desarrollado una aplicación servidor en red y verificado su funcionamiento.
- g. Se han desarrollado aplicaciones que utilizan sockets para intercambiar información.
- h. Se han utilizado hilos para implementar los procedimientos de las aplicaciones relativos a la comunicación en red.

3.- Programa mecanismos de comunicación en red empleando sockets y analizando el escenario de ejecución.

Criterios de evaluación:

- a. Se han identificado escenarios que precisan establecer comunicación en red entre varias aplicaciones.
- b. Se han identificado los roles de cliente y de servidor y sus funciones asociadas.
- c. Se han reconocido librerías y mecanismos del lenguaje de programación que permiten programar aplicaciones en red.
- d. Se ha analizado el concepto de socket, sus tipos y características.
- e. Se han utilizado sockets para programar una aplicación cliente que se comunique con un servidor.
- f. Se ha desarrollado una aplicación servidor en red y verificado su funcionamiento.
- g. Se han desarrollado aplicaciones que utilizan sockets para intercambiar información.
- h. Se han utilizado hilos para implementar los procedimientos de las aplicaciones relativos a la comunicación en red.

4.- Desarrolla aplicaciones que ofrecen servicios en red, utilizando librerías de clases y aplicando criterios de eficiencia y disponibilidad.

Criterios de evaluación:

- a. Se han analizado librerías que permitan implementar protocolos estándar de comunicación en red.
- b. Se han programado clientes de protocolos estándar de comunicaciones y verificado su funcionamiento.
- c. Se han desarrollado y probado servicios de comunicación en red.
- d. Se han analizado los requerimientos necesarios para crear servicios capaces de gestionar varios clientes concurrentes.
- e. Se han incorporado mecanismos para posibilitar la comunicación simultánea de varios clientes con el servicio.
- f. Se ha verificado la disponibilidad del servicio.
- g. Se han depurado y documentado las aplicaciones desarrolladas.

5.- Protege las aplicaciones y los datos definiendo y aplicando criterios de seguridad en el acceso, almacenamiento y transmisión de la información.

Criterios de evaluación:

- a. Se han identificado y aplicado principios y prácticas de programación segura.
- b. Se han analizado las principales técnicas y prácticas criptográficas.



- c. Se han definido e implantado políticas de seguridad para limitar y controlar el acceso de los usuarios a las aplicaciones desarrolladas.
- d. Se han utilizado esquemas de seguridad basados en roles.
- e. Se han empleado algoritmos criptográficos para proteger el acceso a la información almacenada.
- f. Se han identificado métodos para asegurar la información transmitida.
- g. Se han desarrollado aplicaciones que utilicen sockets seguros para la transmisión de información.
- h. Se han depurado y documentado las aplicaciones desarrolladas.

BIBLIOGRAFIA