Análisis de resultados

Pruebas realizadas

Problemas presentados

Estado del proyecto.

Portada completa

abstract en inglés

Introducción

Muchas veces, es necesario en un programa del sistema operativo, transferir informacion a otro programa. Cuando se tiene una necesidad como esa, es comun recurrir a los sockes; en un sistema operativo, un socket se puede ver como la comunicacion entre dos programas que buscan intercambiar informacion para lograr una tarea.

Algunos conceptos importantes a tomar en consideración:

1. **Host**: es una dirección única dentro de la red donde se encuentra el proceso a conectar.
2. **Numero de puerto**: Número que identifica a la aplicación dentro de una computadora conectada a la red. Dos sockets no pueden enviar o recibir información por el mismo puerto al mismo tiempo dentro de un mismo SO, sin que uno sea Cliente y el otro el Servidor.
3. **Flujo de entrada y salida:** mediante estos flujos se puede enviar y recibir información como una secuencia de bytes o caracteres una vez que el socket esté conectado y haya sido abierto.
4. **Bloqueante/No Bloqueante:** Los sockets por defecto son bloqueantes, es decir bloquean el proceso donde se encuentran alojados hasta que la información sea enviada o recibida por completo. En muchas ocasiones es posible configurar esta característica, sin embargo esto no garantiza que la información sea enviada o recibida totalmente. Por ejemplo si un socket es configurado para no ser bloqueante e intenta enviar muchos bytes de información. El socket escribirá tantos bytes como le sea posible sin esperar a que toda la información sea transmitida en el flujo de salida y retorna inmediatamente el control al proceso que lo invoco.

Dicho lo

El código fuente debe estar completo y explicado en un anexo de dicho documento.

+ Deberá enviar un correo antes de la revisión en clase con el PDF de la documentación y en un archivo comprimido aparte el código fuente completo de su servidor y de su cliente.

No incluya archivos ejecutables