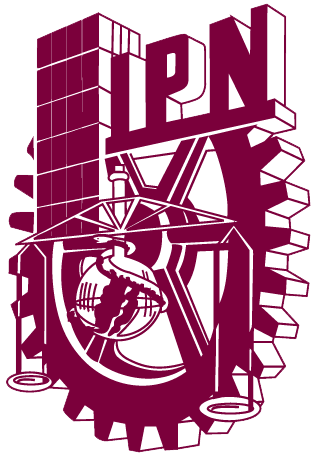


**Instituto Politécnico Nacional**

**Escuela Superior de Cómputo**

**Aplicaciones para Comunicaciones de Red**

**Práctica 1**

**Servicio de Transferencia de Archivos en Servidor**

**Miembros:**

Benítez Ramírez Sergio

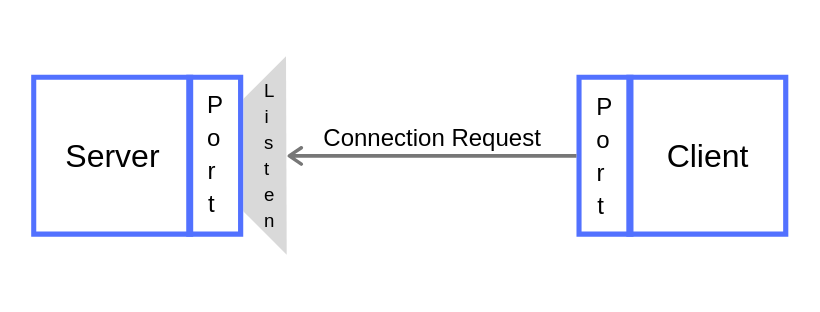
Monteros Cervantes Miguel Angel

3CM15

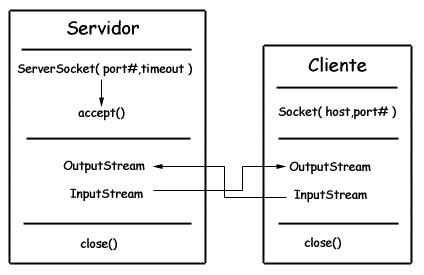
**Introducción**

Sockets

Un socket (enchufe), es un método para la comunicación entre un programa del cliente y un programa del servidor en una red, se define, por tanto, como el punto final en una conexión.



Este mecanismo surge a principios de los 80 con el sistema Unix de Berkeley, para proporcionar un medio de comunicación entre procesos y presentan la misma funcionalidad que tiene la comunicación por correo o por teléfono (de un buzón se extraen mensajes completos, mientras que el teléfono permite el envío de flujos de información que no tienen una estructura claramente definida), es decir permiten que un proceso hable (emita o reciba información) con otro incluso estando estos en distintas máquinas. Esta característica de ínter conectividad hace que el concepto de socket sea de gran utilidad.



Funcionamiento

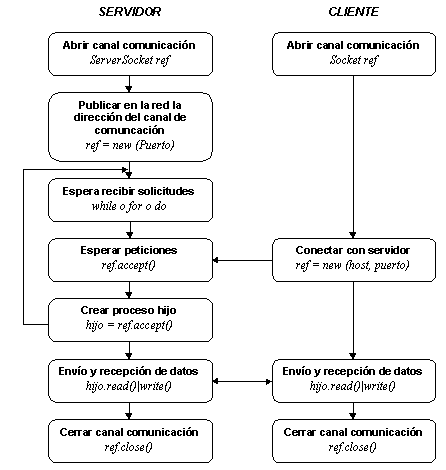
Un socket queda definido por un par de direcciones IP local y remota, un protocolo de transporte y un par de números de puerto local y remoto. Para que dos programas puedan comunicarse entre sí es necesario que se cumplan ciertos requisitos:

* Que un programa sea capaz de localizar al otro.
* Que ambos programas sean capaces de intercambiarse cualquier secuencia de octetos, es decir, datos relevantes a su finalidad.

Para ello son necesarios los tres recursos que originan el concepto de socket:

* Un protocolo de comunicaciones, que permite el intercambio de octetos.
* Un par de direcciones del Protocolo de Red (Dirección IP, si se utiliza el Protocolo TCP/IP), que identifica la computadora de origen y la remota.
* Un par de números de puerto, que identifica a un programa dentro de cada computadora.

Los sockets permiten implementar una arquitectura cliente-servidor o peer to peer. La comunicación debe ser iniciada por uno de los programas que se denomina programa cliente. El segundo programa espera a que otro inicie la comunicación, por este motivo se denomina programa servidor.

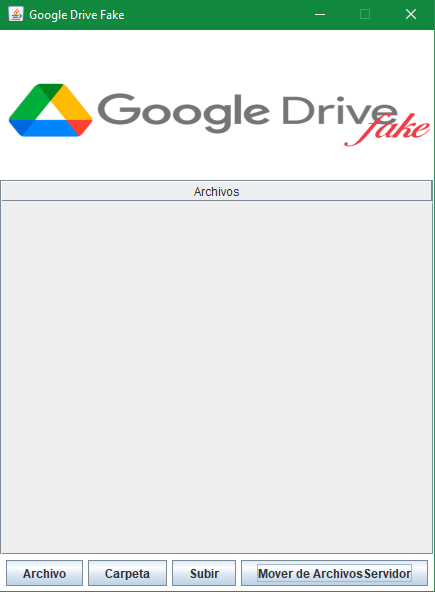


**Desarrollo**

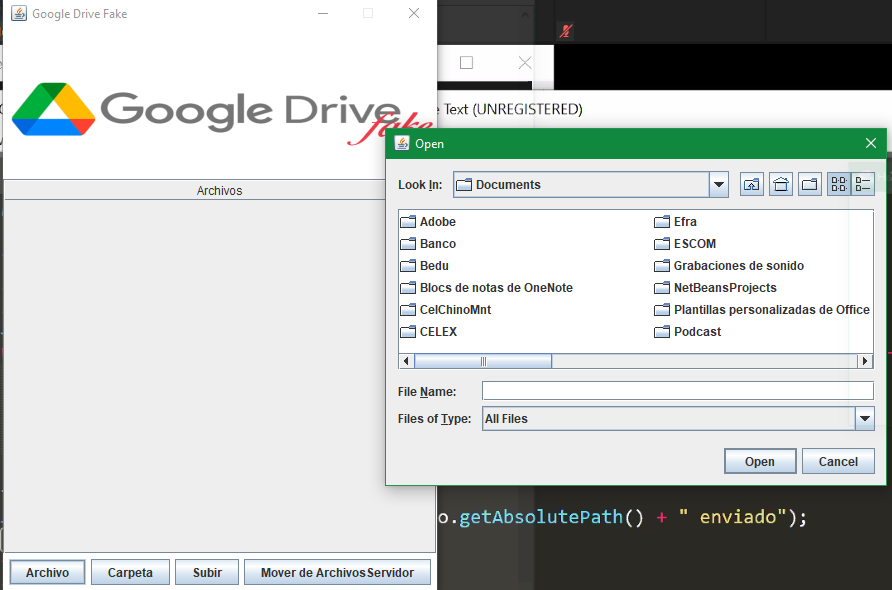
Para la elaboración de esta práctica se hizo uso del lenguaje de programación Java al igual que una serie de paquetes para el uso correcto de los sockets que intervienen entre la comunicación del cliente y el servidor.

Esta práctica consiste en crear un servicio de transferencia de archivos a un servidor tipo Google Drive o Dropbox.

Lo primero que se tiene es una pantalla en la cual se le mostrará al usuario las opciones a elegir, ya sea subir un archivo, una carpeta o mover los archivos dentro del servidor.



Posterior a esto se tiene la posibilidad de entrar a la pantalla donde se visualizarán los archivos locales en tu computadora y elegir el que deseas subir al servidor.



Una vez elegido, damos click en “Open” y será cuestion de esperar a que el archivo o carpeta sea transferido al servidor. 

Ya que se hayan subido los archivos y/o carpetas en el servidor, tendremos la posibilidad de mover de dirección los archivos dentro de la carpeta del servidor.

**Conclusiones**

Con esta práctica realizada podemos llegar a la conclusión de que con una arquitectura básica de conexión con sockets podemos implementar una aplicación completa y totalmente productiva, que podrá facilitar el uso y transferencia de archivos de manera remota, dentro y fuera de nuestro directorio local, también podemos destacar el uso de los métodos que la paqueteria nos proporciona para mayor facilidad en la manipulación de los sockets.

**Referencias.**

¿Qué es un Socket?

[¿Qué es un socket? (speedcheck.org)](https://www.speedcheck.org/es/wiki/socket/)

Socket

[Socket - EcuRed](https://www.ecured.cu/Socket)

Arquitectura Cliente Servidor

[Arquitectura cliente/servidor (uniovi.es)](http://isa.uniovi.es/domotica/Temas/T6/T6-ClienteServidor.htm)