

INFORME DE ELABORACION

DIAGRAMA DE COMUNICACIONES:

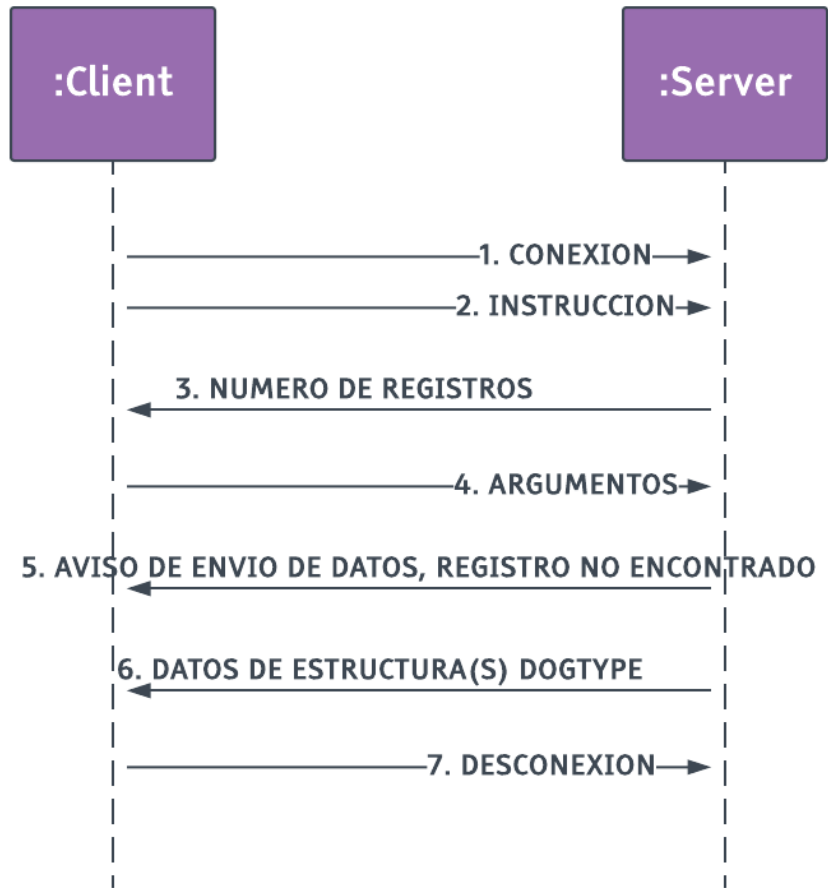
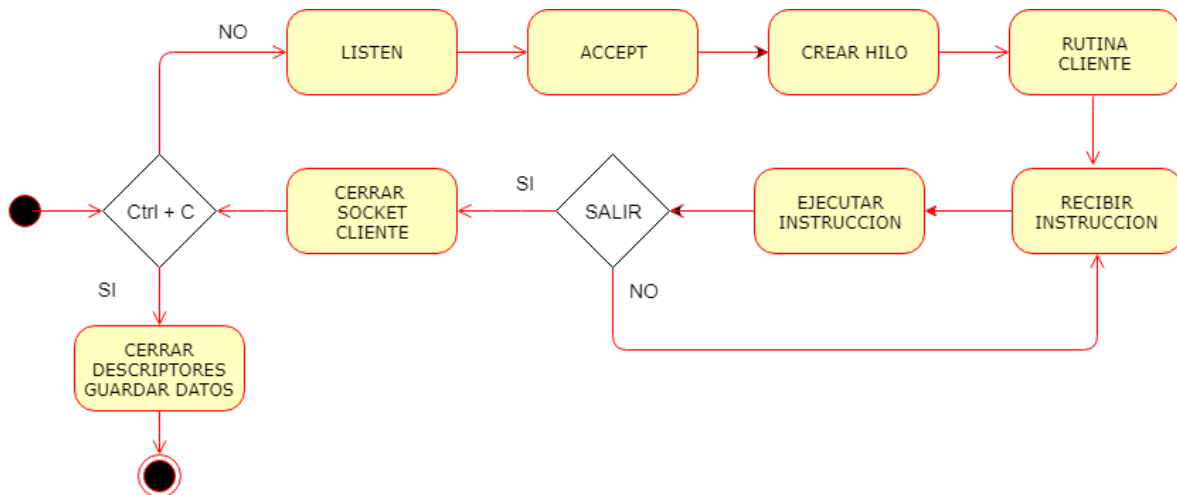


DIAGRAMA DE BLOQUES:



FUNCIONAMIENTO GENERAL:

El programa servidor esta continuamente escuchando pendiente de una solicitud de conexión, cada vez que se acepta una conexión con un nuevo cliente se crea un hilo que ejecuta una rutina que se encarga de administrar todas las peticiones del cliente, así como, de registrarlas en un archivo. Todos los procesos acceden y modifican un archivo único almacenado en disco, lo que les permite a todos compartir información. Todos los hilos creados se encuentran sincronizados por lo que no se presentaran conflictos.

FUNCIONES:

- **SERVIDOR:**
 - **shutdownHandler:** Se activa cuando se detiene la ejecución del proceso servidor mediante Ctrl+C. Guarda los datos modificados y cierra los archivos en disco usados por el programa. Se encarga de realizar los joins de todos los hilos lanzados para que el proceso padre no termine antes de que hallan terminado todos los hilos. Por último, cierra el socket del servidor.
 - **atenderCliente:** Es la rutina que ejecuta cada hilo. Recibe como argumento el descriptor del socket para identificar al cliente con el que se esta comunicando. Su función es recibir los comandos del cliente y lanzar los procedimientos que se encargan de llevarlas a cabo. También registra los logs de cada operación, por lo que, recibe del Accept en el Main la dirección de cada cliente y la escribe junto con el nombre, la fecha y hora de la operación en el archivo serverDogs.log en el momento en que se recibe cada instrucción.
- **CLIENTE, SERVIDOR:**
 - **readString:** Asegura que todos los datos enviados sean recibidos por el proceso que los requiere, al ejecutar un 'rcv()' por cada carácter de cada cadena enviada. Teniendo en cuenta que, la función 'rcv()' puede leer menos datos de los requeridos de los recibidos en

el buffer y por lo que a veces es necesario ejecutarla varias veces para recibir los datos completos.

MANUAL DE USO Y ESPECIFICACIONES

Antes que todo, para que el programa servidor pueda funcionar se requiere haber generado antes los archivos: registroName.bin, tails.bin y dataDogs.dat. Dichos archivos se deben encontrar en la misma carpeta en donde se encuentran los programas servidor y cliente y el archivo makefile.

Para poner en funcionamiento el programa servidor y los programas cliente se debe Introducir en la terminal el comando make, este lanzara las terminales para el servidor y para los clientes y ejecutara cada uno de los procesos.

Luego, para usar las funcionalidades de los programas cliente, simplemente se debe, introducir en la terminal el numero de la opción de la operación que se quiere realizar sobre la base de datos. Dependiendo de la opción elegida el programa le pedirá los datos necesarios y después, mostrará los resultados correspondientes.

Para terminar la ejecución del programa cliente basta con ingresar el numero cinco, que corresponde a la opción de Salir y la ventana se cerrara automáticamente.

Respecto al programa Servidor no se requiere realizar ninguna acción, aparte de la de presionar las teclas Ctrl+C al mismo tiempo para terminar su ejecución.

El programa servidor generara un archivo llamado serverDogs.log que contendrá el registro de cada una de las acciones solicitadas por cada programa cliente con su respectiva dirección IP y la hora y fecha de la realización. Para consultar dichos datos, basta con ejecutar el siguiente comando en la terminal: './consultarLogs.c', acto seguido se mostraran en pantalla cada uno de los registros.

Sincronización:

Todos los métodos para exclusión mutua entre hilos se encuentran en el mismo código fuente. Para cambiar el método de sincronización se debe descomentar el código del método a habilitar y comentar las instrucciones del método que se quiere deshabilitar.