

Cátedra de Sistemas Operativos

Programación de Sockets

Versión 1.0

Trabajo Práctico Final

1. Introducción

El objetivo de este trabajo es familiarizarse con el uso de sistemas cliente-servidor concurrentes, implementando el servidor mediante la creación de procesos hijos o mediante la creación de threads. Al mismo tiempo, ejercitar el uso de un tipo de primitiva de sincronización y comunicación de procesos (IPC): el uso de sockets.

2. Objetivos

- Adquirir los conocimientos básicos necesarios para poder utilizar primitivas de sincronización y comunicación de procesos en ANSI C.
- Conocer y comprender las distintas formas de comunicación entre procesos mediante sockets.
- Generar una aplicación cliente-servidor que atienda clientes en simultáneo garantizando la integridad de los datos.

3. Desarrollo

3.1. Condiciones

3.1.1. Se deberá desarrollar un sistema cliente/servidor, de manera que el servidor pueda atender a varios clientes simultáneamente (servidor concurrente).

3.1.2. Puede considerarse que el servidor y los clientes residen en distintas terminales de la misma máquina.

3.1.3. Se deberán implementar tanto el código del servidor como el del cliente del servicio.

3.1.4. El servicio a ofrecer puede ser uno de los siguientes o se puede ofrecer alguno similar a elección del grupo:

- Reserva de asientos de avión
- Reserva de habitaciones de hotel
- Reserva de asientos de sala de cine

3.1.5. La implementación debe incluir una interfaz que le permita al cliente realizar tanto funciones de usuario del sistema como funciones de administrador del sistema. En caso de elegir el sistema de reserva de asientos de avión el menú que debe mostrar el cliente será:

- **Consultar el estado de un vuelo:** se mostrará la disposición de asientos del vuelo (DAV) indicado, donde en cada asiento se deberá indicar si el mismo está disponible o reservado.
- **Realizar una reserva de un vuelo:** se mostrará la DAV al momento para el vuelo indicado, se le pide qué asiento desea reservar, y luego se mostrará la DAV después del intento de realización de la reserva.
- **Cancelar una reserva:** se le mostrará la DAV al momento para el vuelo indicado, se le pide qué asiento desea cancelar, y luego se mostrará la DAV después del intento de cancelación de la reserva.
- **Alta y Baja de vuelos:** Permitir desde el cliente el alta y baja de vuelos. Al momento de realizar el alta de un vuelo este cuenta con todos sus asientos disponibles.

En caso de elegir otro sistema tomar este ejemplo para ofrecer la misma funcionalidad indicada.

3.1.6. Desarrollar un sistema tolerante a fallos. Tomando el ejemplo anterior contemplar por ejemplo casos de que un cliente quiera reservar un asiento que ya está reservado, que quiera reservar un asiento de un vuelo que fue cancelado.

3.1.7. Para el acceso y modificación de la base de datos del servidor se deberá utilizar la librería **SQLite**. Con esta librería se deberá mantener una base de datos en archivo que contenga toda la información relevante del sistema, incluyendo:

- Qué asientos/habitaciones están disponibles/ocupados
- Listado de clientes que realizaron reservas
- Listado de clientes que cancelaron reservas

3.1.5. Todo punto no explícito en este documento podrá ser interpretado a conveniencia del alumno, siempre dentro de los márgenes del sentido común. Ante la duda, consultar a los docentes o enviar un mail a la cátedra.

3.2. Material a entregar

Cada grupo deberá entregar todos los fuentes del trabajo debidamente comentados, junto a un makefile para compilarlos en un ambiente Linux. Además deberán entregar un informe impreso detallando las decisiones respecto a los ítems que quedaron a elección del grupo, problemas presentados y solución de los mismos durante la realización del trabajo.

4. Puntos a evaluar

Los puntos a evaluar del trabajo son:

- Uso de sockets, procesos en paralelo, y/o threads
- Calidad del código y testeos
- Funcionamiento
- Informe, Manual, y presentación

5. Grupos

Los grupos podrán tener hasta 3 integrantes.

6. Entrega

Todos los integrantes del grupo deben inscribirse en **alguna de las dos fechas de final de febrero (del 5/2/2018 al 15/2/2017, o bien del 16/2/2018 al 24/2/2018)** por SGA.

La entrega del código fuente y el informe deberá hacerse **hasta 24 horas antes** del examen final a través de la actividad Trabajo Práctico Final en Campus ITBA.

El coloquio del trabajo práctico final será **en el día y hora de la fecha del final** en el aula indicada por SGA.