

PROTOCOLO DE COMUNICACIONES SOCKET PARA GESTIÓN DE RESERVAS A TRAVÉS DE INTERNET.

I. Ruano Ruano, R. Mata Campos, M^a D. Molina González, P. Muñoz Palomino

Departamento de Electrónica
Universidad de Jaén
alonso@ujaen.es

RESUMEN

Este trabajo muestra las comunicaciones realizadas por un programa de gestión hotelera, los formatos de los mensajes utilizados y la forma de realizar las transferencias de información. El programa forma parte de un modelo de comunicaciones empleado para la gestión de reservas a través de Internet siguiendo el modelo de cliente-servidor. Dichos mensajes se generan en un entorno mixto en el que se combina la tecnología *Web* con las aplicaciones cliente-servidor personalizadas que hacen uso de *winsockets* (objeto del presente trabajo). En primer lugar se va a presentar el modelo de comunicaciones completo: las aplicaciones involucradas en los distintos clientes y servidores y los servicios que prestan cada uno de ellos. A continuación el trabajo se centrará en las comunicaciones empleadas utilizando *winsockets*, concretamente en el estudio de los mensajes de control que intercambian las aplicaciones finales para realizar las transferencias de información relativa a la gestión de las reservas y terminará mostrando los resultados del mismo.

1. INTRODUCCIÓN

El fenómeno Internet sigue creciendo sin pausa, todas las empresas que pretendan dar un servicio al público lo tienen en cuenta y lo utilizan para darse a conocer y, si su negocio lo permite, realizar todo tipo de operaciones a distancia con el ahorro que ello conlleva. El sector hotelero se está integrando a las TICs (tecnologías de la Información y Comunicaciones), permitiendo hoy en día a algunas empresas realizar reservas, consultas, cancelaciones o cualquier otro tipo de gestión a través de Internet. Este es el marco elegido para el desarrollo mostrado en este trabajo.

1.1. Marco del desarrollo mostrado.

El esquema del marco mencionado en el punto anterior se muestra en la Fig. 1. Se trata de un sistema que permite la gestión de reservas a través de Internet en tiempo real proporcionando una respuesta inmediata a la gestión realizada. Los usuarios del servicio pueden realizar la consulta, realización y modificación de reservas a través de un *browser* HTML (*Hypertext Markup Language*) genérico y obtener una respuesta de forma casi inmediata. El proceso se desarrolla como se describe a continuación: el cliente, a través del *browser*, se pone en contacto

siguiendo el protocolo HTTP (*Hypertext Transport Protocol*) [1] con la estación donde se tiene instalado el servidor *Web* (Conexión 1), de forma que recibe del servidor HTTP una página *Web* con un formulario que tras ser rellenado por el cliente origina el envío de los datos introducidos a la estación. En la estación (Central de reservas) su recepción provoca la ejecución de un programa CGI (*Common Gateway Interface*) que con los parámetros obtenidos del cliente se pone en comunicación (como un cliente *winsocket*) [2] con el hotel seleccionado para realizar la acción pedida (mediante protocolo propio), tras lo cual manda una respuesta indicando el resultado obtenido al programa CGI. Este genera entonces una página HTML que es enviada al cliente mostrando el resultado de la operación.

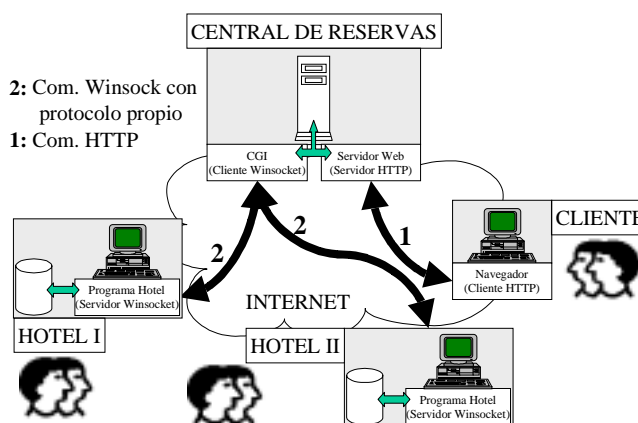


Figura 1. Esquema del marco de desarrollo.

2. COMUNICACIONES CON SOCKETS

El presente trabajo se centra en las comunicaciones existentes entre los modelos de hotel, que actúan en las mismas como servidores *winsocket*, y el programa CGI de la estación central de reservas, que actúa como cliente *winsocket*, tal y como muestra la Fig. 1 (2: Comunicaciones *winsocket*). La forma de tratar y almacenar los datos (empleo de Bases de datos o estructuras de datos propias) por cada modelo de hotel es independiente de las comunicaciones por lo que el trabajo mostrado en este artículo no se ve afectado por la implementación elegida.

2.1. Operaciones Básicas

Las operaciones que provocan el uso de comunicaciones entre el cliente y servidor socket son las siguientes: Realización de reserva, Cancelación de reserva, Consulta de reserva y Modificación de reserva.

La Fig. 2 muestra los diagramas de transferencia de información implicados en cada uno de los procesos que se pueden producir.

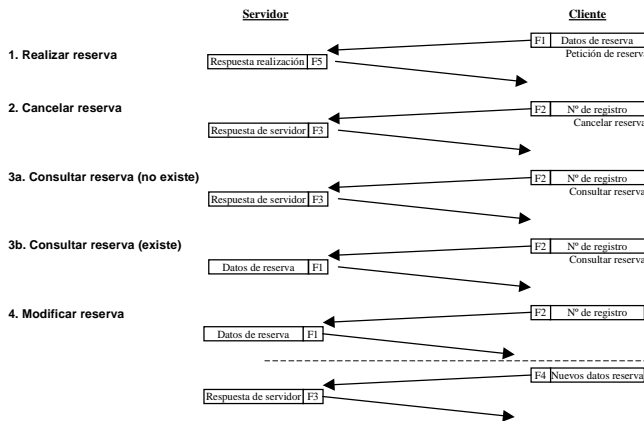


Figura 2. Diagrama de transferencia de mensajes.

En la Fig.2 se puede comprobar que para llevar a cabo cualquier operación es necesario que se realice el intercambio de, como mínimo, un mensaje en cada dirección. Cuando el servidor está activo (se puede activar o desactivar desde el programa) está siempre en modo escucha esperando que se realice una conexión. Al utilizar sockets no bloqueantes el programa servidor no está inoperante mientras se encuentra a la escucha de una conexión entrante por lo que puede realizar otras operaciones a nivel local (comprobación de reservas realizadas a través de Internet, activación de reservas de clientes que llegan al hotel y acceden a las habitaciones, realización de reservas en el mismo hotel, etc...).

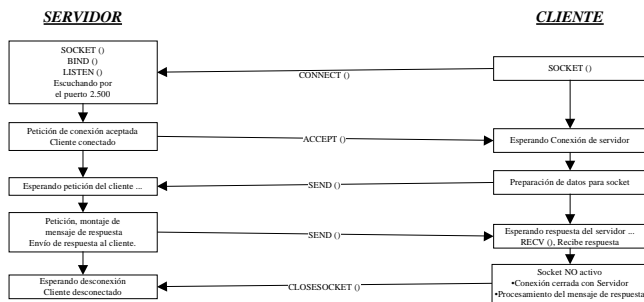


Figura 3. Desarrollo de la conexión con sockets.

2.2. Desarrollo de la conexión y formatos de mensajes.

Como se ha comentado en el apartado anterior, es necesario realizar una conexión para realizar la transferencia de mensajes. La Fig. 3 muestra un esquema de bloques en el que se pueden ver las fases de establecimiento de conexión, transferencia de mensajes y liberación de conexión que debe realizarse para cada una de las operaciones básicas de transferencia de mensajes

mostradas en la Fig. 2. Es el típico esquema de comunicación socket.

Se han definido 5 tipos de mensajes cuyo formato se muestra en la Fig.4. Los campos de cada uno de los mismos se describen a continuación:

- F1, datos de reserva, empleado en realización, consulta y modificación:

TM: Tipo de mensaje (R, M, C). T: N° de formato = 1. LN: Longitud del nombre. N: Nombre. LA1: Longitud del Apellido 1°. A1: Apellido 1°. LA2: Longitud del Apellido 2°. A2: Apellido 2°. DNI: D.N.I. DLL: Día de llegada. MLL: Mes de llegada. ALL: Año de llegada. DSA: Día de salida. MSA: Mes de Salida. ASA: Año de Salida. TH: Tipo de habitación. ?AD: ¿N° de adultos 0? ?NL: ¿N° de niños 0? ?NI: N° de niños. NHA: N° de habitaciones. ?DO: ¿Datos Opcionales 0? DO: Datos Opcionales. ?O1: ¿Byte 1 Opciones 0? O1: Byte 1 Opciones. ?O2: ¿Byte 2 Opciones. O2: Byte 2 Opciones. ?O3: ¿Byte 3 Opciones 0? O3: Byte 3 Opciones.

- F2, número de registro, empleado en cancelación, consulta y modificación:

TM: Tipo de Mensaje (M, C, S). T: N° de formato = 2. NRE = N° de Registro.

- F3, respuesta de servidor, empleado en realización, cancelación, consulta y modificación:

TM: Tipo de mensaje (R, M, C, S). T: N° de formato = 3. ?OR: ¿Operación realizada? NRE: N° de registro. LME: Longitud del mensaje. MEN: Mensaje.

- F4, nuevos datos de reserva, empleado en modificación:

TM: Tipo de mensaje (M). T: N° de formato = 4. NRE: N° de registro. DLL: Día de llegada. MLL: Mes de llegada. ALL: Año de llegada. DSA: Día de salida. MSA: Mes de salida. ASA: Año de salida.

- F5, respuesta de realización, empleado en realización:

TM: Tipo de mensaje (R). T: N° de formato = 4. NHA: N° de habitación. DLR: Delimitador de Reserva. DLF: Delimitador de Fin. RR?: Mensaje de formato 3.

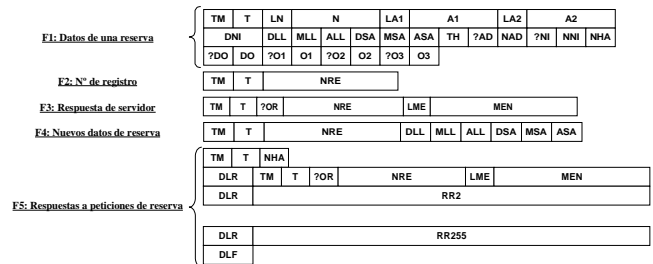


Figura 4. Formatos de mensajes.

3. CONCLUSIONES

Se ha obtenido un protocolo de comunicaciones basado en sockets que permite realizar operaciones de gestión de reservas a través de redes de comunicaciones. De esta forma se pueden crear o modificar diferentes versiones de programas de gestión de reservas a los que se les puede añadir funciones básicas para que actúen como servidores. Esto permite la gestión de reservas de remota empleando una única central de reservas a la que se conectan los clientes por medio de un navegador web de forma automática y con respuesta inmediata, lo cual constituye una novedad respecto a otras aplicaciones que tratan el mismo problema estudiadas en la red, donde la respuesta no se realiza de forma automática en tiempo real.

4. REFERENCIAS

[1] Breedlove B. "Desarrollo de aplicaciones Web al descubierto", Prentice Hall, ISBN 84-89660-80-8, 1997.
 [2] Bonner, P. "Network Programming with Windows sockets", Prentice Hall, ISBN 0-13-230152-0, 1996.