

Requisitos y Diseño

Lucía Pereira Menchero
Iker Manuel Pérez Villa

Introducción

En esta práctica llevaremos a cabo la implementación de nuestros propios servidores persistentes en c.

Por un lado, se realizará una investigación de los distintos tipos de servidores que hay y se compararán los tiempos que tarda cada uno de ellos con una cantidad considerable de clientes enviando mensajes. Implementaremos un cliente que pruebe dicha funcionalidad y unos scripts para medición de tiempo.

Tras ello, se escogerá el servidor más eficiente bajo nuestro criterio e implementaremos el servidor web en sí. Éste será capaz de recibir, analizar y responder a peticiones HTTP, implementando únicamente los métodos GET, POST y OPTIONS. Su correcto funcionamiento se podrá verificar por el propio navegador o bien, mediante el cliente que fue implementado para la parte de tiempos.

Requisitos

[REQ-S-1] El servidor debe establecer un socket en el puerto 8080 y quedar a la escucha de nuevas conexiones.

[REQ-S-2] El servidor debe ser capaz de establecer conexiones con distintos clientes y gestionar una cantidad alta de las mismas.

[REQ-S-3] El servidor debe ser capaz de soportar las peticiones http POST, GET y OPTIONS y saber analizarlas correctamente.

[REQ-S-4] El servidor debe ser capaz de responder a las peticiones que le solicita el cliente con los códigos de respuesta mínimos (200, 400 y 404).

[REQ-S-5] El servidor debe gestionar los posibles errores durante la ejecución de las peticiones y saber responder a ello con el código de respuesta adecuado.

Diseño

Nuestro servidor web se basa en un **servidor concurrente con pool de conexiones de tamaño fijo**. Cada conexión es atendida por un hilo y las sesiones son persistentes, es decir, el cliente será quien cierre la conexión y solo entonces se cerrará.

Nuestro programa servidor crea tantos hilos como conexiones como máximo se haya establecido en el fichero de configuración. Cada hilo de ejecución se queda a la espera de una conexión de entrada, y cuando un cliente se conecta, un hilo de los que estaban esperando atiende a dicho cliente.

Casos de uso

