



Introducción a Distribuidos

Taller 1

Grupo 1

Sophia Aristizábal Flórez

Alanis Forero Salas

Kevin Jaimes Carrillo

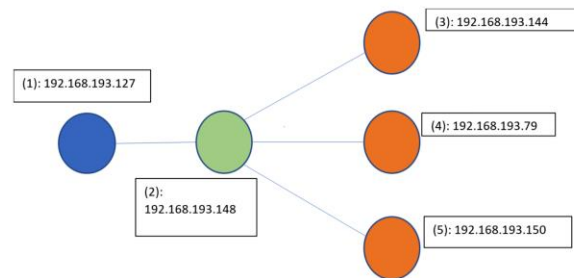
Luisa Vargas Gomez

Daniel Vidal De Leon

Bogotá D.C.

2023

A continuación, se muestra un dibujo que describe la arquitectura implementada para la realización del taller. En este se muestran las IPs asignadas para cada máquina y se usó una diferenciación de colores para identificar al cliente (azul), el servidor central (verde) y los de operaciones (naranjas).



La arquitectura construida se dispone a resolver un cálculo matemático simple del teorema de Pitágoras, donde el cliente (de color azul) envía la petición junto con el valor de los catetos y los servidores se encargan de realizar los cálculos necesarios para devolver el valor de la hipotenusa al cliente. Las máquinas se comunican por medio del socket 7000 (exceptuando el cliente), el protocolo TCP y utilizan el formato JSON para enviarse la información de la petición y la respuesta. Además, se decidió utilizar hilos para que el servidor central pudiera atender solicitudes de varios clientes al mismo tiempo.

A continuación, se describe con mayor detalle qué hace cada uno de los computadores:

Funciones de cada servidor:

CLIENTE (1): Se encarga de realizar la petición. Recibe por medio de la línea de comandos el valor de los catetos que el usuario haya decidido enviar

SERVIDORC (2): Realiza solicitudes a cada servidor de cálculo, y en el caso de no recibir una respuesta o no poder establecer conexión, procede a realizar el cálculo localmente. También este es el que devuelve la respuesta al cliente.

SERVIDOR1 (3): Recibe el primer cateto que da el cliente y lo eleva al cuadrado para devolver ese resultado al servidor central.

SERVIDOR2 (4): Recibe el segundo cateto proporcionado para luego elevarlo al cuadrado y devolverlo al servidor central para ser usado para el teorema de Pitágoras.

SERVIDOR3 (5): Recibe los dos catetos al cuadrado, los suma y saca la raíz cuadrada para sacar la hipotenusa y devolver ese cálculo al servidor central

Fallo de servidor:

Para abordar la situación de que uno de los servidores de operación se caiga o no se pueda realizar la comunicación entre alguno de ellos, se decidió utilizar la excepción del tipo Timeout de la librería Socket. Para cada petición por parte del servidor central se indicaba una duración de 1 segundo de espera para la realización del timeout y si este se cumplía el

servidor central se encargaba de realizar el cálculo respectivo que el de operaciones no pudo realizar. En la imagen se muestra que el servidor central tuvo que realizar todas las operaciones porque los servidores estaban apagados. En la imagen anexa se puede observar que el servidor central se prende, recibe una

```
[ENCENDIENDO] SERVIDOR CENTRAL
[ESCUCHANDO] Servidor escuchando en 192.168.193.148
[CONEXION NUEVA] ('192.168.193.126', 50122) conectado.
[CONEXIONES ACTIVAS] 1
[FALLO] Servidor ('192.168.193.144', 7000)
[FALLO] Servidor ('192.168.193.79', 7000)
[FALLO] Servidor ('192.168.193.150', 7000)
[ENVIADO] RESULTADO: 5.0 a ('192.168.193.126', 50122)
```

petición del cliente y obtiene fallos de timeout. En cada fallo el central hizo el cálculo respectivo y finalmente envió la respuesta al cliente.