

# Web Service Framework

Valentina Siabatto

Febrero 2020

## 1 Resumen

Este artículo muestra la implementación y arquitectura de un servidor web desarrollado en java implementando un framework. El cual soporta múltiples solicitudes seguidas y retorna los archivos solicitados.

## 2 Introducción

El objetivo de este laboratorio es crear un servidor web usando un framework implementado usando reflexion y anotaciones en java. Esta implementación del framework permite realizar peticiones a recursos estáticos y dinámicos, además es posible para el desarrollador crear sus métodos para responder peticiones usando anotaciones en java. El servidor web soporta múltiples solicitudes seguidas del cliente y atenderlas adecuadamente.

## 3 Contenido

Usando los elementos de Java se realizó un servidor que respondiera las peticiones de un cliente. Este fue implementado a través de sockets. Los sockets permiten establecer un enlace entre dos máquinas independientes. La librería Java.net provee dos clases para implementar la conexión desde el lado del Cliente (Socket) y desde el lado del servidor (ServerSocket). El servidor que se ejecuta está vinculado a un número de puerto específico, el servidor espera escuchando el socket para que el cliente haga una solicitud de conexión.[1]

Para permitir al desarrollador crear sus propios métodos para responder las peticiones, se creó una anotación "web", la cual fue utilizada en métodos que se quisieran para responder alguna petición del cliente.[2] Al iniciar el servicio, se busca entre los directorios de la aplicación, las clases que tengan métodos con la anotación "web" y se agregan a una lista donde se encuentran todos los métodos que tengan esta anotación, de manera que cuando se reciba una petición, se busque en la lista si existe allí la petición requerida. Una vez hecho este proceso, el servidor se pondrá a escuchar un puerto hasta que un cliente establezca la conexión con el servidor usando un socket y el puerto específico,

una vez establecida la conexión entre el cliente y el servidor, el cliente es capaz de hacer peticiones al servidor.

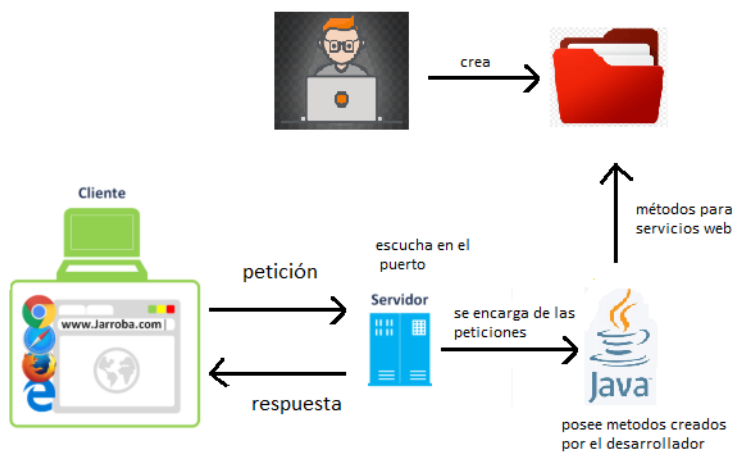


Figure 1: Cliente Servidor

Al llegar estas peticiones al servidor, el encabezado http se desglosa por partes para poder reconocer el tipo de solicitud que necesita. Se verifica si es un petición de un recurso dinámico o estático, si la petición del servidor esta en la lista de los metodos de la anotación "web", se invocará el método correspondiente y se retorna el contenido del método, en caso de que no se encuentre en esta lista, se reconoce el recurso que hace referencia la petición, si es una imagen (png o jpg) o un html, y si este existe. En caso de que no exista, se retorna una página explicando que el método no esta soportado. Del mismo modo, Si el recurso existe se retorna la respuesta al cliente con el recurso que solicitó. De esta forma el servidor web recibe las peticiones y las responde.

## 4 Conclusiones

El servidor fue capaz de resolver las peticiones enviadas por el cliente. La implementación del servidor fue capaz de resolver las solicitudes usando los metodos creados por el desarrollador para responder a las peticiones, gracias a la reflexión y anotación en java, además de la información del encabezado http. Es posible implementar y extender el código para crear otras anotaciones que respondan otro tipo de peticiones desde el servidor.

## References

- [1] Escuela Colombiana de Ingenieria. Introduccion a esquemas de nombres, redes, clientes y servicios con java.
- [2] Escuela Colombiana de Ingenieria. Meta-\*, reflexión y anotaciones.