

DOCUMENTO DE DISEÑO Y ARQUITECTURA

Fecha: 05/11/2017

Versión: 1.0

Responsable: Mario Arias Escalona

ÍNDICE

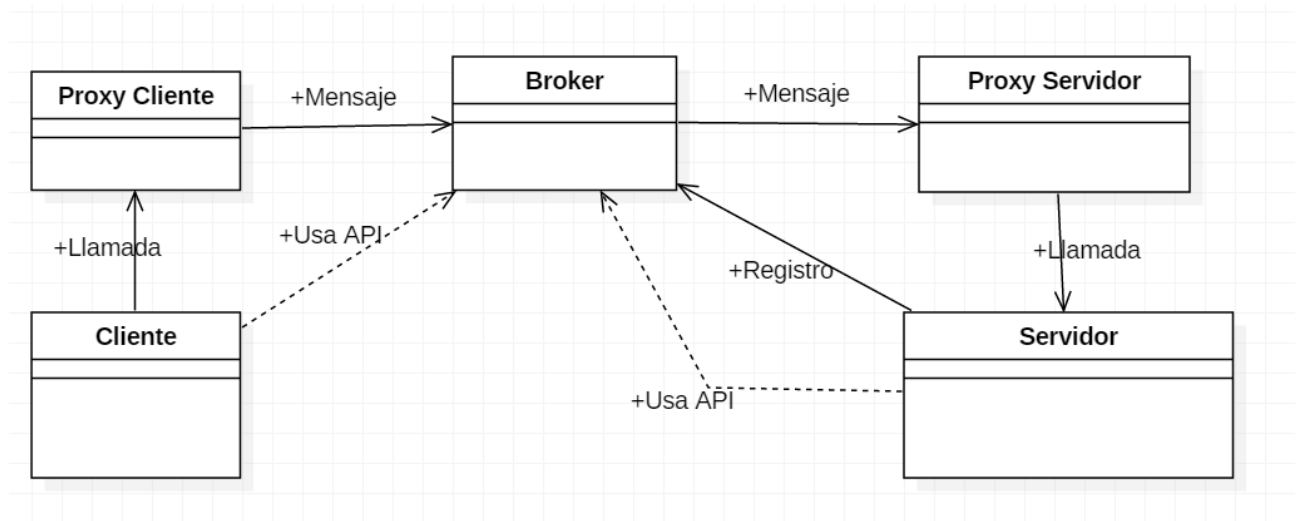
1	ESTILO ARQUITECTÓNICO	1
1.1	Presentación.....	1
1.2	Justificación.....	2
2	DISEÑO.....	2
2.1	Diagrama del Sistema	3
2.2	Diagrama de despliegue.....	Error! Bookmark not defined.

1 ESTILO ARQUITECTÓNICO

El estilo arquitectónico seleccionado para la construcción de la solución es el patrón **Broker**.

1.1 Presentación

El patrón de diseño Broker se utiliza para organizar sistemas distribuidos con componentes desacoplados que interaccionan realizando invocaciones remotas a servicios. El mediador o "Broker" es responsable de coordinar la comunicación, tanto de enviar las peticiones, como de transmitir los resultados y las excepciones. A continuación, se muestra de manera simplificada los diferentes elementos de un sistema bróker estándar:



1.2 Justificación

El patrón bróker es útil para sistemas que evolucionan y crecen con el tiempo: el “bróker” se encarga de que podamos acceder a los distintos servicios del sistema sin necesidad de conocer su localización. En el caso que nos atañe, el sistema puede ir ampliándose ofreciendo cada vez más servicios, por lo que esta implementación nos permitiría que el sistema pudiera crecer de manera fácil y compacta.

Frente a una arquitectura clásica cliente-servidor el sistema nos ofrece más flexibilidad, mantenibilidad y adaptabilidad. Si alguno de los servicios fallase los demás podrían seguir funcionando de manera independiente.

El bróker se encargaría de abstraer al cliente donde están los servidores apropiados para un servicio en concreto. Esto facilitaría mucho la implementación del lado del cliente. Por la parte del servidor únicamente se debe encargar de realizar una determinada tarea y proporcionar el servicio solicitado por el bróker. Toda la lógica de comunicación queda centralizada en un mismo punto desacoplando en gran medida todos los componentes del sistema.

2 DISEÑO

Cliente: Aplicación de Twitter que accede a los servicios de al menos un servidor. Para invocar servicios remotos, el cliente enviará solicitudes al bróker. Después de que la operación se haya ejecutado, el cliente recibirá la respuesta o excepción del bróker. No necesita conocer la ubicación de los servidores a los cuales solicita un determinado servicio. Esto permite la agregación de nuevos servicios existentes a otras ubicaciones, aun cuando el sistema está siendo ejecutado.

Servidor: Implementa objetos que exponen su funcionalidad a través de una interfaz que contiene operaciones y atributos. Es el lugar donde están alojados los servicios de tratamiento de imagen del sistema.

Broker: Es un mensajero, responsable de la transmisión de solicitudes de clientes y servidores, así como de la transmisión de respuestas. Una vez iniciado el bróker, este se quedará a la escucha de cualquier solicitud que el cliente haga. El bróker siempre estará en ejecución.

Proxy-Cliente: Representa una capa adicional entre los clientes y el bróker, para proveer de transparencia en el sentido que un objeto remoto aparece como local ante el cliente, es decir, esconde los detalles de implementación.

Proxy-Servidor: Es el responsable de recibir solicitudes, desempaquetar los mensajes de entrada, llamar al servicio apropiado y de hacer el marshaling de resultados (transformar la representación en memoria de un objeto a un formato apropiado para almacenaje o transmisión).

2.1 Diagrama del Sistema

