Configuración de servicios

**Fecha:** 04/11/2017

**Versión:** 1.0

**Responsable:** Mario Arias Escalona

# Funcionamiento

Para el correcto funcionamiento del sistema todos los servicios que este pueda proporcionar serán almacenados en una base de datos no-relacional a la cual tendrá acceso tanto el bróker como el administrador del sistema.

* **Broker:** Realiza peticiones para recibir información de un determinado servicio. Si un determinado servicio se encuentra inactivo actualiza el estado de dicho servicio en la base de datos.
* **Administrador del sistema:** Puede realizar consultas, añadir y borrar servicios. En el caso de que un servicio vuelva a estar estable después de una caída, es el encargado de cambiar el estado de dicho servicio de manera manual.

## Base de datos no relacional

Dado a que solamente necesitaremos una tabla que contenga información, y esta no necesita relacionarse con ninguna otra, se ha tomado la decisión de utilizar una base de datos no relacional. Este tipo de tecnología si bien no estructura los datos de la misma forma que las relacionales, el acceso es mucho más rápido y dado que el sistema debe responder de manera casi inmediata, se ha optado por este tipo de almacenamiento.

## Mongo DB

MongoDB ha sido creado para brindar escalabilidad, rendimiento y gran disponibilidad, escalando de una implantación de servidor único a grandes arquitecturas complejas de centros multidatos. MongoDB brinda un elevado rendimiento, tanto para lectura como para escritura, potenciando la computación en memoria.

Es una base de datos ágil que permite a los esquemas cambiar rápidamente cuando las aplicaciones evolucionan, proporcionando siempre la funcionalidad que los desarrolladores esperan de las bases de datos tradicionales, tales como índices secundarios, un lenguaje completo de búsquedas y consistencia estricta.

## Esquema

Para almacenar la información en la base de datos seguiremos el siguiente esquema en formato JSON:

[

{

\_id: '{{ServiceID()}}',

serviceName: '{{serviceName()}}',

server: '{{server()}}',

port: '{{port()}}',

status: '{{status()}}',

description: '{{description()}}',

input\_parameters: '{{parameters()}}',

output: '{{output()}}'

}

]