CONFIGURACIÓN DE NUEVOS SERVICIOS

# Características generales

Para registrar un nuevo servicio en el sistema, este debe cumplir una serie de especificaciones técnicas para que el bróker pueda consumir correctamente los servicios, y más tarde redirigir la respuesta hacia el cliente.

# Arquitectura del servicio

La arquitectura donde estará alojado el servicio debe ser del tipo API REST. Las implicaciones de esto se expondrán a continuación.

Es un tipo de arquitectura de desarrollo web que se apoya totalmente en el estándar HTTP.

REST se compone de una lista de reglas que se deben cumplir en el diseño de la arquitectura de una API.

## Llamadas a la API

Las llamadas al API se implementan como peticiones HTTP, en las que:

La URL representa el **recurso:**

* Ej. http://www.webdeejemplo.com/api/recurso/1

El método (HTTP Verbs) representa la **operación**:

* Ej. GET http://www.webdeejemplo.com/api/recurso

El código de estado HTTP representa el **resultado**:

* Ej. 200 OK HTTP/1.1
* EJ. 404 NOT FOUND HTTP/1.1

## HTTP VERBS

En este caso únicamente utilizaremos unos de los verbos de CRUD (Crear, Leer, Actualizar y Borrar), ya que el bróker no se encarga de administrar las API’s, únicamente solicita un servicio en particular. Por este motivo se utilizará:

* **GET**: Para consultar y leer recursos.

Por lo tanto, lo único que debe tener el recurso para poder ser consumido por el sistema es tener implementado el método GET.

## Nombres de los recursos

Para que el sistema sea lo más uniforme posible, a la hora de añadir un nuevo servicio se deben seguir un seguido de buenas prácticas para asignarle los nombres a los recursos.

* Plural mejor que singular, para lograr uniformidad:
  + Obtenemos un listado de imagenes: GET /v1/imagenes
  + Obtenemos una imagen en partícular: GET /v1/imagen/1
* Url's lo más cortas posibles.
* Evitar guiones y guiones bajos.
* Deben ser semánticas para el cliente.
* Utiliza nombres y no verbos.
* Estructura jerárquica para indicar la estructura: /v1/imagen/1234/descripcion