

# Servicios REST : Modelo de Madurez de Richardson

## Nivel 0: Uso del protocolo HTTP

El punto de inicio del modelo de madurez es usar el protocolo HTTP como sistema de transporte para una interacción remota, sin usar nada más de la web. Esto se puede describir como solo el uso de un método http y una sola url para la invocación de un programa. Un ejemplo es la invocación un servicio RPC

Request	Response
<pre>POST / WsSunatRUCService ###Mas datos del header &lt;consultaRUCRequest&gt;   &lt;ruc&gt;10423485672&lt;/ruc&gt; &lt;/consultaRUCRequest&gt;</pre>	<pre>HTTP 1.1 200 OK &lt;consultaRUCResponse&gt;   &lt;contribuyente&gt;     &lt;ruc&gt;10423485672&lt;/ruc&gt;   &lt;razonSocial&gt;Supermercados peruanos&lt;/razonSocial&gt;   &lt;/contribuyente &gt; &lt;/consultaRUCResponse&gt;</pre>

## Nivel 1: Recursos (Muchas URI solo un método HTTP)

El primer paso para el modelo de madures de Richardson es la introducción de recursos. A diferencia del nivel 0 ahora nuestros request son dirigidos a recursos individuales a través de su única URI

Request	Response
<pre>#Usar GET o POST u otro para traer el recurso HTTP 1.1 POST /ruc/10423485672 ###Mas datos del header</pre>	<pre>HTTP 1.1 200 OK [{"ruc": "10423485672","razonsocial": "Supermercados Peruanos"}]</pre>

## Nivel 2: Metodos HTTP

En este nivel, el modelo no indica que se deben usar los metodos HTTP para realizar acciones sobre los recursos.

GET	Obtener la representación del recurso de acuerdo al URI indicado
POST	Crear un nuevo recurso
PUT	Actualiza los datos del recurso representado por el URI
DELETE	Para Borrar el recurso asociado a la URI

### Request

```
HTTP 1.1
GET /ruc/10423485672
// Header
```

### Response

```
HTTP 1.1 200 OK
/ruc/10423485672
// Header
[{"ruc": "10423485672", "razonsocial": "Supermercados Peruanos", "direccion": "Calle las Flores 235 ...."}]
```

### Request

```
HTTP 1.1
PUT /ruc/10423485672
// Header
[{"direccion": "Calle Morelli 1001 ...."}]
```

### Response

```
HTTP 1.1 200 OK
location : /ruc/10423485672

[{"ruc": "10423485672", "razonsocial": "Supermercados Peruanos", "direccion": "Calle Morelli 1001 ...."}]
```

## Nivel 3: Hypermedia

El nivel 3 introduce los conceptos de controles Hypermedia. La idea central es que después de que el cliente realiza un request, el recurso solicitado nos trae la información de sus capacidades, interconexiones. Esto se llama HATEOS (Hypermedia as the engine of application state)

este principio indica que el cliente puede interactuar con los recursos a travez de una serie de hyperlinks proporcionados por el recurso, el cliente REST no necesita conocer de manera particular como debe interactuar con el servidor mas alla de una compresion generica de hypermedia.

La importancia de HATEOAS radica en el desacoplamiento del cliente y del servidor, de manera que los componentes del servidor (API REST) pueda evolucionar en funcionalidad de manera independiente.

### Request

```
HTTP 1.1
GET /contribuyentes
// Header
```

### Response

```
HTTP 1.1 200 OK
GET /contribuyentes
// Header
[ {
  "contribuyente": [ { "ruc": "10423485672",
    "razonsocial": "Supermercados Peruanos",
    "direccion" : "Calle las Flores 235 ....",
    "links": [
      { "rel" : "self"
        "href": "/contribuyentes/10423485672" },
      { "rel" : "accionesfiscalizacion"
        "href": "/contribuyentes/10423485672/accionesfiscalizacion" },
    ]
  },
  { "ruc": "10763348861",
    "razonsocial": "Saga Falabella",
    "direccion" : "Calle las Begonias 702 ....",
    "links": [
      { "rel" : "self"
        "href": "/contribuyentes/10763348861" },
      { "rel" : "accionesfiscalizacion"
        "href": "/contribuyentes/10763348861/accionesfiscalizacion" },
    ]
  }
]
}]
```

