

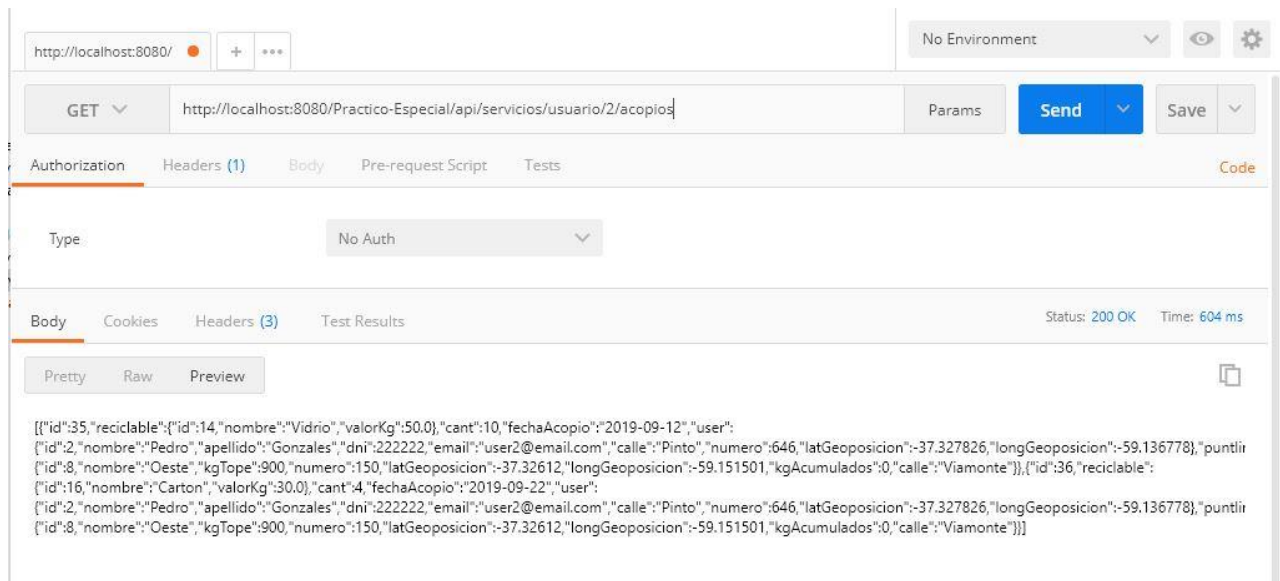
Ejecución de Servicios REST

A continuación mostraremos los resultados de la ejecución de los servicios REST

Acopios Por Usuario: Dado el id de un Usuario devuelve todas las deposiciones de residuos realizadas.

Path: /servicios/usuario/{id}/acopios

Captura de la ejecución:



Como se ve el resultado devuelto es un Json que contiene los acopios realizados por un usuario, en este caso el usuario es el número 2.

El Json de resultado tiene el siguiente formato:

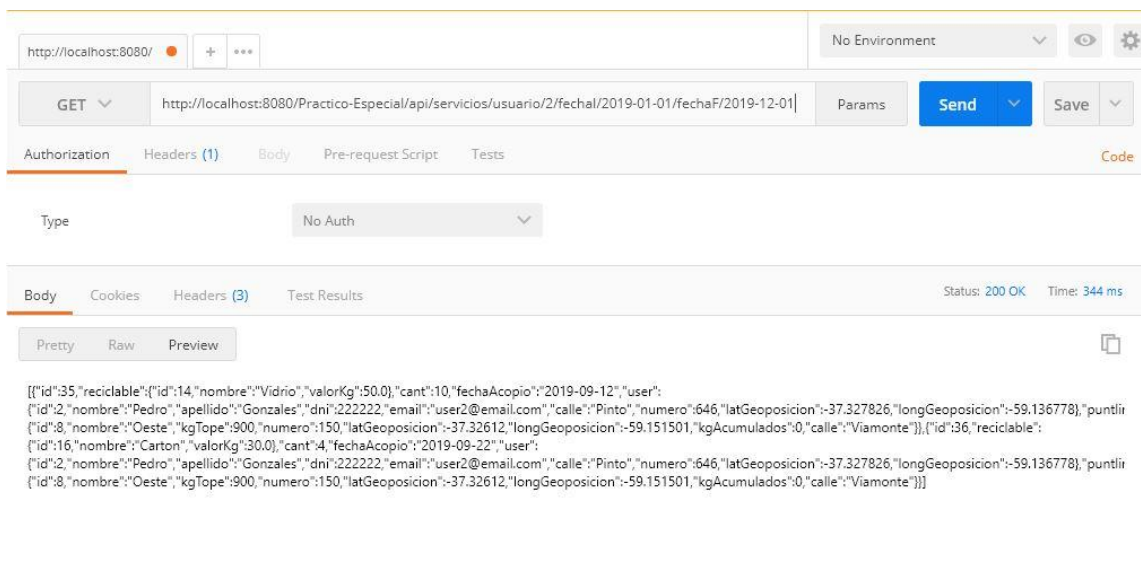
```
{  
  id del acopio;  
  residuo depositado  
  {id de residuo, nombre del residuo, valor en $ por Kg};  
  Cant. De residuo depositado;  
  Fecha del depósito;
```

```
Usuario que lo realizo
{datos del usuario};
punto limpio donde se realizo el deposito
{datos punto limpio}
}
```

Acopios por usuario y rango de fechas: Devuelve las deposiciones realizadas por un Usuario dentro de un rango de fechas pasado por parámetro

Path: /servicios/usuario/{id}/fechal/{fecIn}/fechaF/{fecF}

Captura de la ejecución:



Como se ve el resultado devuelto es un Json que contiene los acopios realizados por un usuario dentro de un rango de fechas, en este caso el usuario es el número 2 y las fechas son 2019-01-01 al 2019-12-01 (formato de fecha YYYY-MM-DD).

El Json de resultado tiene el siguiente formato:

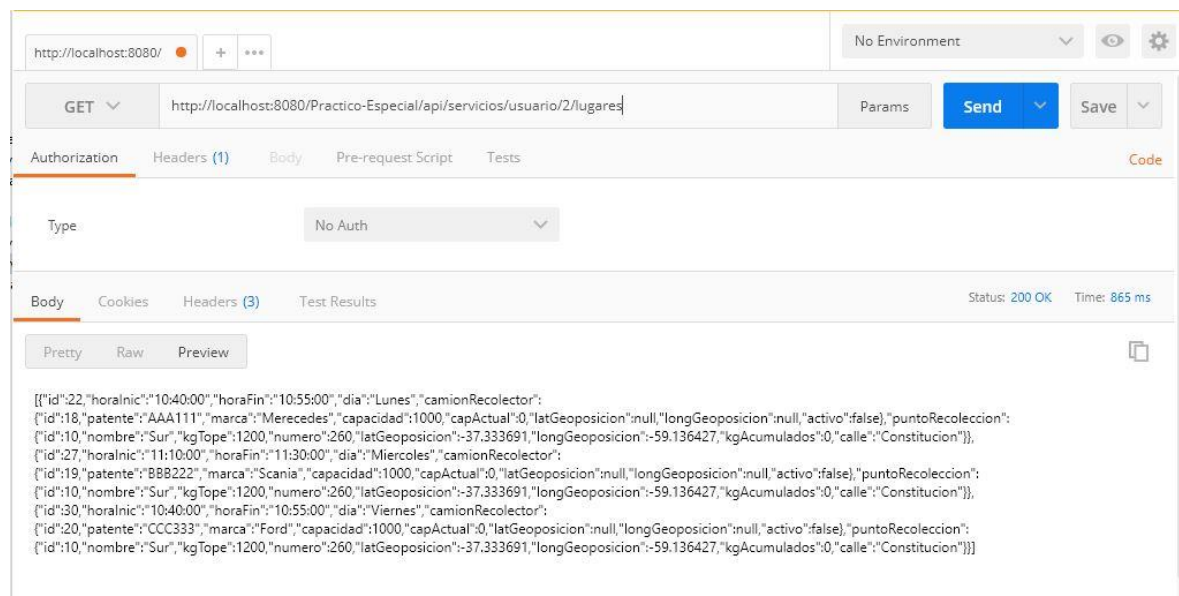
```
{
id del acopio;
residuo depositado
{id de residuo, nombre del residuo, valor en $ por Kg};
Cant. De residuo depositado;
Fecha del depósito;
```

```
Usuario que lo realizo
{datos del usuario};
punto limpio donde se realizó el deposito
{datos punto limpio}
}
```

Lugares de acopio para un Usuario: Este servicio devuelve en base a la geoposición de un Usuario los lugares de acopio (Punto Limpio) más cercanos con los datos de recolección de estos puntos.

Path: /servicios/usuario/{id}/lugares

Captura de la ejecución:



El resultado devuelto es un listado de Recolectores , estas, son los puntos de recolección por los que pasa un camión, en este caso estos puntos son los más cercanos a la geolocalización del usuario enviado por parámetro.

El resultado tiene el siguiente formato:

```
{
  Id de recolección;
  horaln: hora inicio de comienzo de recolección
  horaFin; hora fin de recolección
}
```

día de la recolección

{Datos camión(id, patente, etc)}

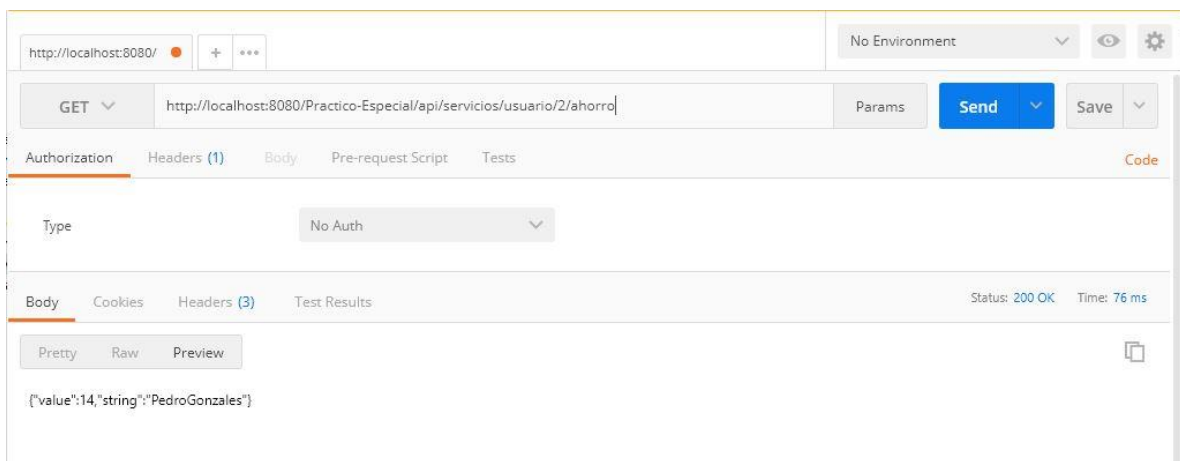
{Datos punto limpio (id, dirección, etc) }

}

Ahorro por Usuario: Dado un Usuario devuelve la cantidad de Kg de residuos por él depositados en puntos limpios.

Path:/servicios/usuario/{id}/ahorro

Captura de la ejecución:



El resultado devuelto es un Json que contiene el nombre del usuario y la cantidad en kg de residuos que reciclo.

{

Value: Cant en KG

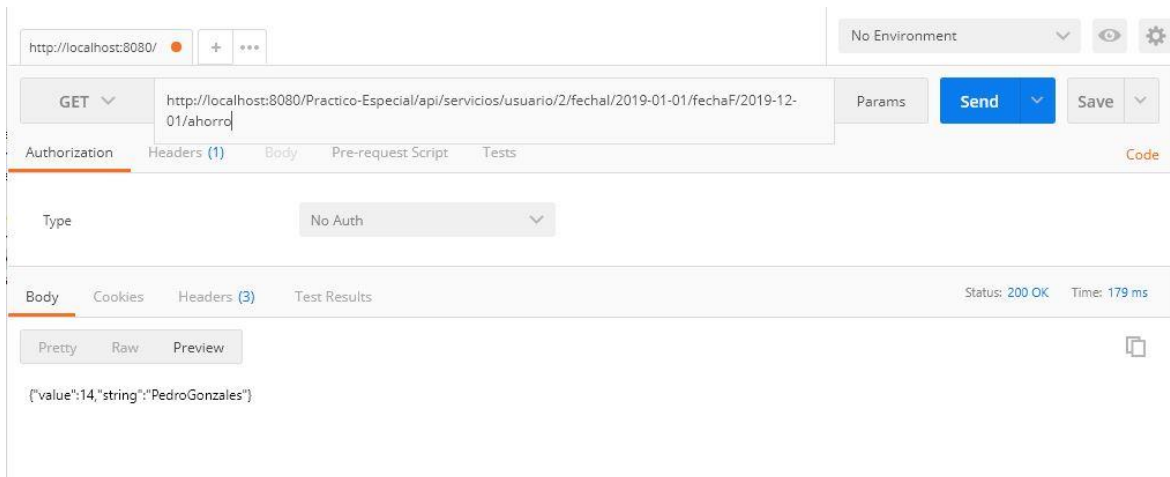
String: nombre del usuario

}

Ahorro por Usuario en rango de fechas: Dado un Usuario devuelve la cantidad de Kg de residuos por él depositados en puntos limpios dentro de un rango de fechas dado.

Path: /servicios/usuario/{id}/fechal/{fecIn}/fechaF/{fecF}/ahorro

Captura de la ejecución:



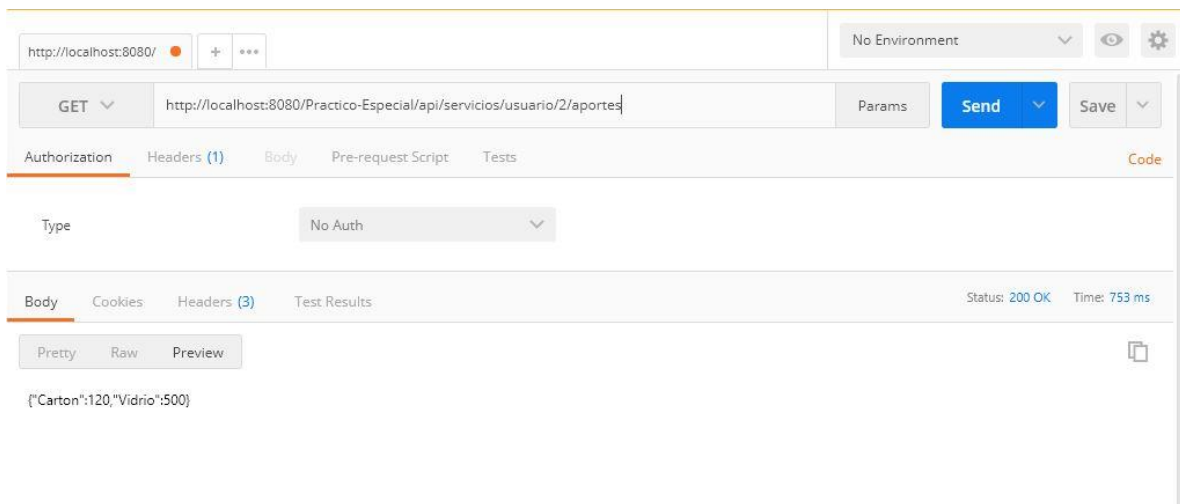
El resultado devuelto es un Json que contiene el nombre del usuario y la cantidad en kg de residuos que reciclo en el rango de fechas especificado.

```
{  
  Value: Cant en KG  
  String: nombre del usuario  
}
```

Aportes por Usuario: dado un usuario devuelve un conjunto de tuplas que describen el tipo de residuos que deposito y la cantidad para cada uno.

Path: /servicios/usuario/{id}/aportes

Captura de la ejecución:



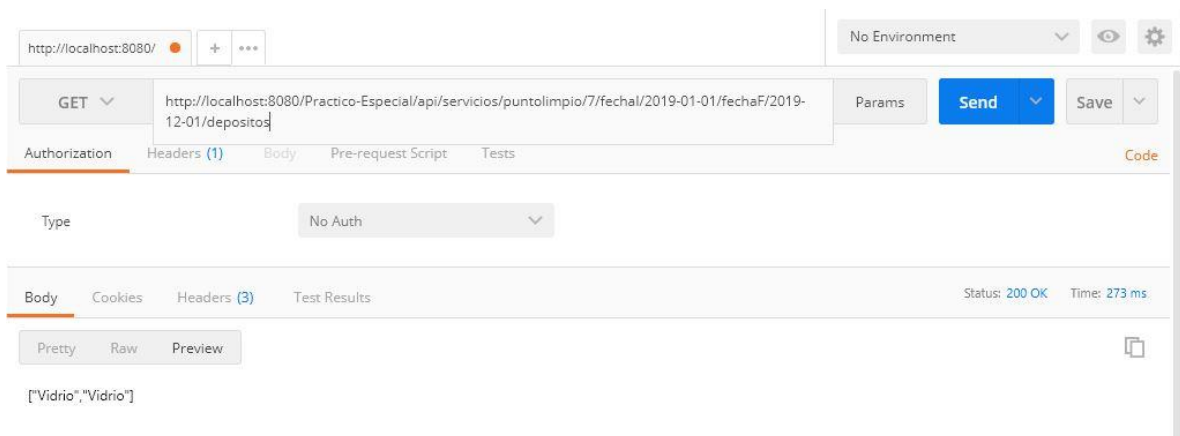
El resultado devuelto es Hastable que contiene el nombre del residuo como clave y la cantidad en kg como valor que reciclo un usuario.

```
{  
  "Nombre residuo": Cant en kg;  
}
```

Residuos por Punto Limpio y rango de fecha: Retorna un listado de los residuos procesados por un punto limpio dentro de un rango de fechas.

Path: /servicios/puntolimpio/{id}/fechaI/{fecI}/fechaF/{fecF}/depósitos

Captura de la ejecución:



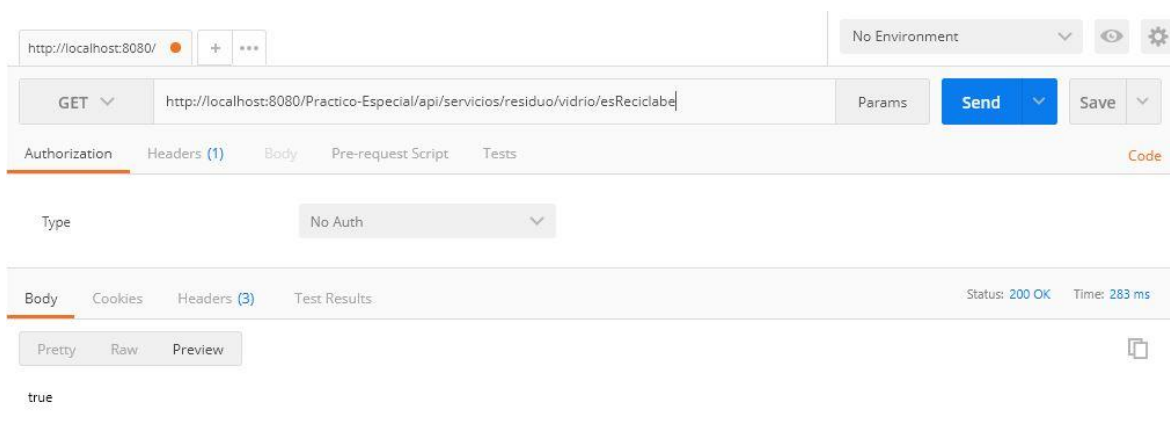
El resultado devuelto es un listado con el nombre de cada uno de los residuos que fue depositado y procesado por un punto limpio dentro de un rango de fecha.

{nombre residuo; ...}

Recurso reciclable: Dado un nombre retorna si el mismo se corresponde con el de un residuo reciclable.

Path: /servicios/residuo/{resid}/esReciclabe

Captura de la ejecución:

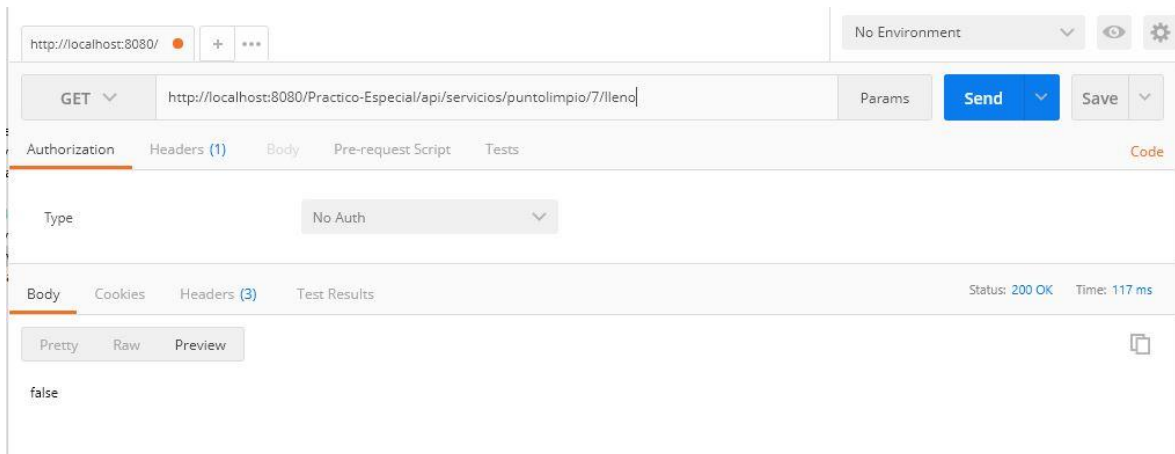


Devuelve true o false si un residuo es reciclable.

Punto Limpio lleno: Dado un Punto Limpio retorna si el mismo llego al límite de su capacidad.

Path: /servicios/puntolimpio/{id}/lleno

Captura de la ejecución:

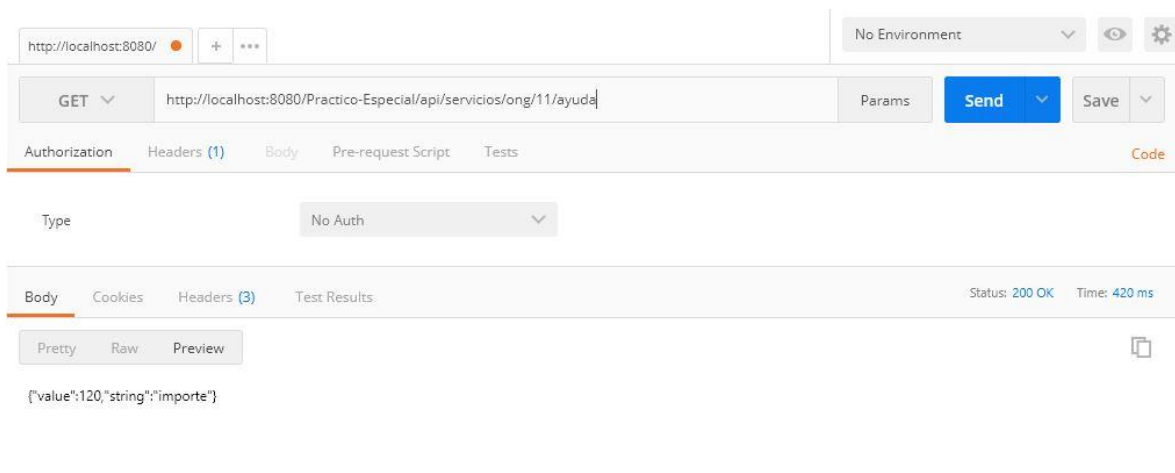


Retorna true o false si un punto limpio se encuentra al límite de su capacidad

Ayuda recibida por una Ong: dado el id de una Ong retorna el monto recibido en concepto de donaciones de residuos.

Path: `/servicios/ong/{id}/ayuda`

Captura de la ejecución:



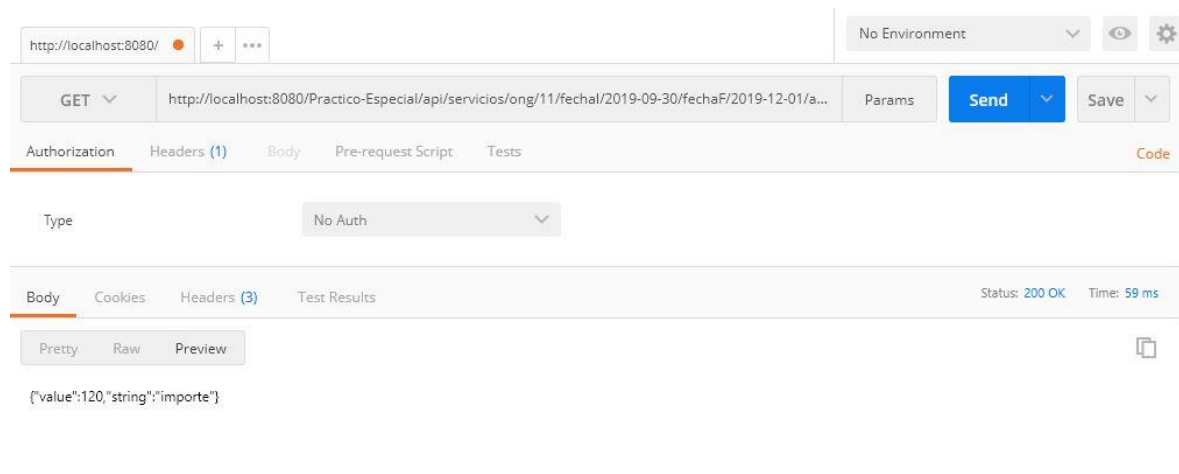
La respuesta es una tupla del tipo de resultado la cual es un Json que contiene un campo string y un campo numérico, en este caso sería:


```
{  
  value: importe en pesos del dinero equivalente recibido en donaciones de residuos  
  String: Concepto que representa el valor numérico, en este caso es un importe.  
}
```

Ayuda recibida por una Ong en un rango de fechas: dado el id de una Ong retorna el monto recibido en concepto de donaciones de residuos dentro de un rango de fechas pasado por parámetro.

Path: /servicios/ong/{id}/fechal/{fecIn}/fechaF/{fecF}/ayuda

Captura de la ejecución:



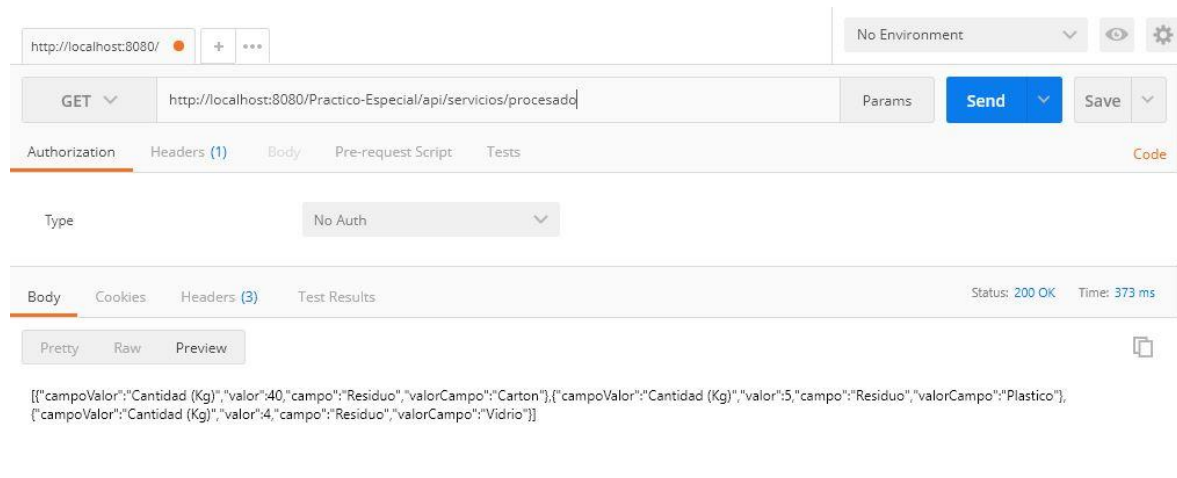
La respuesta es una tupla del tipo de resultado la cual es un Json que contiene un campo string y un campo numérico, en este caso seria:

```
{  
  value: importe en pesos del dinero equivalente recibido en donaciones de residuos en el  
  rango especificado  
  String: Concepto que representa el valor numérico, en este caso es un importe.  
}
```

Residuos procesados: Retorna una listas de tuplas punto limpio residuo y cantidad procesados por todos los punto limpio.

Path: /servicios/procesado

Captura de la ejecución:



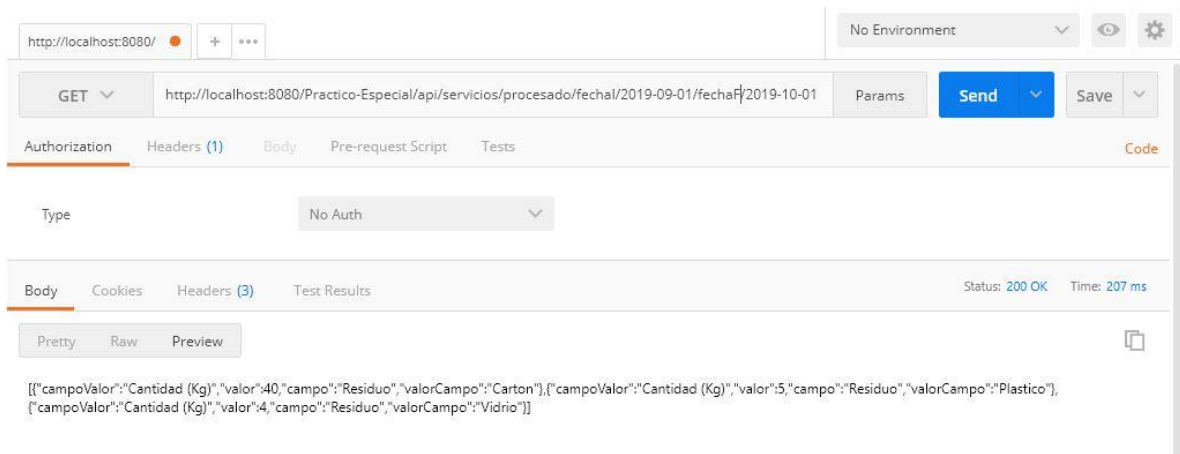
En este caso las tuplas devueltas contienen los datos de los residuos procesados por todos los puntos limpios el formato es el siguiente:

```
{
  Campo: Especifica el recurso, en este caso residuo.
  Valor campo: Especifica el tipo del recurso, por ejemplo Cartón
  campoValor: Especifica que se detalla del recurso, en este caso Cantidad en Kg
  Valor: el valor numérico del campo valor.
}
```

Residuos procesados por fecha: Retorna una listas de tuplas punto limpio residuo y cantidad procesados por todos los punto limpio dentro de un rango de fechas.

Path: /servicios/procesado/fechaI/{fecI}/fechaF/{fecF}/

Captura de la ejecución:



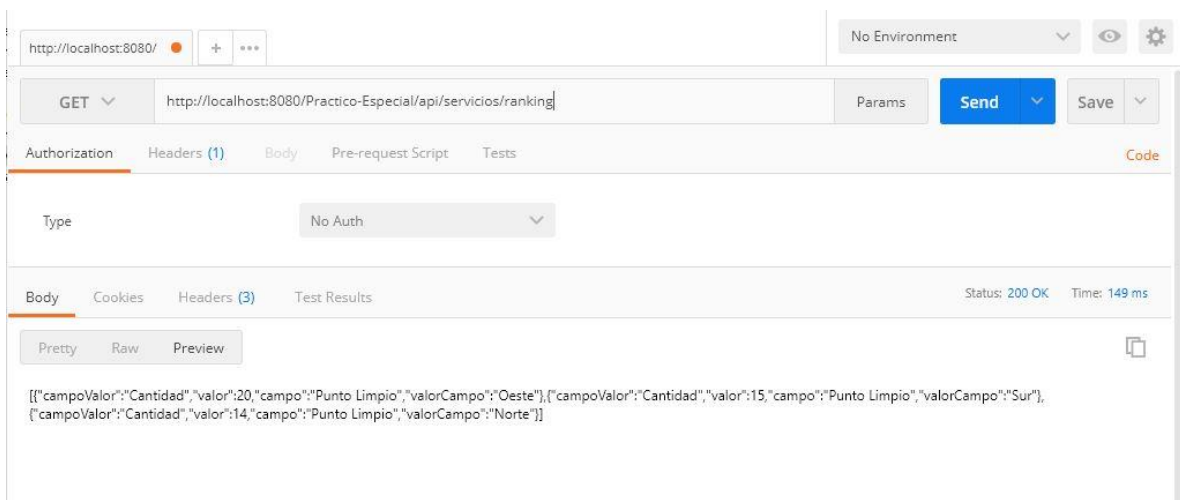
En este caso las tuplas devueltas contienen los datos de los residuos procesados por todos los punto limpio dentro del rango especificado el formato es el siguiente:

```
{
  Campo: Especifica el recurso, en este caso residuo.
  Valor campo: Especifica el tipo del recurso, por ejemplo Cartón
  campoValor: Especifica que se detalla del recurso, en este caso Cantidad en Kg
  Valor: el valor numérico del campo valor.
}
```

Ranking Punto Limpio: retorna el listado de Puntos limpios ordenado en forma descendente según la cantidad de residuos procesados.

Path: `/servicios/ranking`

Captura de la ejecución:



El resultado es un listado con los puntos limpios y los kg que proceso en orden descendente, el formato es el siguiente.

{

Campo: Especifica el recurso, en este caso Punto Limpio.

Valor campo: Especifica el nombre del recurso, por ejemplo Oeste

campoValor: Especifica que se detalla del recurso, en este caso Cantidad en Kg

Valor: el valor numérico del campo valor.

}