

Lista de Sucursales - [listOfErpBranchs]

? Descripción

Obtiene la lista completa de **sucursales registradas en el ERP**, filtrando únicamente aquellas que:

- No sean concesionarios (`concesionario = 0`)
- Tengan un identificador válido (`ideidentificador != "0"`)

El servicio consulta directamente el ERP a través del método `apiService()` y devuelve las sucursales ordenadas alfabéticamente por su nombre.

También realiza un post-proceso convirtiendo el campo `ideidentificador` en **string**, garantizando consistencia en el frontend (evita problemas cuando el ERP envía valores numéricos).

? Endpoint

GET `/list-of-erp-branches`

No recibe parámetros.

Toda la data proviene del ERP mediante:

```
GET Branch?limit=None&order_by=name asc&fields=["name","ideidentificador"]&filters=[["concesionario","=",0],["ideidentificador","!=", "0"]]
```

? Seguridad

Requiere autenticación del ERP mediante `apiService()`.

El token se gestiona internamente (servicio backend-backend).

? Flujo del Servicio

1. Llama al ERP solicitando todas las sucursales:
 - Sin límite (`limit=None`)
 - Ordenadas (`order_by=name asc`)
 - Filtradas por:
 - `concesionario = 0`
 - `ideidentificador ≠ "0"`
 - Campos solicitados:
 - `name`
 - `ideidentificador`
 2. Decodifica la respuesta.
 3. Valida que exista el campo `"data"` en la respuesta del ERP.
 - Si no existe, retorna error.
 4. Recorre cada registro y convierte `ideidentificador` a string.
 5. Retorna la lista final de sucursales.
-

? Request Body

No tiene body.

```
{}
```

? Response 200 – Ejemplo

```
{
  "valor": true,
  "msn": "Lista de Sucursales generado correctamente",
  "data": [
    {
      "name": "Sucursal Lima",
      "ideidentificador": "101"
    },
    {
      "name": "Sucursal Arequipa",
      "ideidentificador": "205"
    }
  ]
}
```

? Posibles Errores

1. Error al obtener data del ERP

```
{
  "valor": false,
  "msn": "Error al traer la lista de sucursales",
  "data": []
}
```

2. Excepción interna del servidor

```
{
  "valor": false,
  "msn": "Error del servidor",
  "data": []
}
```

? Esquema de datos (Branch)

Branch (GET)

Campos utilizados:

```
{
  "name": "string",
  "ideidentificador": "string | number"
}
```

? Pseudocódigo del servicio

```
response = apiService("Branch?limit=None&order by=name  
asc&fields=[name,ideentificador]&filters=[[concesionario,=,0],[ideentificador,!=,0]]")  
  
if !response.data:  
    return error  
  
foreach branch in response.data:  
    branch.ideentificador = str(branch.ideentificador)  
  
return success(branch)
```

Revisión #1

Creado 2025-11-25 15:35:52 -05 por Elian

Actualizado 2025-11-25 15:38:26 -05 por Elian